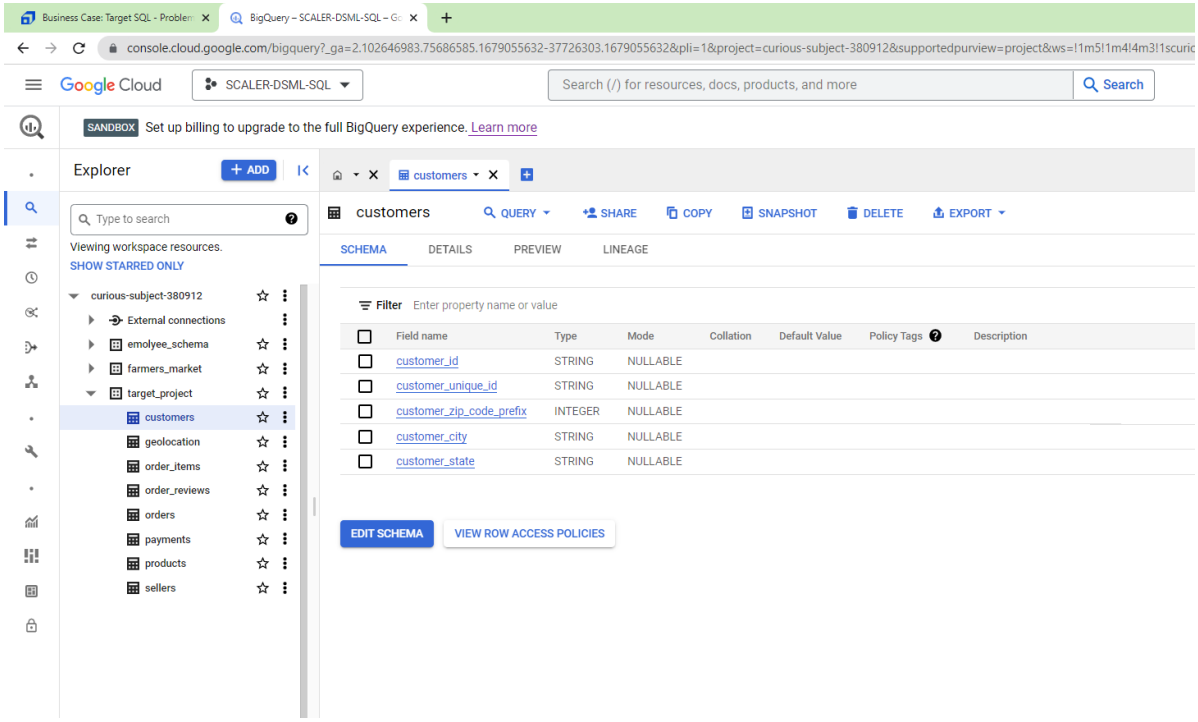
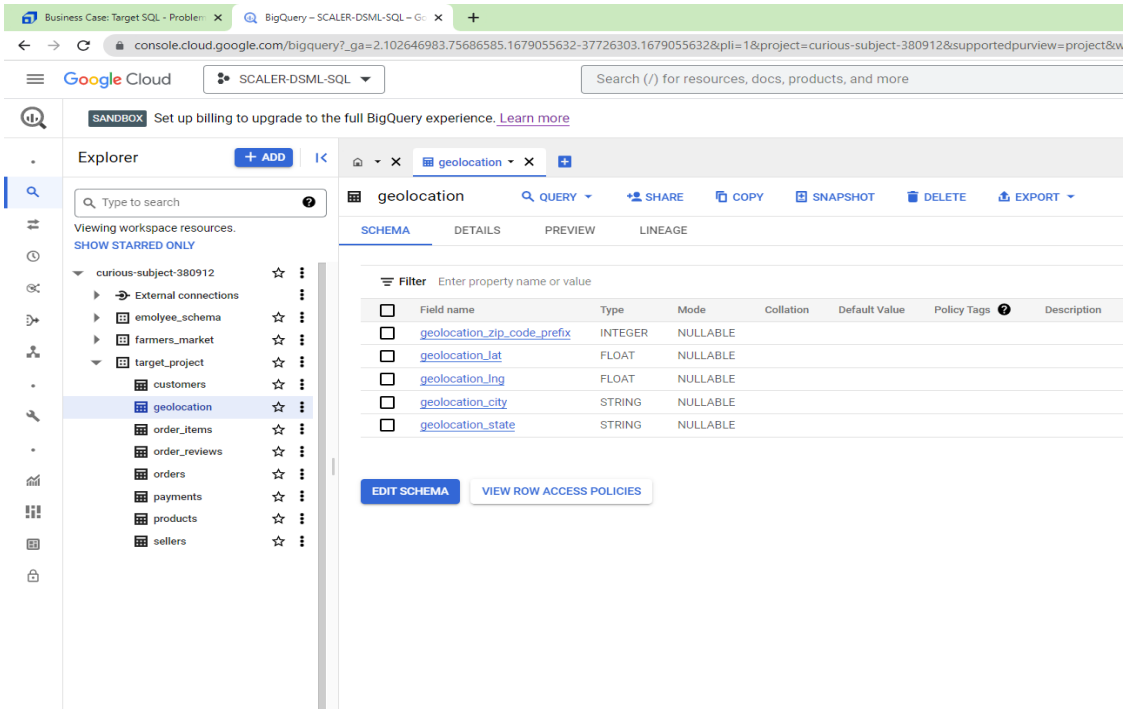
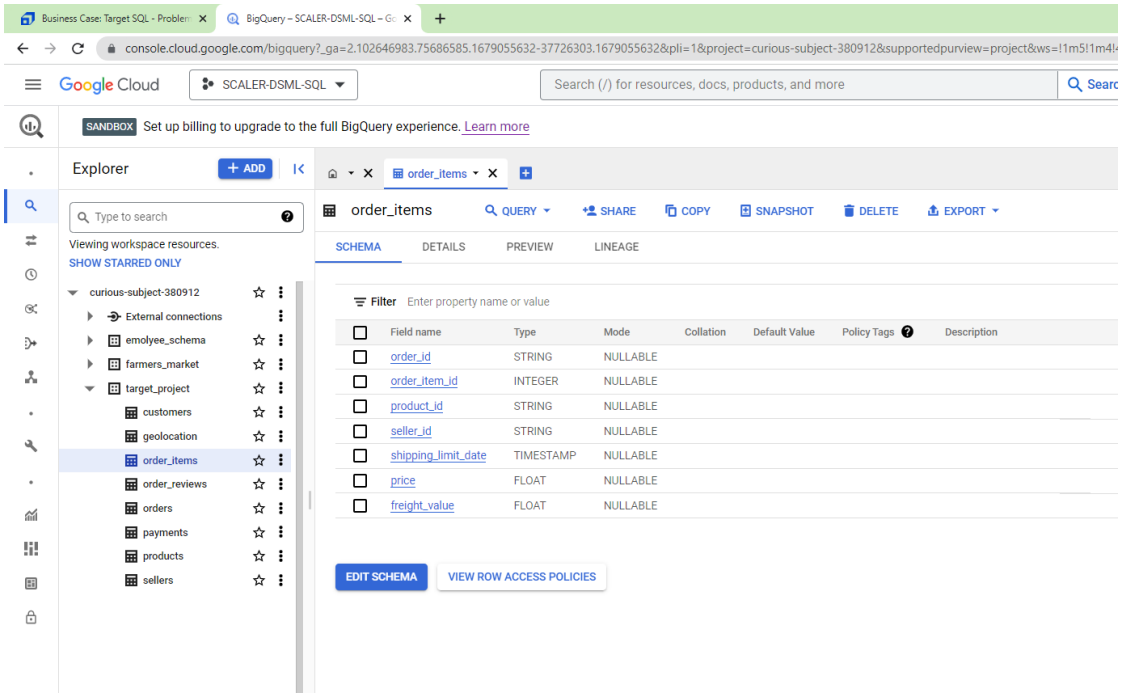
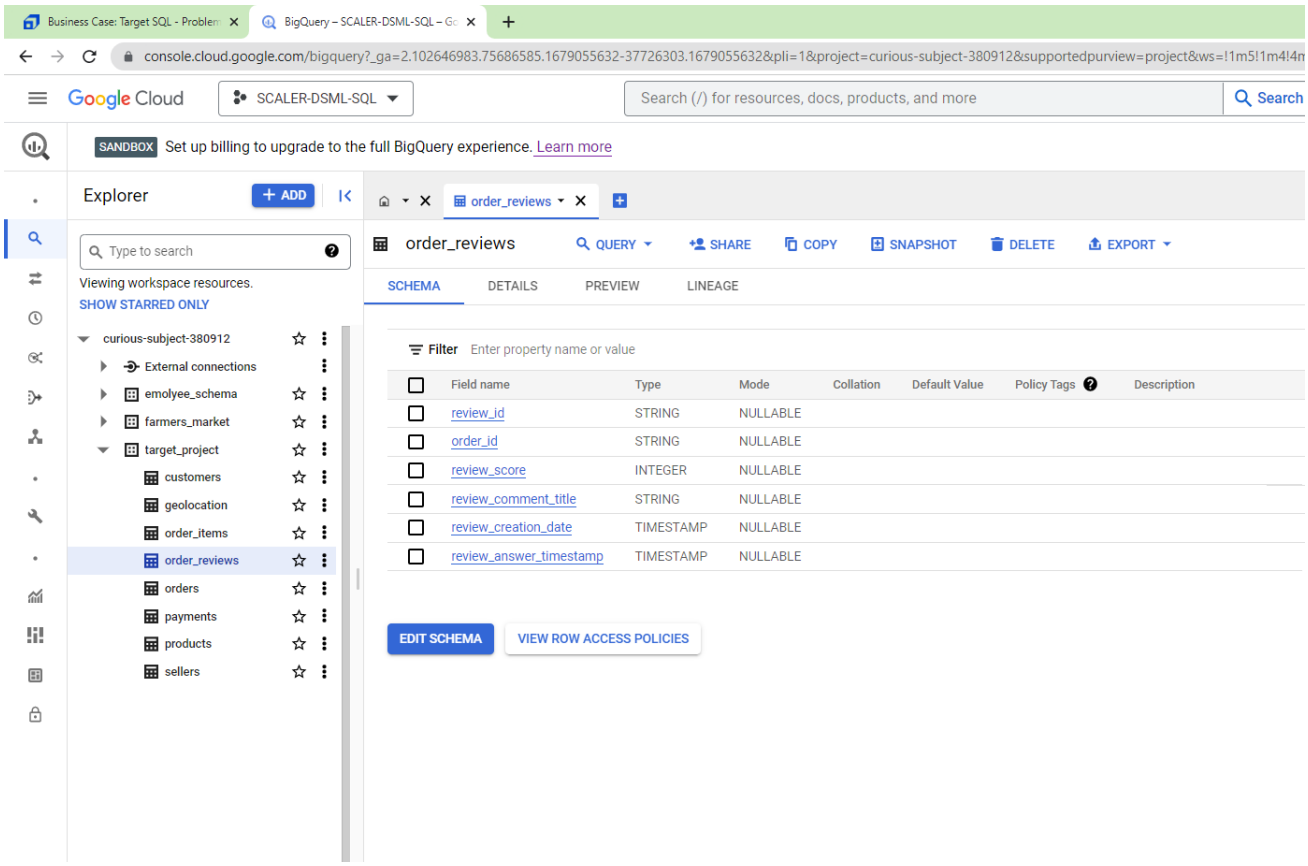
Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset

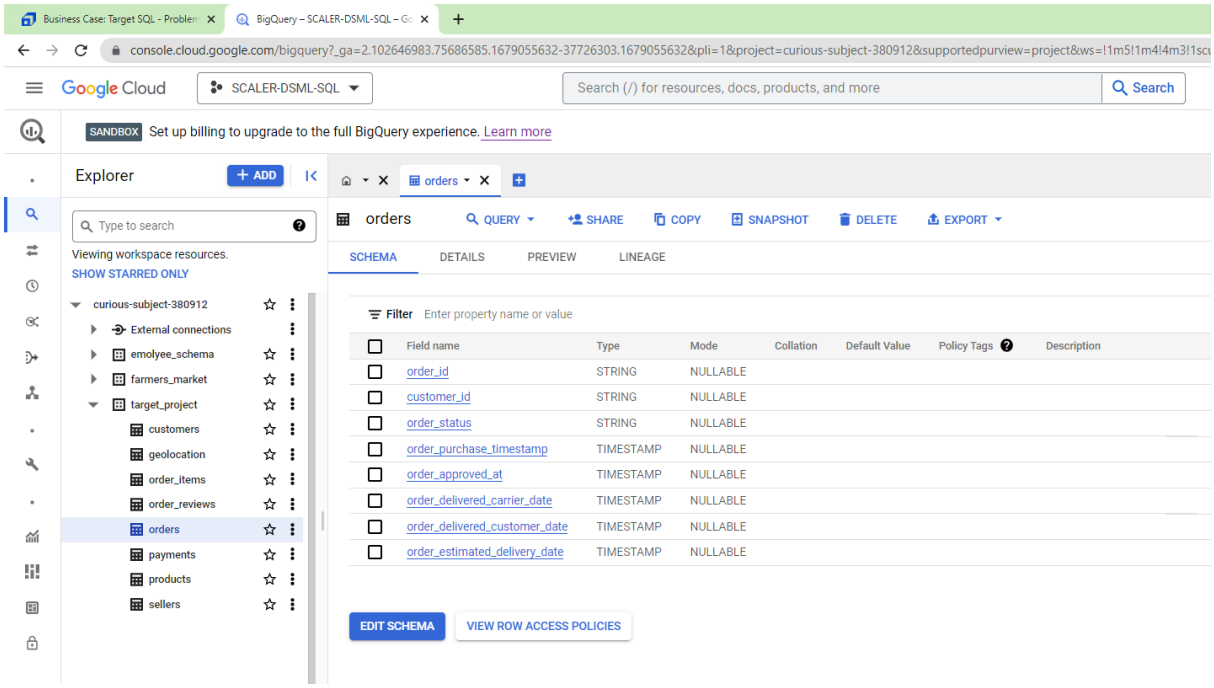
1. Data type of columns in a table

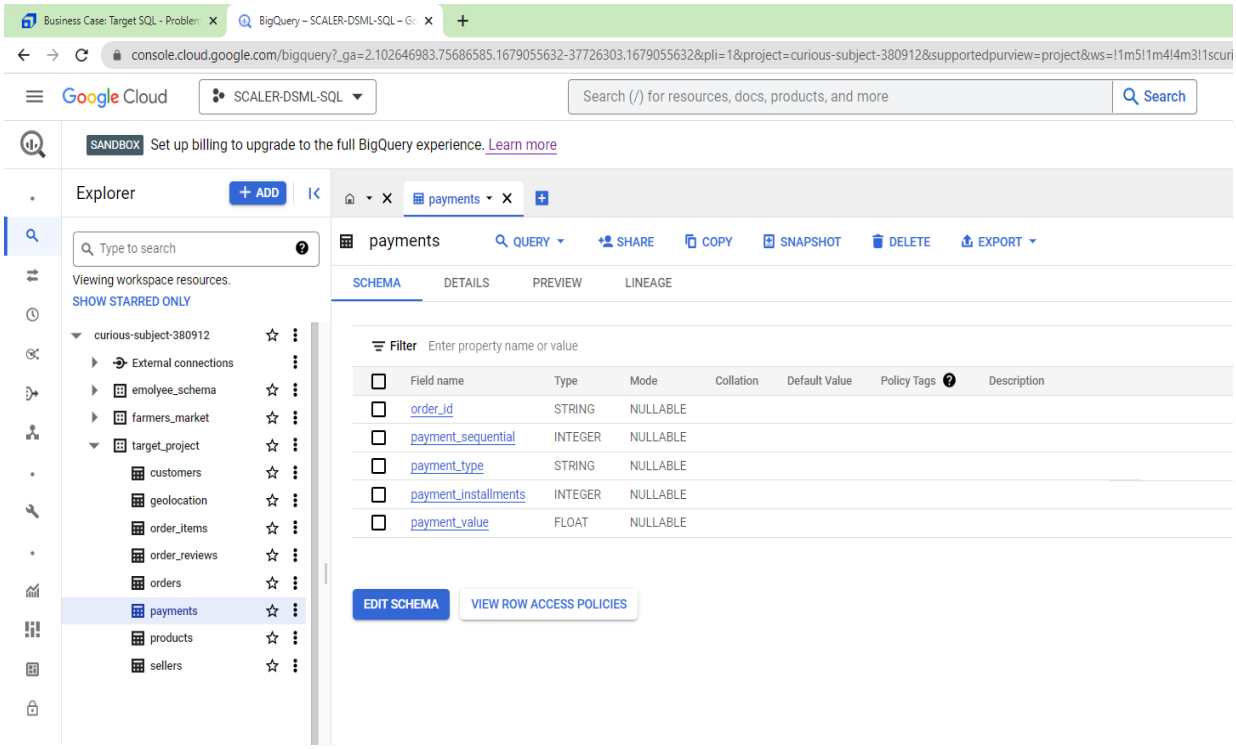


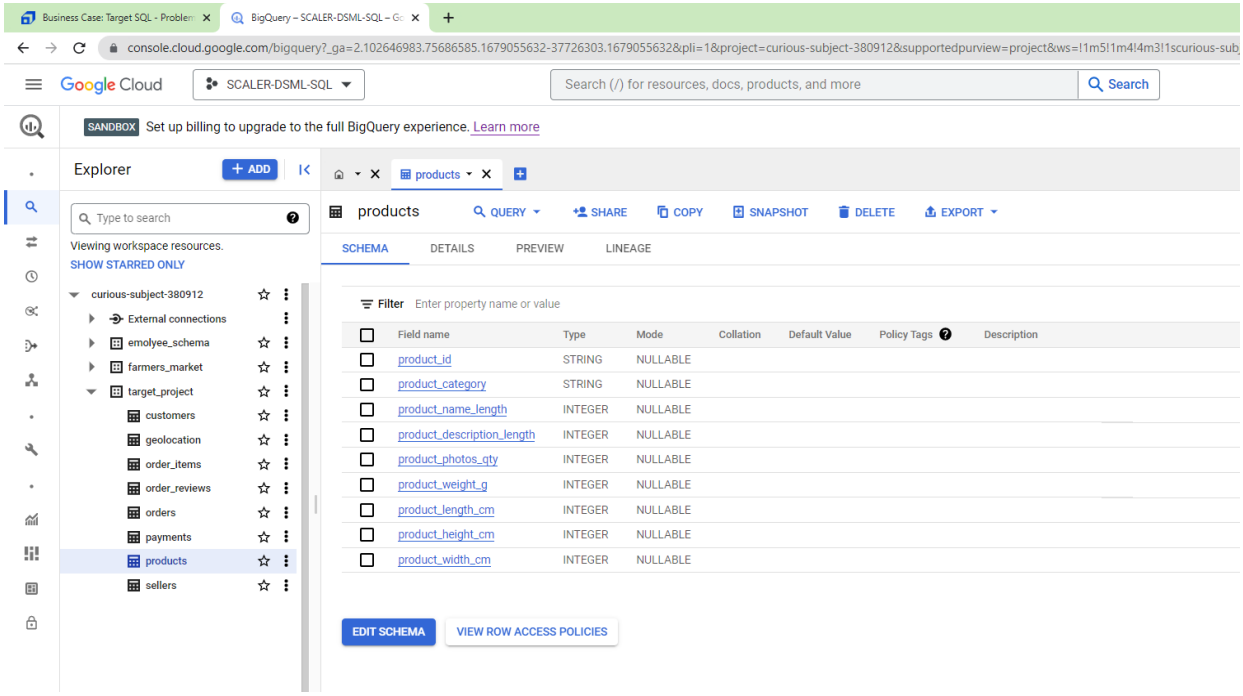


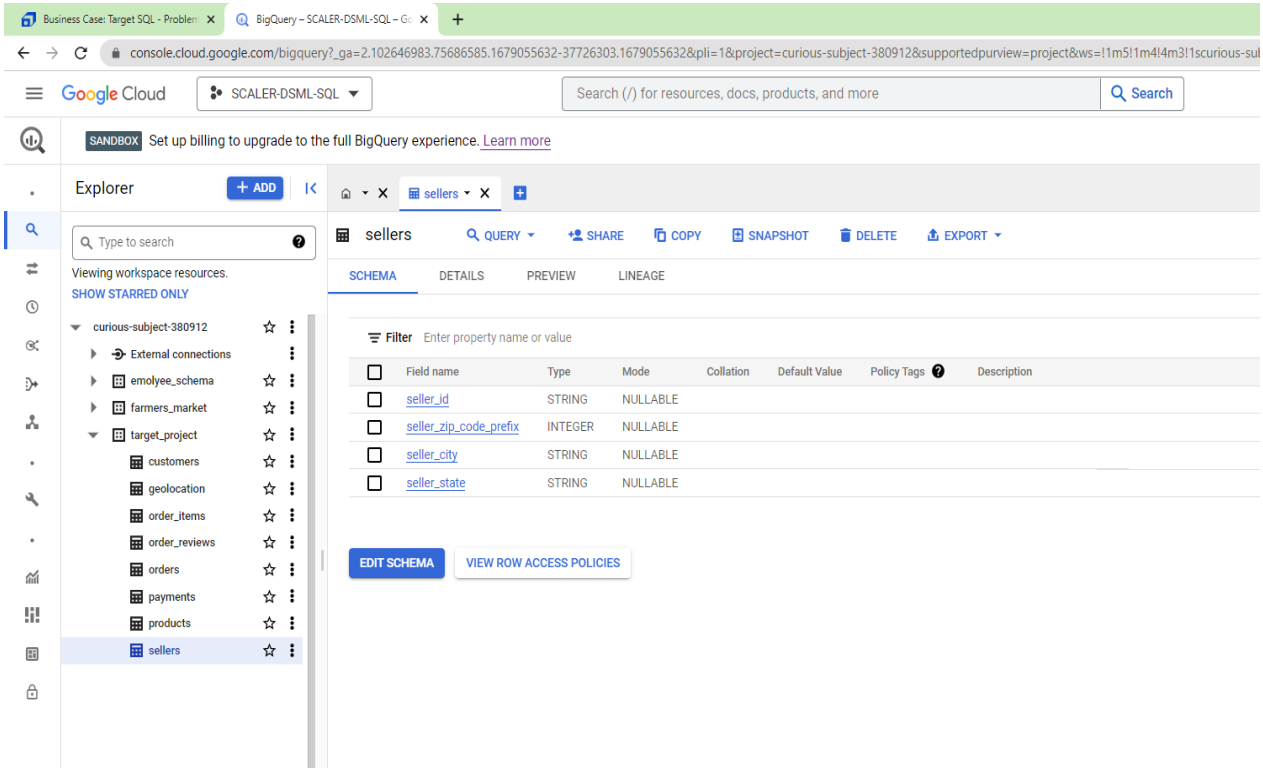

Business Case: Target SQL - Problem
BigQuery – SCALER-DSML-SQL – Go
console.cloud.google.com/bigquery?_ga=2.102646983.75686585.1679055632-37726303.1679055632&pli=1&project=curious-subject-380912&supportedpurview=project&ws=!1m5!1m4!4
Google Cloud
SCALER-DSML-SQL
Search (/) for resources, docs, products, and more
SANDBOX
Set up billing to upgrade to the full BigQuery experience. Learn more
Explorer
ADD
Type to search
Viewing workspace resources.
SHOW STARRED ONLY
curious-subject-380912
External connections
emolyee_schema
farmers_market
target_project
customers
geolocation
order_items
order_reviews
orders
payments
products
sellers
order_items
QUERY
SHARE
COPY
SNAPSHOT
DELETE
EXPORT
SCHEMA
DETAILS
PREVIEW
LINEAGE
Filter
Enter property name or value
Field name
Type
Mode
Collation
Default Value
Policy Tags
Description
order_id
STRING
NULLABLE
order_item_id
INTEGER
NULLABLE
product_id
STRING
NULLABLE
seller_id
STRING
NULLABLE
shipping_limit_date
TIMESTAMP
NULLABLE
price
FLOAT
NULLABLE
freight_value
FLOAT
NULLABLE
EDIT SCHEMA
VIEW ROW ACCESS POLICIES


Business Case: Target SQL - Problem
BigQuery – SCALER-DSML-SQL – Go
console.cloud.google.com/bigquery?_ga=2.102646983.75686585.1679055632-37726303.1679055632&pli=1&project=curious-subject-380912&supportedpurview=project&ws=!1m5!1m4!4m
Google Cloud
SCALER-DSML-SQL
Search (/) for resources, docs, products, and more
Search
SANDBOX
Set up billing to upgrade to the full BigQuery experience. Learn more
Explorer
ADD
Type to search
Viewing workspace resources.
SHOW STARRED ONLY
curious-subject-380912
External connections
emolyee_schema
farmers_market
target_project
customers
geolocation
order_items
order_reviews
orders
payments
products
sellers
order_reviews
QUERY
SHARE
COPY
SNAPSHOT
DELETE
EXPORT
SCHEMA
DETAILS
PREVIEW
LINEAGE
Filter
Enter property name or value
Field name
Type
Mode
Collation
Default Value
Policy Tags
Description
review_id
STRING
NULLABLE
order_id
STRING
NULLABLE
review_score
INTEGER
NULLABLE
review_comment_title
STRING
NULLABLE
review_creation_date
TIMESTAMP
NULLABLE
review_answer_timestamp
TIMESTAMP
NULLABLE
EDIT SCHEMA
VIEW ROW ACCESS POLICIES


Business Case: Target SQL - Problem
BigQuery – SCALER-DSML-SQL – Go
console.cloud.google.com/bigquery?_ga=2.102646983.75686585.1679055632-37726303.1679055632&pli=1&project=curious-subject-380912&supportedpurview=project&ws=!1m5!1m4!4m3!1sc
Google Cloud
SCALER-DSML-SQL
Search (/) for resources, docs, products, and more
Search
SANDBOX
Set up billing to upgrade to the full BigQuery experience. Learn more
Explorer
ADD
orders
Type to search
Viewing workspace resources.
SHOW STARRED ONLY
curious-subject-380912
External connections
emolyee_schema
farmers_market
target_project
customers
geolocation
order_items
order_reviews
orders
payments
products
sellers
QUERY
SHARE
COPY
SNAPSHOT
DELETE
EXPORT
SCHEMA
DETAILS
PREVIEW
LINEAGE
Filter
Enter property name or value
Field name
Type
Mode
Collation
Default Value
Policy Tags
Description
order_id STRING NULLABLE
customer_id STRING NULLABLE
order_status STRING NULLABLE
order_purchase_timestamp TIMESTAMP NULLABLE
order_approved_at TIMESTAMP NULLABLE
order_delivered_carrier_date TIMESTAMP NULLABLE
order_delivered_customer_date TIMESTAMP NULLABLE
order_estimated_delivery_date TIMESTAMP NULLABLE
EDIT SCHEMA
VIEW ROW ACCESS POLICIES

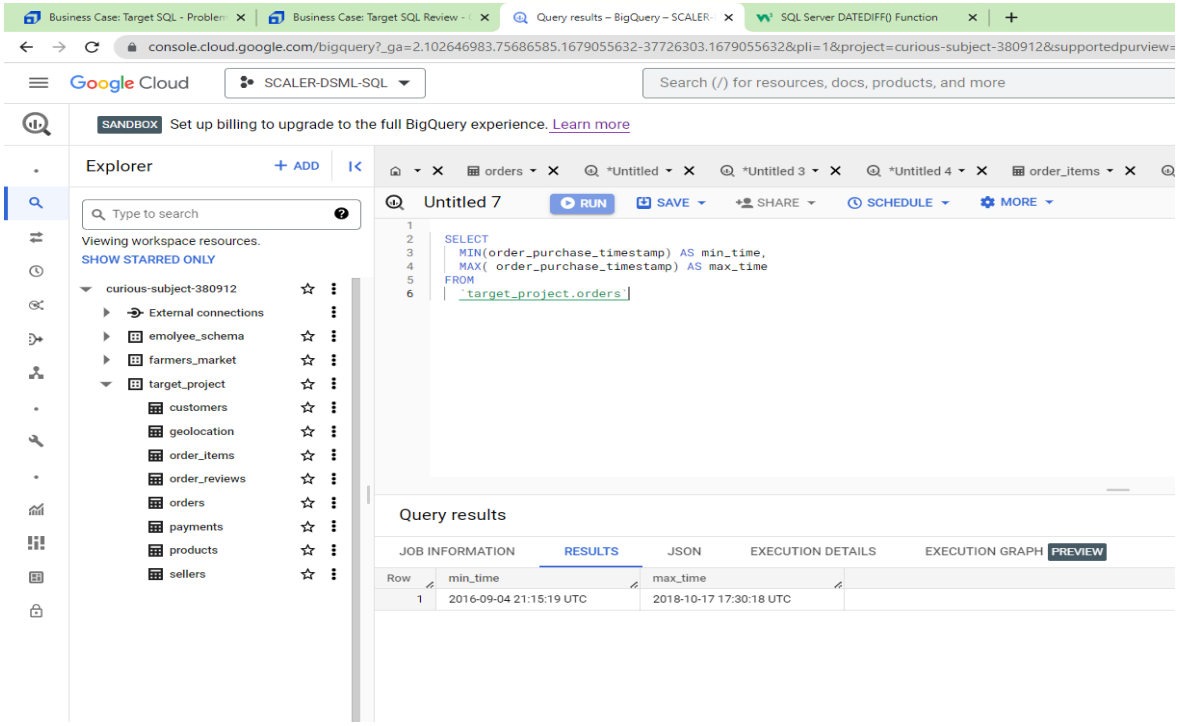

Business Case: Target SQL - Problem
BigQuery – SCALER-DSML-SQL – Go
console.cloud.google.com/bigquery?_ga=2.102646983.75686585.1679055632-37726303.1679055632&pli=1&project=curious-subject-380912&supportedpurview=project&ws=!1m5!1m4!4m3!1scuri
Google Cloud
SCALER-DSML-SQL
Search (/) for resources, docs, products, and more
Search
SANDBOX
Set up billing to upgrade to the full BigQuery experience. Learn more
Explorer
ADD
payments
Type to search
Viewing workspace resources.
SHOW STARRED ONLY
curious-subject-380912
External connections
emolyee_schema
farmers_market
target_project
customers
geolocation
order_items
order_reviews
orders
payments
products
sellers
QUERY
SHARE
COPY
SNAPSHOT
DELETE
EXPORT
SCHEMA
DETAILS
PREVIEW
LINEAGE
Filter
Enter property name or value
Field name
Type
Mode
Collation
Default Value
Policy Tags
Description
order_id STRING NULLABLE
payment_sequential INTEGER NULLABLE
payment_type STRING NULLABLE
payment_installments INTEGER NULLABLE
payment_value FLOAT NULLABLE
EDIT SCHEMA
VIEW ROW ACCESS POLICIES


Business Case: Target SQL - Problem
BigQuery – SCALER-DSML-SQL – Go
console.cloud.google.com/bigquery?_ga=2.102646983.75686585.1679055632-37726303.1679055632&pli=1&project=curious-subject-380912&supportedpurview=project&ws=!1m5!1m4!4m3!1scurious-sub
Google Cloud
SCALER-DSML-SQL
Search (/) for resources, docs, products, and more
Search
SANDBOX
Set up billing to upgrade to the full BigQuery experience. Learn more
Explorer
ADD
Type to search
Viewing workspace resources.
SHOW STARRED ONLY
curious-subject-380912
External connections
emolyee_schema
farmers_market
target_project
customers
geolocation
order_items
order_reviews
orders
payments
products
sellers
products
QUERY
SHARE
COPY
SNAPSHOT
DELETE
EXPORT
SCHEMA
DETAILS
PREVIEW
LINEAGE
Filter
Enter property name or value
Field name
Type
Mode
Collation
Default Value
Policy Tags
Description
product_id STRING NULLABLE
product_category STRING NULLABLE
product_name_length INTEGER NULLABLE
product_description_length INTEGER NULLABLE
product_photos_qty INTEGER NULLABLE
product_weight_g INTEGER NULLABLE
product_length_cm INTEGER NULLABLE
product_height_cm INTEGER NULLABLE
product_width_cm INTEGER NULLABLE
EDIT SCHEMA
VIEW ROW ACCESS POLICIES


Business Case: Target SQL - Problem
BigQuery – SCALER-DSML-SQL – Go
console.cloud.google.com/bigquery?_ga=2.102646983.75686585.1679055632-37726303.1679055632&pli=1&project=curious-subject-380912&supportedpurview=project&ws=!1m5!1m4!4m3!1scurious-sul
Google Cloud
SCALER-DSML-SQL
Search (/) for resources, docs, products, and more
Search
SANDBOX
Set up billing to upgrade to the full BigQuery experience. Learn more
Explorer
ADD
Type to search
Viewing workspace resources.
SHOW STARRED ONLY
curious-subject-380912
External connections
emolyee_schema
farmers_market
target_project
customers
geolocation
order_items
order_reviews
orders
payments
products
sellers
sellers
QUERY
SHARE
COPY
SNAPSHOT
DELETE
EXPORT
SCHEMA
DETAILS
PREVIEW
LINEAGE
Filter
Enter property name or value
Field name
Type
Mode
Collation
Default Value
Policy Tags
Description
seller_id STRING NULLABLE
seller_zip_code_prefix INTEGER NULLABLE
seller_city STRING NULLABLE
seller_state STRING NULLABLE
EDIT SCHEMA
VIEW ROW ACCESS POLICIES

2. Time period for which the data is given

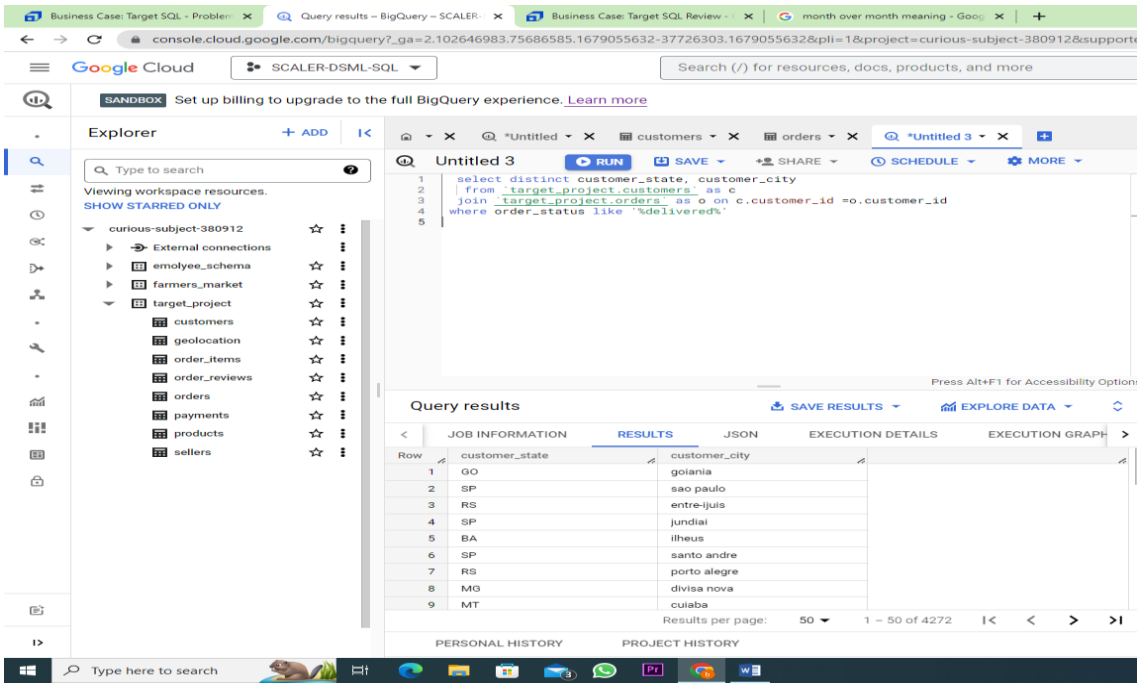
```
SELECT
  MIN(order_purchase_timestamp) AS min_time,
  MAX( order_purchase_timestamp) AS max_time
FROM
  `target_project.orders`
```



Data is given from 2016 to 2018

3. Cities and States of customers ordered during the given period

```sql
  select distinct customer_state, customer_city
   from `target_project.customers` as c
  join `target_project.orders` as o on c.customer_id =o.customer_id
 where order_status like '%delivered%'

```

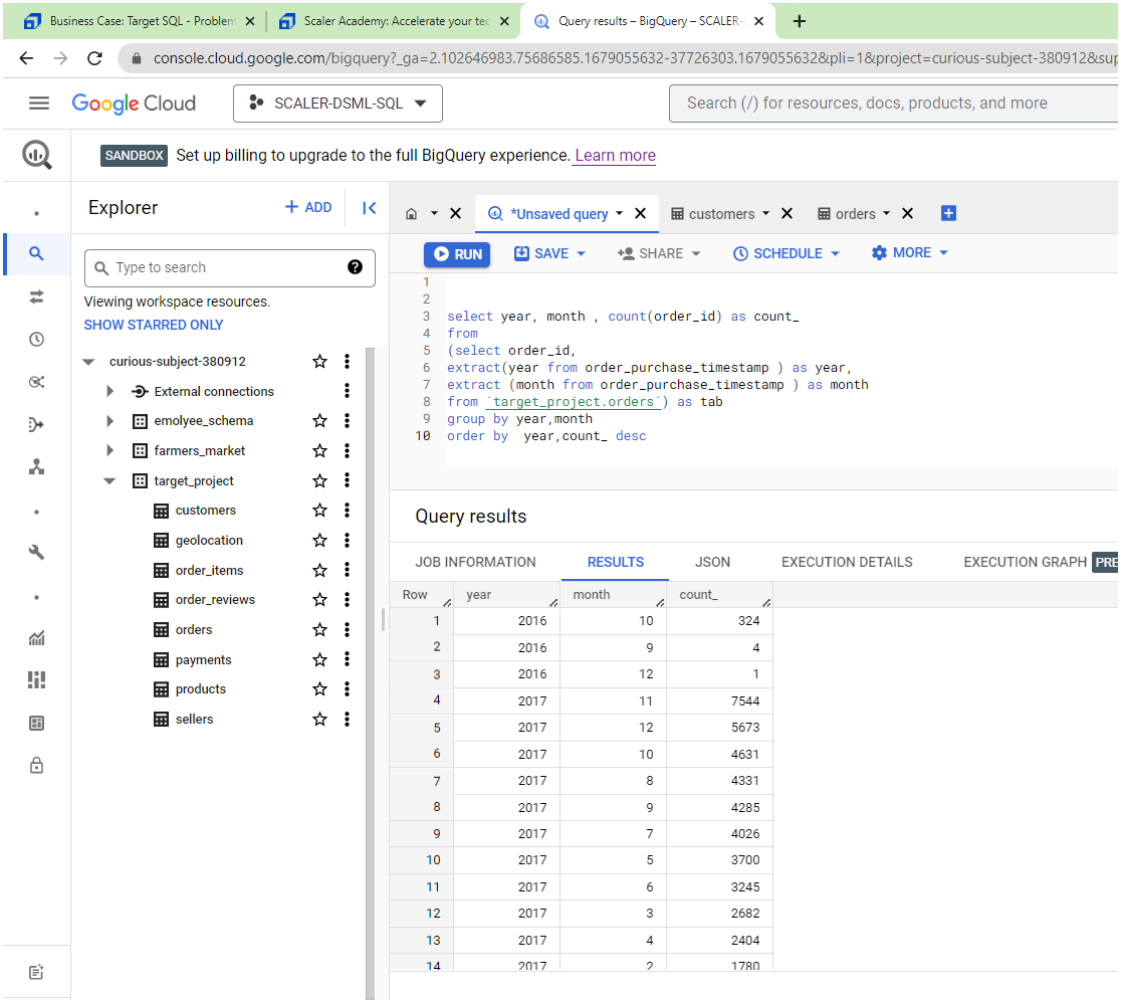


Data about city and state are obtained from customers table and to obtain customer_ids that are ordered , inner joining with orders table made and in the order status delivered are only included

2. In-depth Exploration:

1.Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario? Can we see some seasonality with peaks at specific months

```
select year, month , count(order_id) as count_
from
(select order_id,
extract(year from order_purchase_timestamp ) as year,
extract (month from order_purchase_timestamp ) as month
from `target_project.orders`) as tab
group by year,month
order by  year,count_ desc
```



by referring increased order_id count as increased sales , from data it is concluded there Is a growing trend on e-commerce in Brazil from 2016 to 2018 .

in 2016 , orders are high during 1oth month,followed by decline in orders in 12th month.

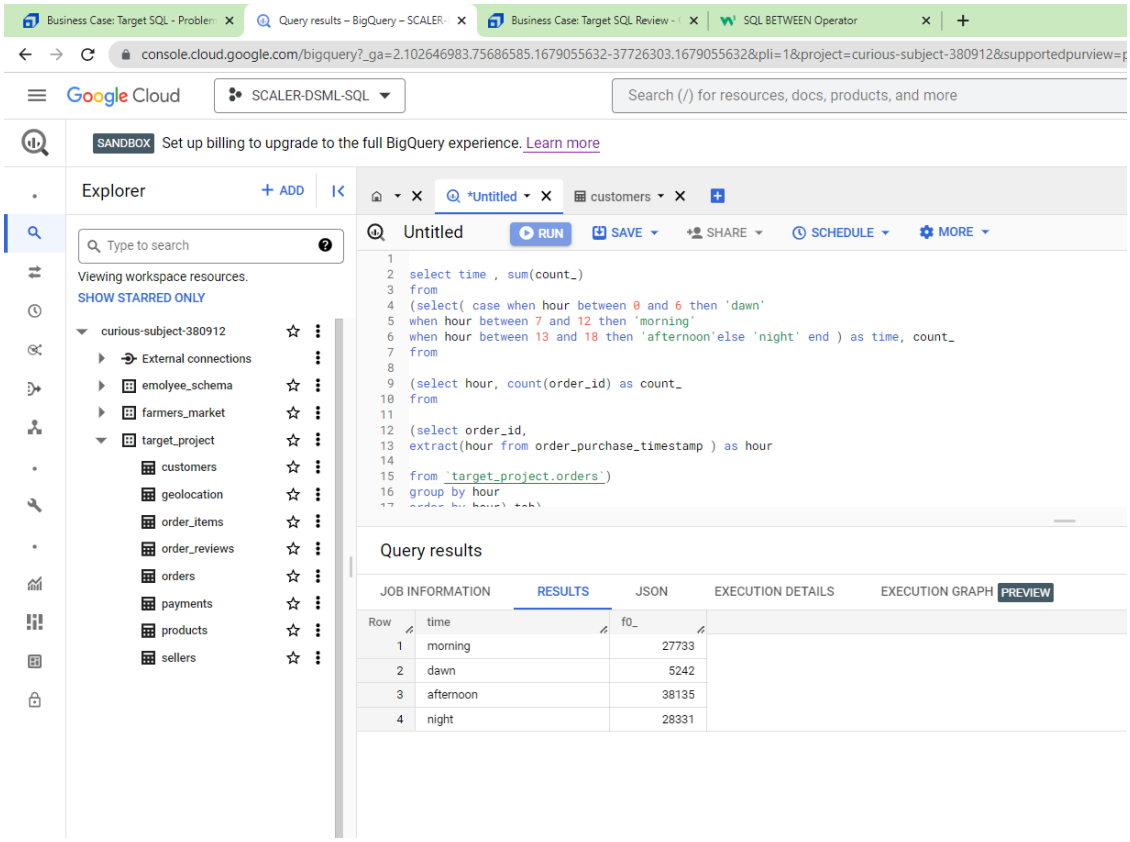
in 2017 , orders gradually increased from 1st month to last month , peaked at 11th month followed by 12th month

in 2018 , peak orders are observed in 1st month , sustainable orders are observed upto 8th month followed by rapid decline of order count .

from 2017 to 2018 , high amount of orders are seen during winter times .

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

```
select time , sum(count_)
from
(select( case when hour between 0 and 6 then 'dawn'
when hour between 7 and 12 then 'morning'
when hour between 13 and 18 then 'afternoon'else 'night' end ) as time, count_
from

(select hour, count(order_id) as count_
from

(select order_id,
extract(hour from order_purchase_timestamp ) as hour

from `target_project.orders`)
group by hour
order by hour) tab)
group by time
```



Trend suggest that , afternoon orders are high followed by night orders and morning orders and dawn orders.

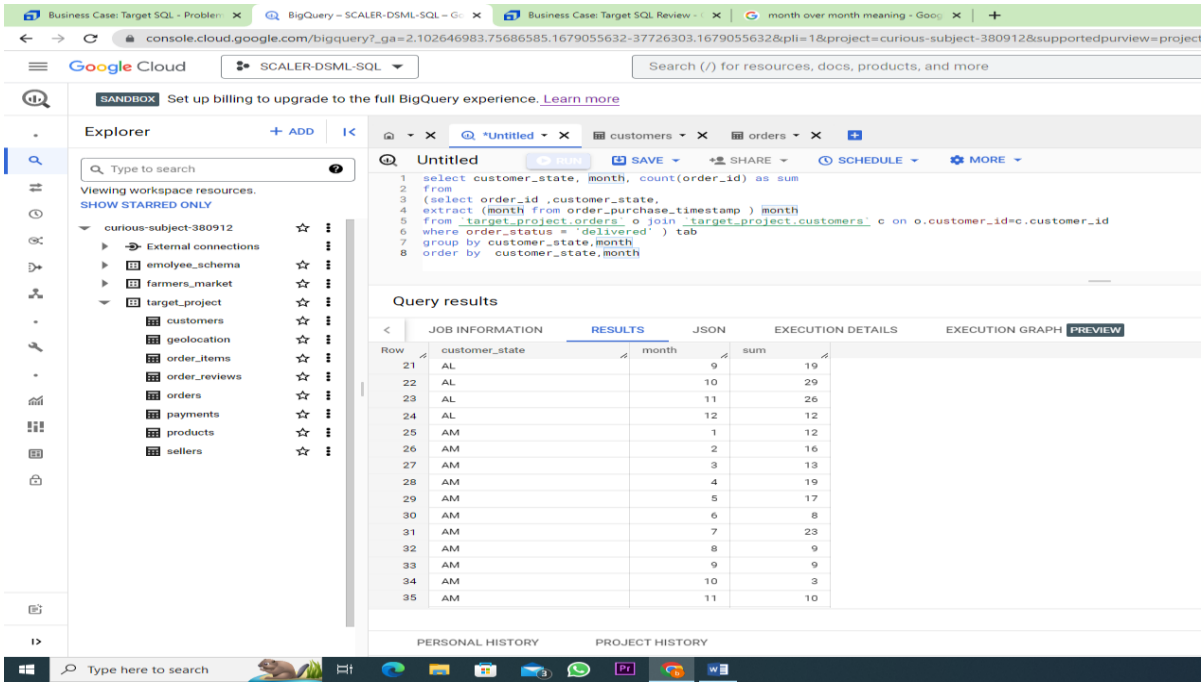
3.Evolution of E-commerce orders in the Brazil region:

1. Get month on month orders by states

```
select customer_state, month, count(order_id) as sum
from
(select order_id ,customer_state,
extract (month from order_purchase_timestamp ) month
from `target_project.orders` o join `target_project.customers` c on o.customer_id=c
.customer_id
where order_status = 'delivered' ) tab
group by customer_state,month
order by  customer_state,month
```

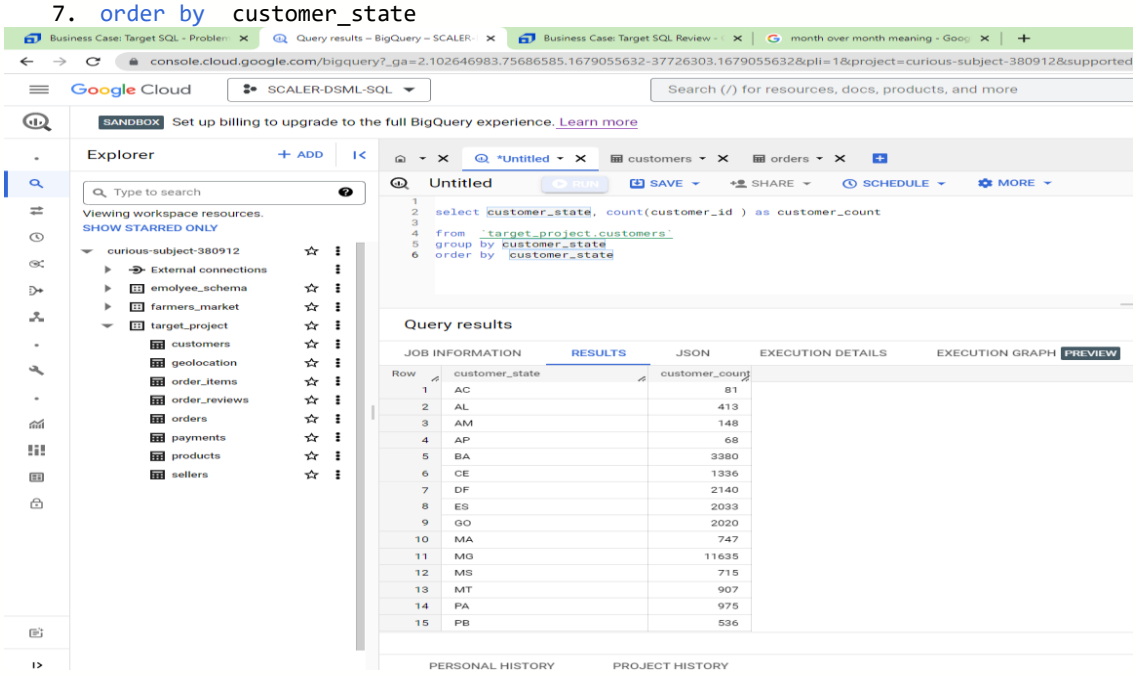
profound amounts of orders are seen in 5^{th} month from a state ‘SP’ in brazil

Delivered orders are taken into consideration . group by and order by is performed on both cuatomer_state and month to know, in each state for each month number of orders delivered .



2. Distribution of customers across the states in Brazil

```
3. select customer_state, count(customer_id ) as customer_count
4.
5. from  `target_project.customers`
6. group by customer_state
7. order by  customer_state
```

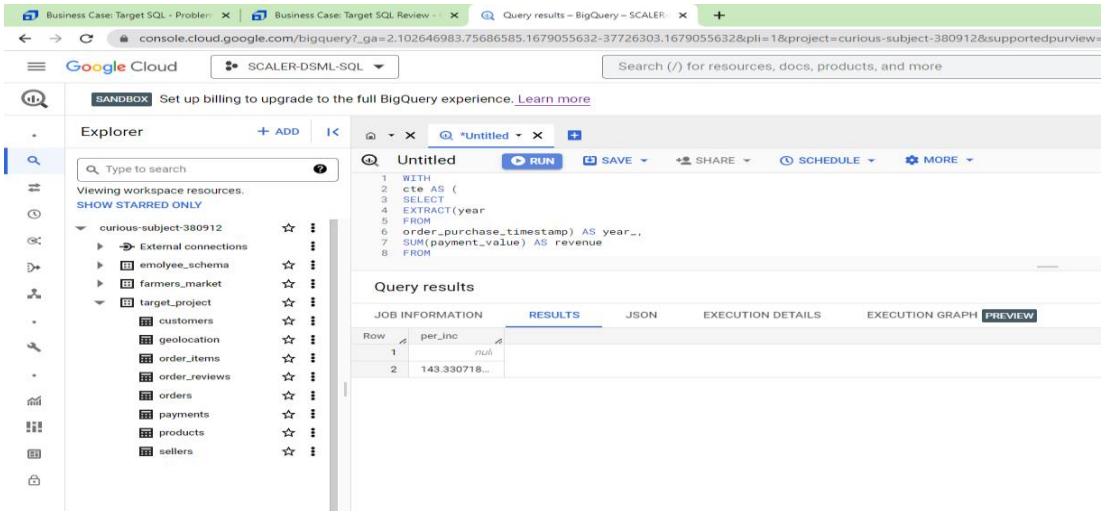


Upon ordering by customer_count it is noticed that, Huge number of customers are from a state called ‘SP’ WITH WHOOPING 41746 CUSTOMERS

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

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

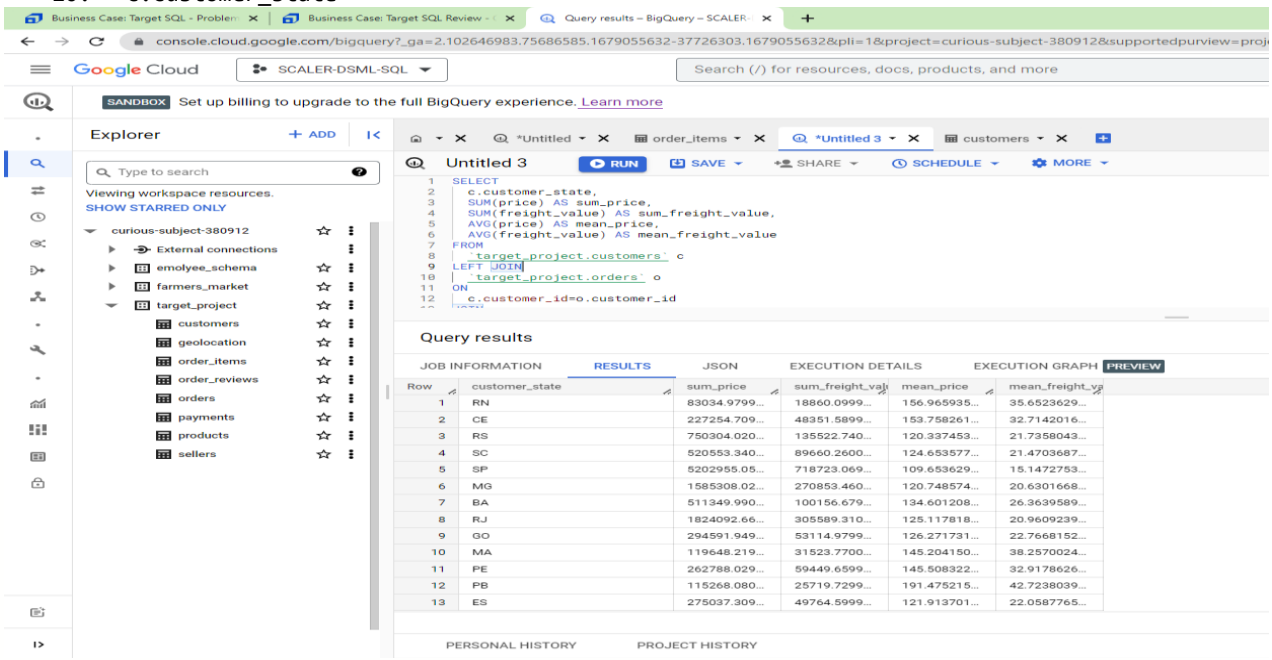
```
WITH
cte AS (
SELECT
EXTRACT(year
FROM
order_purchase_timestamp) AS year_,
SUM(payment_value) AS revenue
FROM
`target_project.orders` AS o
INNER JOIN
`target_project.payments` AS p
ON
p.order_id=o.order_id
WHERE
order_status='delivered'
AND EXTRACT(month
FROM
order_purchase_timestamp) BETWEEN 0
AND 8
GROUP BY
year_),
lastRev AS (
SELECT
*,
LAG(revenue) OVER(ORDER BY year_ ASC) AS prev_rev
FROM
cte)
SELECT
(revenue-prev_rev)/prev_rev*100 AS per_inc
FROM
lastRev
```



143 % growth of revenue is observed

2. Mean & Sum of price and freight value by customer state

```sql
SELECT
  c.customer_state,
  SUM(price) AS sum_price,
  SUM(freight_value) AS sum_freight_value,
  AVG(price) AS mean_price,
  AVG(freight_value) AS mean_freight_value
FROM
  `target_project.customers` c
LEFT JOIN
  `target_project.orders` o
ON
  c.customer_id=o.customer_id
JOIN
  `target_project.order_items` AS ot
ON
  o.order_id=ot.order_id
GROUP BY
  c.customer_state
```

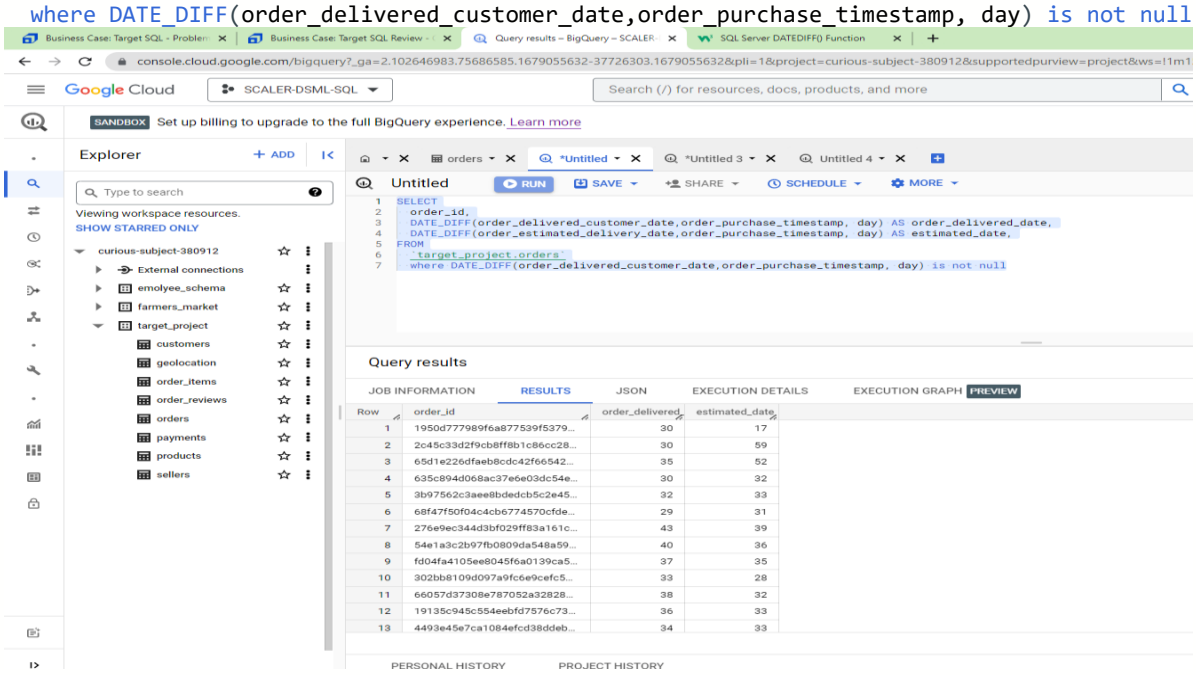


sum and mean are performed in select clause and as they are asked according to customer state multiple joins are performed followed by customer state grouping

5. Analysis on sales, freight and delivery time

5.1 Calculate days between purchasing, delivering and estimated delivery

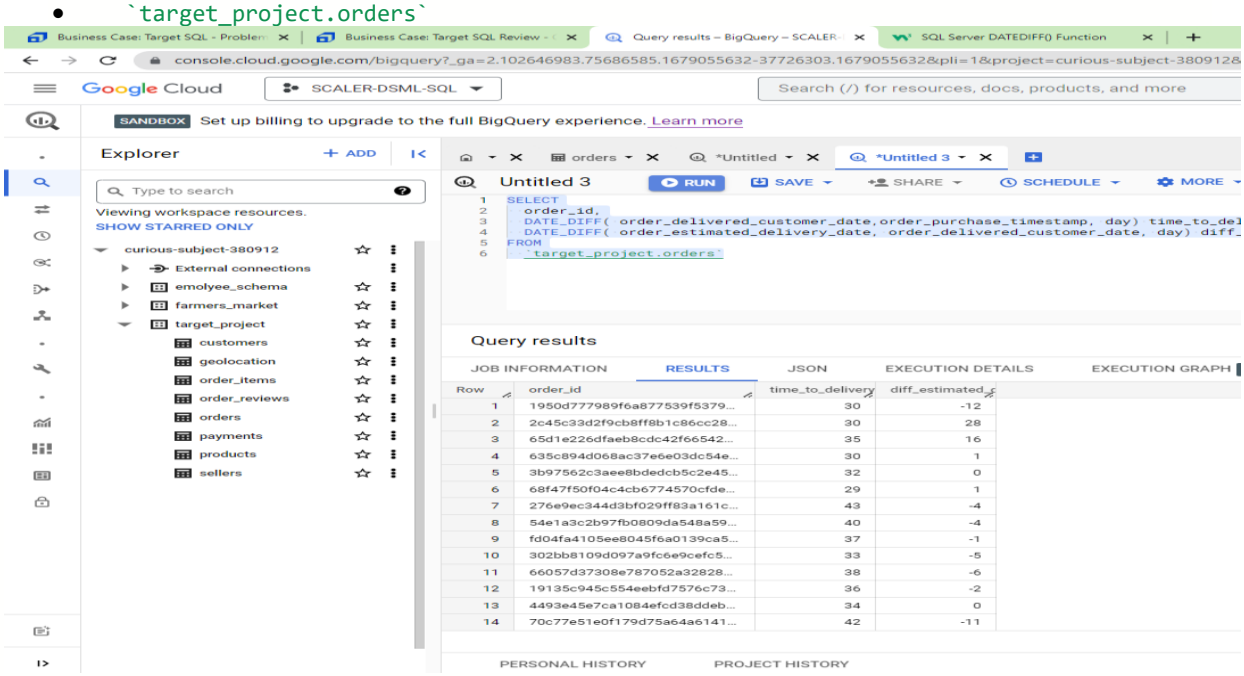
```
SELECT
  order_id,
  DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, day) AS order_delivered
_date,
  DATE_DIFF(order_estimated_delivery_date,order_purchase_timestamp, day) AS estimated_date,
FROM
  `target_project.orders`
  where DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, day) is not null
```



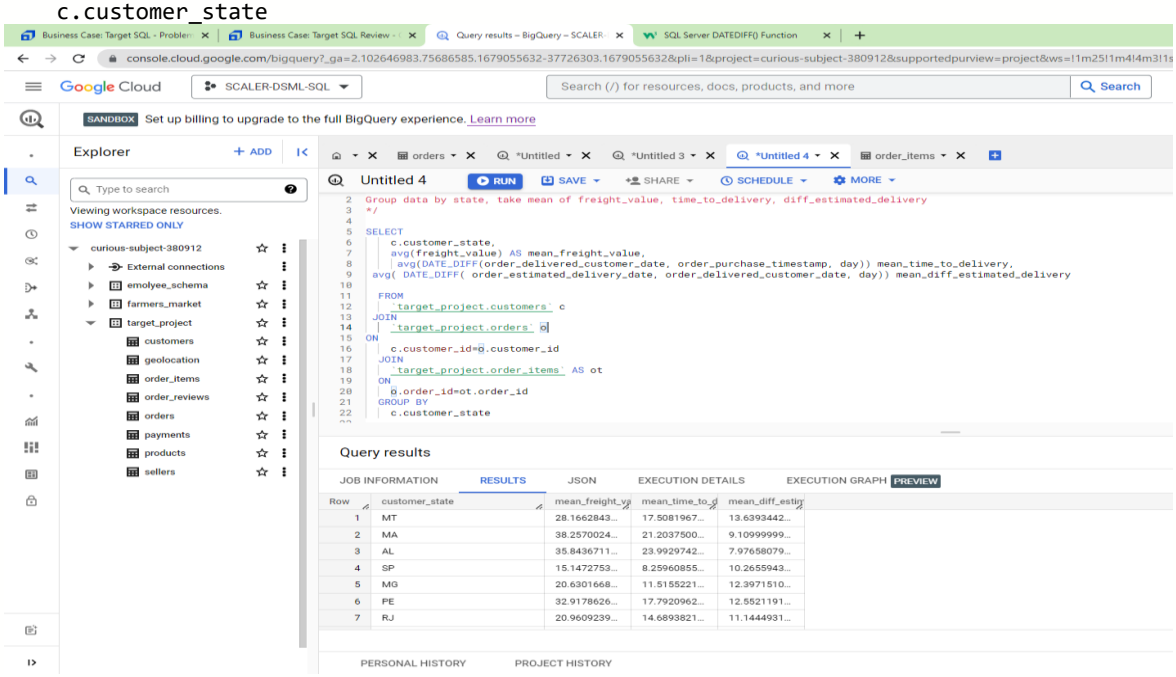
5.2.

- time_to_delivery = order_purchase_timestamp-order_delivered_customer_date
- diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date

```
SELECT
  order_id,
  DATE_DIFF( order_delivered_customer_date,order_purchase_timestamp, day) time_to_delivery,
  DATE_DIFF( order_estimated_delivery_date, order_delivered_customer_date, day) diff_estimated_delivery
FROM
  `target_project.orders`
```



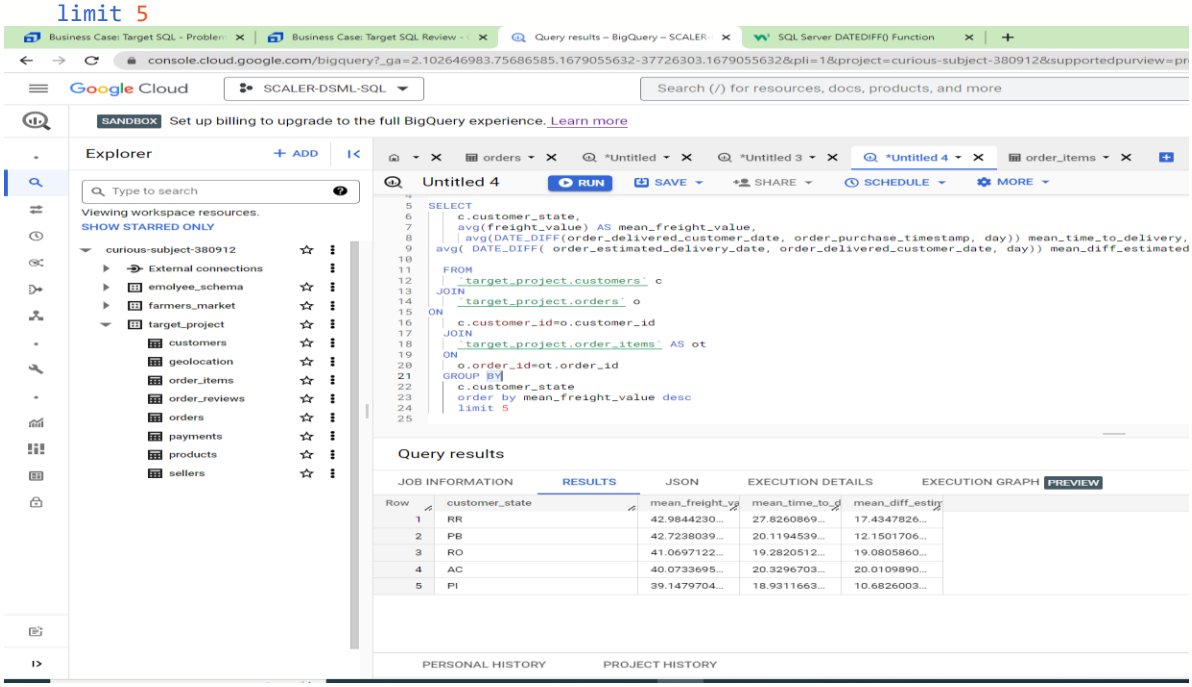
```sql
/*
Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery
*/

SELECT
    c.customer_state,
    avg(freight_value) AS mean_freight_value,
     avg(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day)) mean_time
_to_delivery,
 avg( DATE_DIFF( order_estimated_delivery_date, order_delivered_customer_date, day)) mean_d
iff_estimated_delivery

  FROM
    `target_project.customers` c
 JOIN
    `target_project.orders` o
ON
    c.customer_id=o.customer_id
  JOIN
    `target_project.order_items` AS ot
  ON
    o.order_id=ot.order_id
  GROUP BY
    c.customer_state
```



5.5. Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

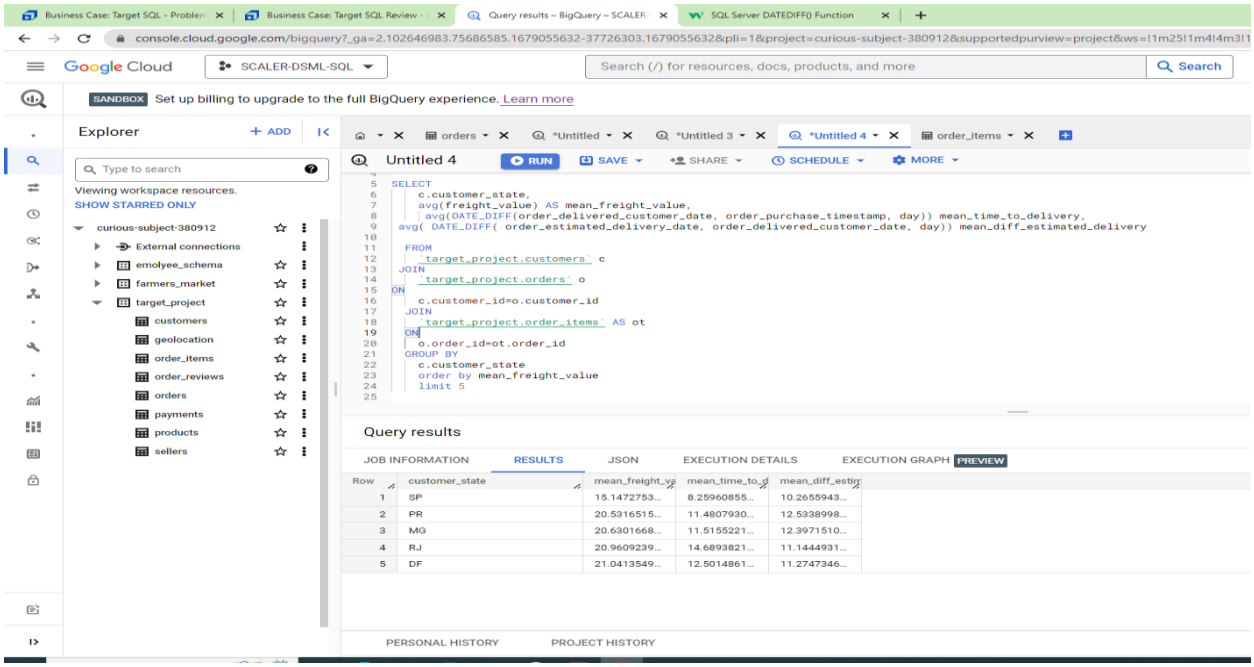
(Top 5 states with highest average freight value)

```
SELECT
    c.customer_state,
    avg(freight_value) AS mean_freight_value,
     avg(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day)) mean_time
_to_delivery,
 avg( DATE_DIFF( order_estimated_delivery_date, order_delivered_customer_date, day)) mean_d
iff_estimated_delivery

  FROM
    `target_project.customers` c
 JOIN
    `target_project.orders` o
ON
    c.customer_id=o.customer_id
  JOIN
    `target_project.order_items` AS ot
  ON
    o.order_id=ot.order_id
  GROUP BY
    c.customer_state
    order by mean_freight_value desc
    limit 5
```



(Top 5 states with lowest average freight value)

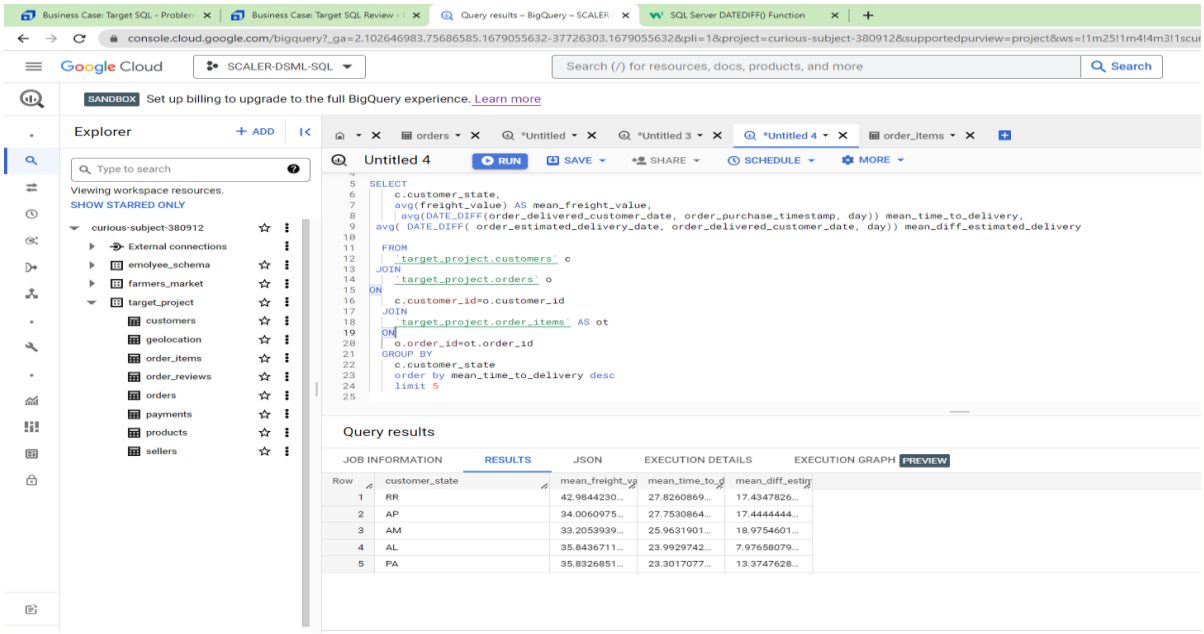
```
SELECT
    c.customer_state,
    avg(freight_value) AS mean_freight_value,
     avg(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day)) mean_time
_to_delivery,
 avg( DATE_DIFF( order_estimated_delivery_date, order_delivered_customer_date, day)) mean_d
iff_estimated_delivery

  FROM
    `target_project.customers` c
 JOIN
    `target_project.orders` o
ON
    c.customer_id=o.customer_id
  JOIN
    `target_project.order_items` AS ot
  ON
    o.order_id=ot.order_id
  GROUP BY
    c.customer_state
    order by mean_freight_value
    limit 5
```



5.6. Top 5 states with highest/lowest average time to delivery

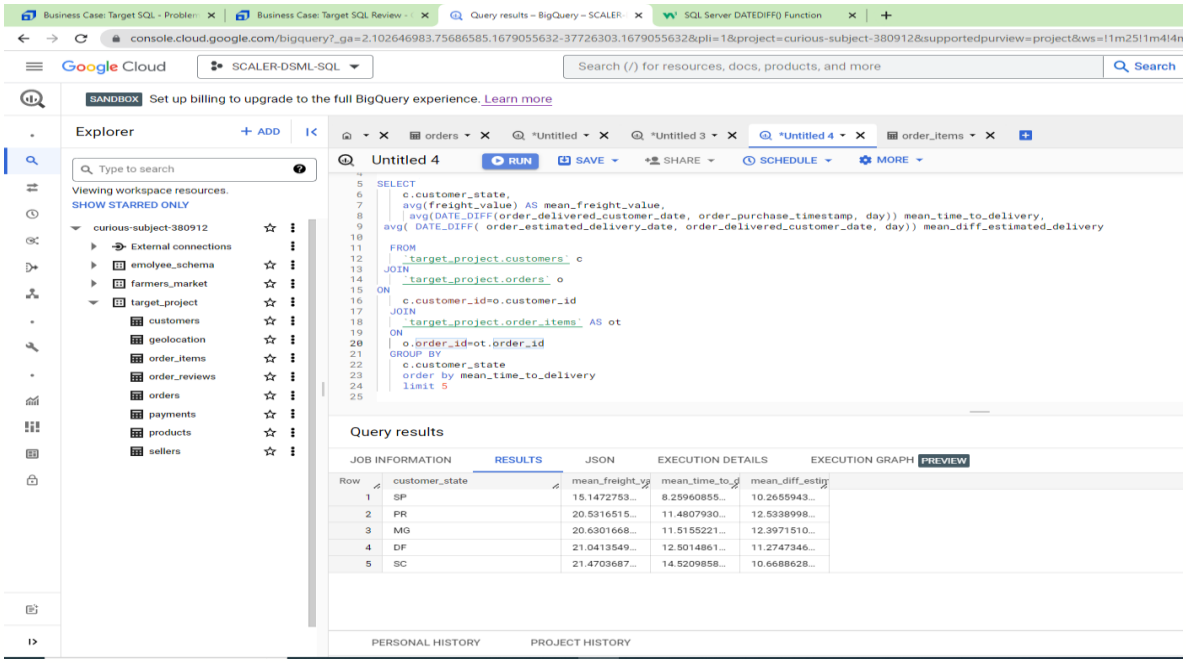
(Top 5 states with highest average time to delivery)

```
SELECT
    c.customer_state,
    avg(freight_value) AS mean_freight_value,
     avg(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day)) mean_time
_to_delivery,
 avg( DATE_DIFF( order_estimated_delivery_date, order_delivered_customer_date, day)) mean_d
iff_estimated_delivery

  FROM
    `target_project.customers` c
 JOIN
    `target_project.orders` o
ON
    c.customer_id=o.customer_id
  JOIN
    `target_project.order_items` AS ot
  ON
    o.order_id=ot.order_id
  GROUP BY
    c.customer_state
    order by mean_time_to_delivery desc
    limit 5
```



(Top 5 states with lowest average time to delivery)

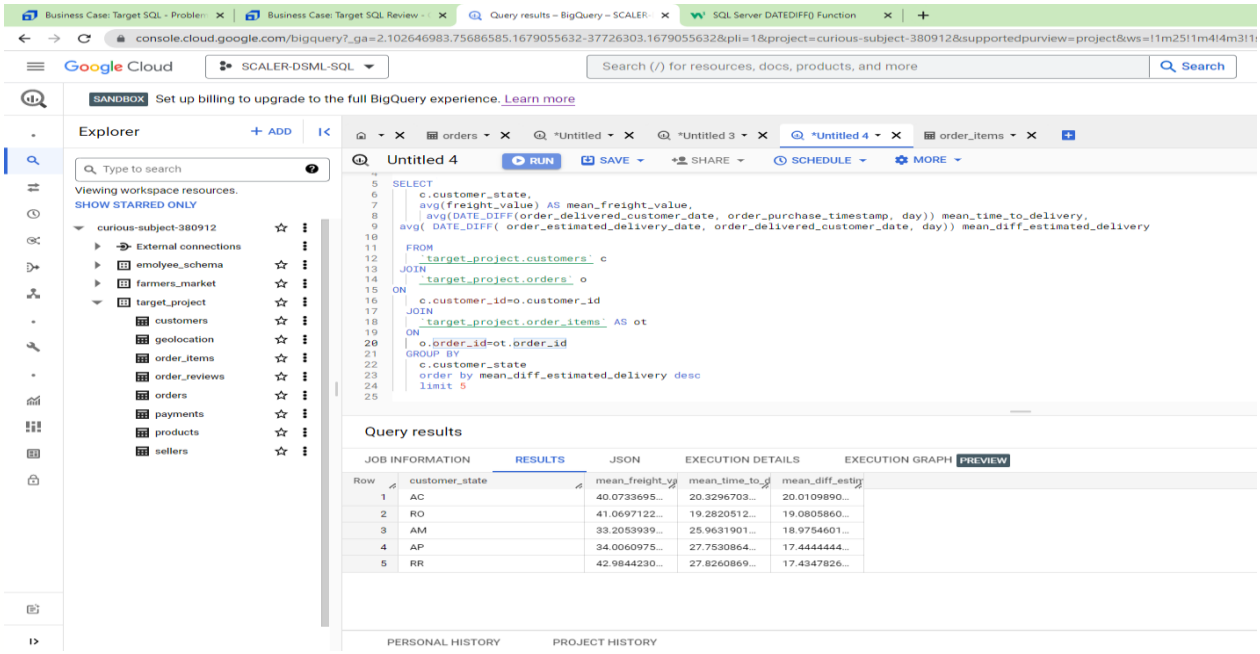
```
SELECT
    c.customer_state,
    avg(freight_value) AS mean_freight_value,
     avg(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day)) mean_time
_to_delivery,
 avg( DATE_DIFF( order_estimated_delivery_date, order_delivered_customer_date, day)) mean_d
iff_estimated_delivery

  FROM
    `target_project.customers` c
 JOIN
    `target_project.orders` o
ON
    c.customer_id=o.customer_id
  JOIN
    `target_project.order_items` AS ot
  ON
    o.order_id=ot.order_id
  GROUP BY
    c.customer_state
    order by mean_time_to_delivery
    limit 5
```



5.7.Top 5 states where delivery is really fast/ not so fast compared to estimated date

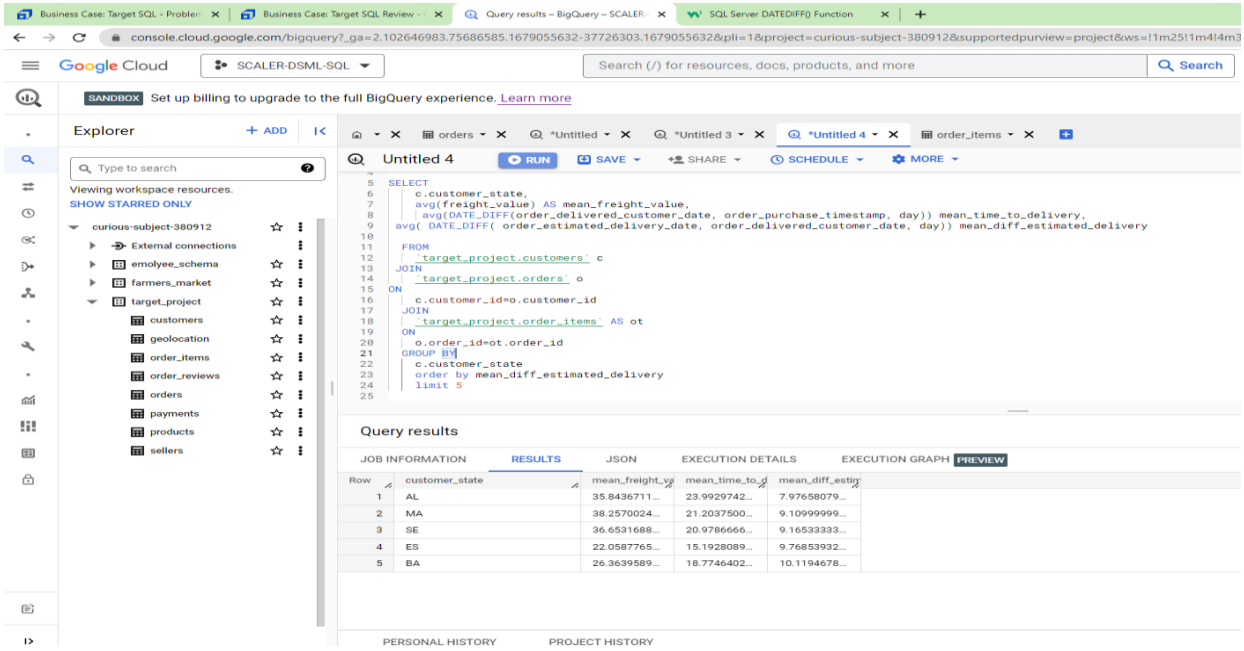
Top 5 states where delivery is really not so fast compared to estimated date

```
SELECT
    c.customer_state,
    avg(freight_value) AS mean_freight_value,
     avg(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day)) mean_time
_to_delivery,
 avg( DATE_DIFF( order_estimated_delivery_date, order_delivered_customer_date, day)) mean_d
iff_estimated_delivery

  FROM
    `target_project.customers` c
 JOIN
    `target_project.orders` o
ON
    c.customer_id=o.customer_id
  JOIN
    `target_project.order_items` AS ot
  ON
    o.order_id=ot.order_id
  GROUP BY
    c.customer_state
    order by mean_diff_estimated_delivery desc
    limit 5
```



Top 5 states where delivery is really fast compared to estimated date

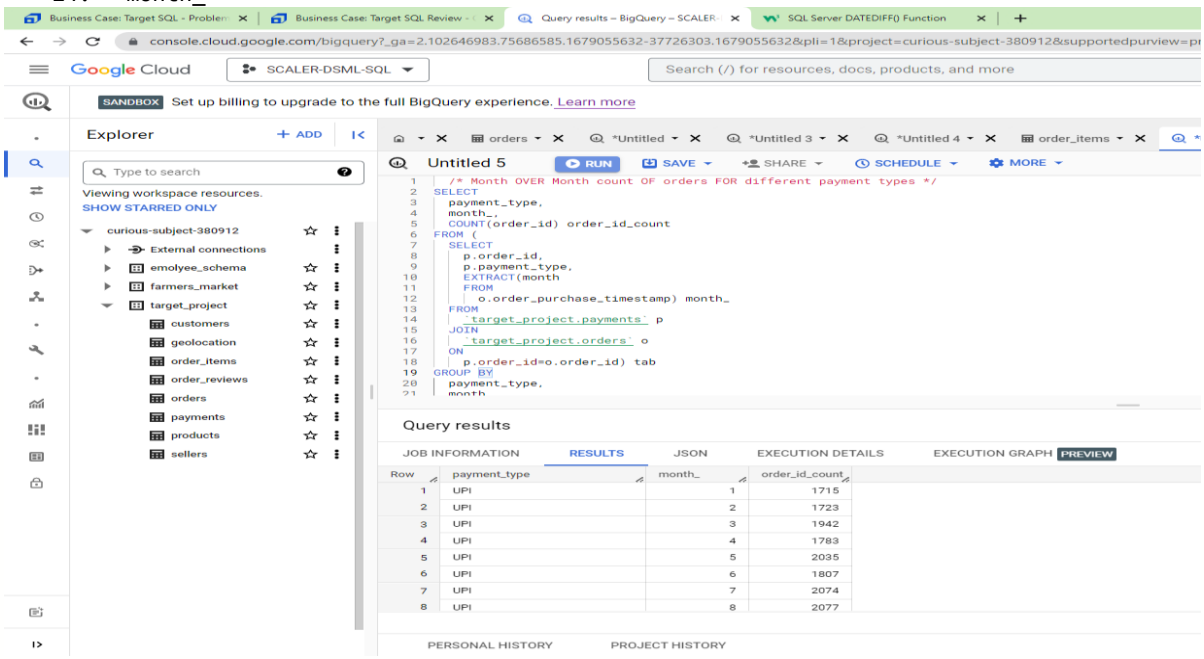
```
SELECT
    c.customer_state,
    avg(freight_value) AS mean_freight_value,
     avg(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day)) mean_time
_to_delivery,
 avg( DATE_DIFF( order_estimated_delivery_date, order_delivered_customer_date, day)) mean_d
iff_estimated_delivery

  FROM
    `target_project.customers` c
 JOIN
    `target_project.orders` o
ON
    c.customer_id=o.customer_id
  JOIN
    `target_project.order_items` AS ot
  ON
    o.order_id=ot.order_id
  GROUP BY
    c.customer_state
    order by mean_diff_estimated_delivery
    limit 5
```



6. Payment type analysis:

1. Month over Month count of orders for different payment types

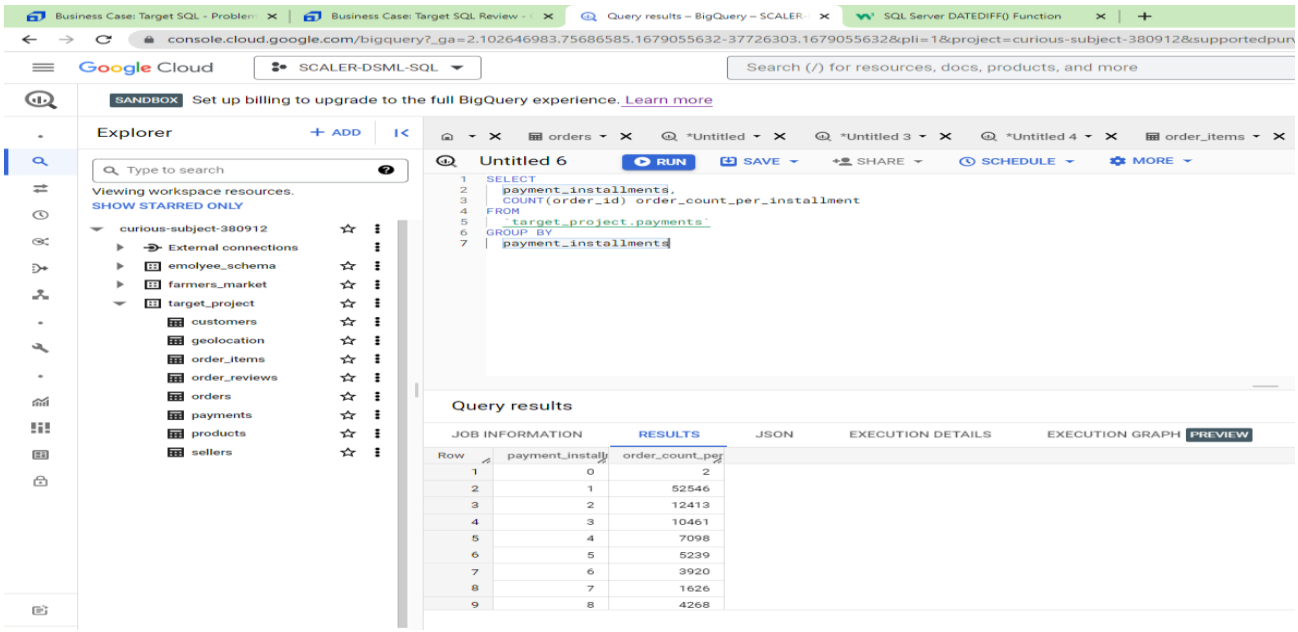
```
SELECT
  payment_type,
  month_,
  COUNT(order_id) order_id_count
FROM (
  SELECT
    p.order_id,
    p.payment_type,
    EXTRACT(month
    FROM
      o.order_purchase_timestamp) month_
  FROM
    `target_project.payments` p
  JOIN
    `target_project.orders` o
  ON
    p.order_id=o.order_id) tab
GROUP BY
  payment_type,
  month_
ORDER BY
  payment_type,
  month_
```



Upon sorting data through ordercounts it is established that, Huge number of orders are placed in 5th month through creditcards

2. Count of orders based on the no. of payment instalments

```
1.  SELECT
2.    payment_installments,
3.    COUNT(order_id) order_count_per_installment
4.  FROM
5.    `target_project.payments`
6.  GROUP BY
7.    payment_installments
```



Order count is huge for 1st payment_installment
18 orders took most number of payment instalments that are 24