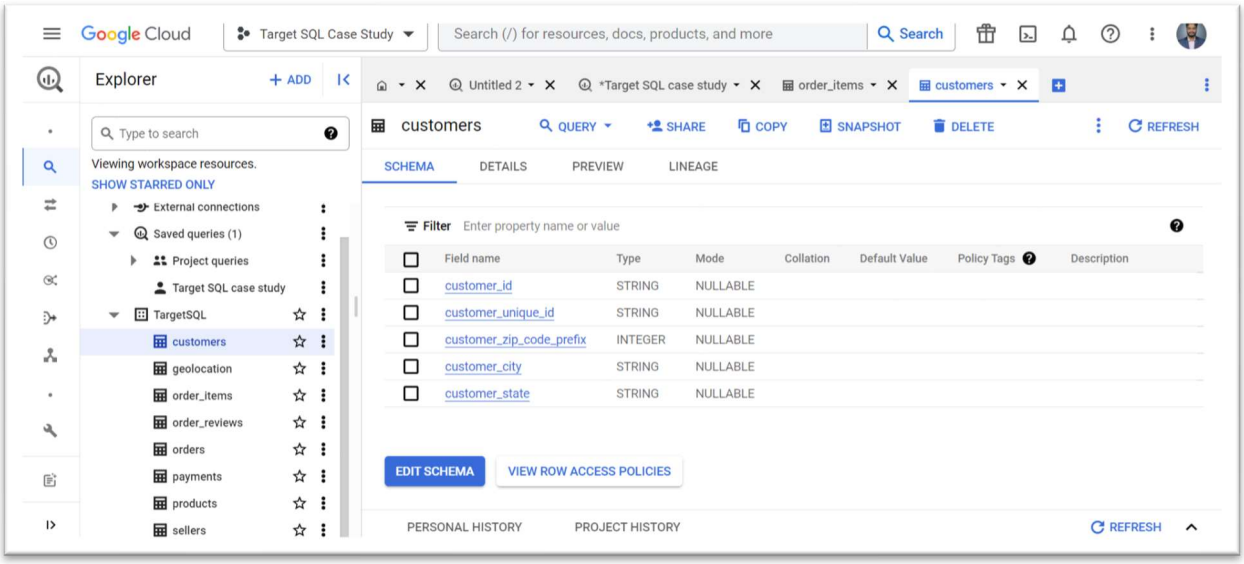
Overview:

Data analysis was carried out on *'Target: SQL Business Case'* data set which is information of orders made at Target in Brazil, valuable insights were drawn from the data set, and useful recommendations were provided using Google's BigQuery SQL tool.

1. **Exploratory Analysis:**

At first, a new project is created in Google BigQuery console and 'Target: SQL Business Case' data set is imported in the project. With the help of BigQuery console, it was analysed that there were total eight tables in the data set. The schema, columns and their datatypes in the table were observed in the BigQuery as per screenshot shown below,



With the help of below query, it was observed that the given data is from **04-09-2016 to 17-10-2018,**

```
select min(order_purchase_timestamp) as First_order,
max(order_purchase_timestamp) as Last_order
from `TargetSQL.orders`
```

There were total **8011** geolocation cities in Brazil,but total **4119 cities** and **27 states** from where customers have place orders. This information was drawn with the help of below queries,

```
select  count(distinct geolocation_city) as total_geo_cities
from `TargetSQL.geolocation`;

select count(distinct customer_city) as total_cust_cities
from `TargetSQL.customers`;

select count(distinct customer_state) as total_states
from `TargetSQL.customers`;
```

2. In-depth exploration:

With the help of below query, we can say that, e-commerce in Brazil is showing growing trend for period of sept 2016 to Aug 2018, and some seasonality peaks were observed in March 2017, May 2017, Nov 2017 and Jan 2018 as shown in the below figure 2.2,

```
select distinct extract(YEAR from o.order_purchase_timestamp) as years,
extract(MONTH from o.order_purchase_timestamp) as months,
count(o.order_id) as No_of_orders, round(sum(p.payment_value),2) as Total_order_value
from `TargetSQL.orders` o
JOIN `TargetSQL.payments` p
ON o.order_id=p.order_id
group by years,months
order by years,months;
```

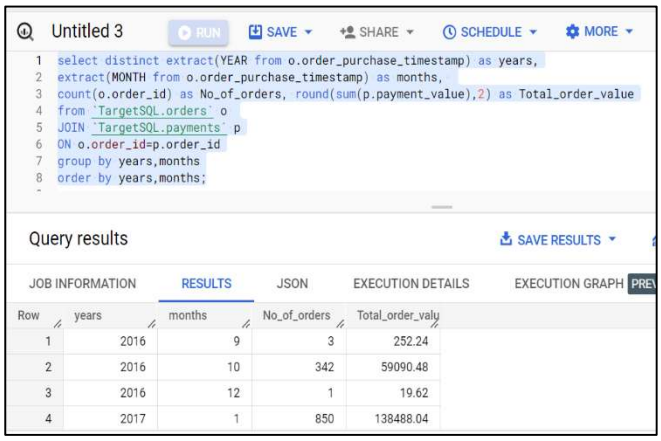


figure2.1

months	No_of_orders	Total_order_value
Sep-16	3	252.24
Oct-16	342	59090.48
Dec-16	1	19.62
Jan-17	850	138488.04
Feb-17	1886	291908.01
Mar-17	2837	449863.6
Apr-17	2571	417788.03
May-17	3944	592918.82
Jun-17	3436	511276.38
Jul-17	4317	592382.92
Aug-17	4550	674396.32
Sep-17	4516	727762.45
Oct-17	4860	779677.88
Nov-17	7863	1194882.8
Dec-17	5895	878401.48
Jan-18	7563	1115004.18
Feb-18	6952	992463.34
Mar-18	7512	1159652.12
Apr-18	7209	1160785.48
May-18	7135	1153982.15
Jun-18	6419	1023880.5
Jul-18	6507	1066540.75
Aug-18	6698	1022425.32

figure 2.2

It is observed that, highest number of orders were placed during **Afternoon time**. This insight was drawn with the help of below mentioned query and figure 2.3 and the graph 2.4 shows number of orders placed in time periods like Dawn, Morning, Afternoon and Night.

```
select distinct count(order_id) as No_of_orders,
case
when extract(HOUR from order_purchase_timestamp) <=6 and extract(HOUR from order_purchase_time
stamp)>=0 then 'Dawn'
when extract(HOUR from order_purchase_timestamp) <=12 and extract(HOUR from order_purchase_tim
estamp)>=7 then 'Morning'
when extract(HOUR from order_purchase_timestamp) <=18 and extract(HOUR from order_purchase_tim
estamp)>=13 then 'Afternoon'
when extract(HOUR from order_purchase_timestamp) <=23 and extract(HOUR from order_purchase_tim
estamp)>=19 then 'Night'
end as Time_Zone
from `TargetSQL.orders`
group by Time_Zone
order by Time_Zone;
```

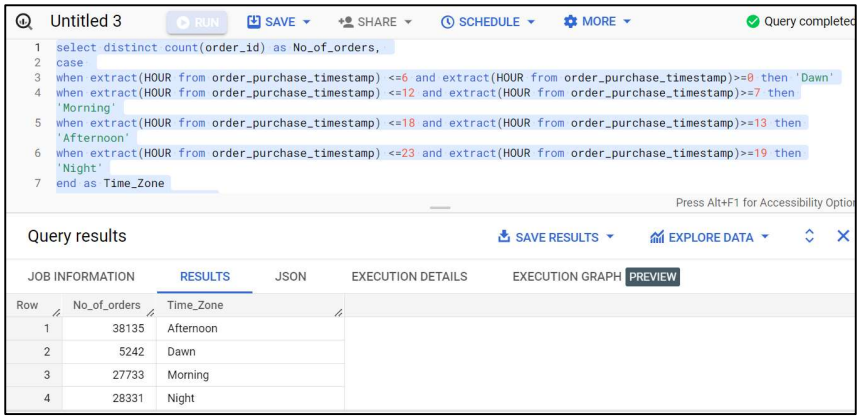
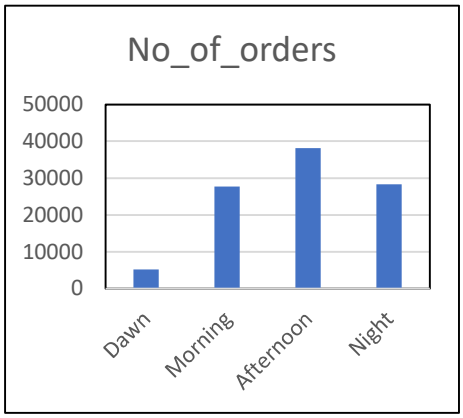


Figure 2.3



Graph 2.4

Note: For this query, time periods are considered as,

0 – 6 as Dawn

7 – 12 as Morning

13 – 18 as Afternoon

19 – 24 as Night

3. Evolution of e-commerce in Brazil region:

With the help of below query, the month by month orders by states were fetched from data set. To draw insight from the data, state SP was considered and with the help of figure 3.2 it was observed that there is growing trend in month by month orders in state SP.

```
select distinct c.customer_state, extract(YEAR from o.order_purchase_timestamp) as years,
extract(MONTH from o.order_purchase_timestamp) as months,
count(order_id) as No_of_order
from `TargetSQL.orders` o
join `TargetSQL.customers` c
On o.customer_id = c.customer_id
group by c.customer_state,years,months
order by c.customer_state,years,months;
```

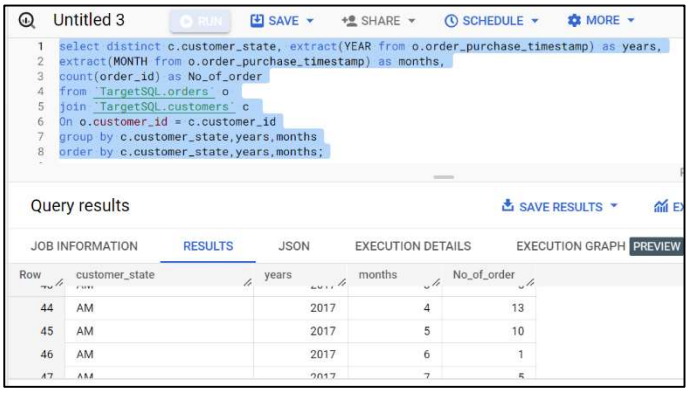


Figure 3.1

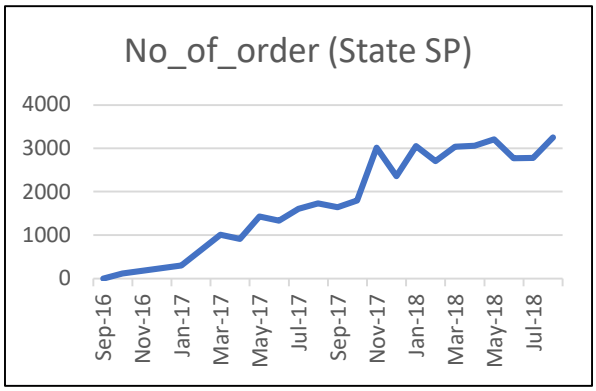


Figure 3.2

To find the customer distribution across states in Brazil, below query was executed and results were observed using figure 3.4

```
select customer_state, count(customer_unique_id) as No_of_customer
from `TargetSQL.customers`
group by customer_state
order by customer_state;
```

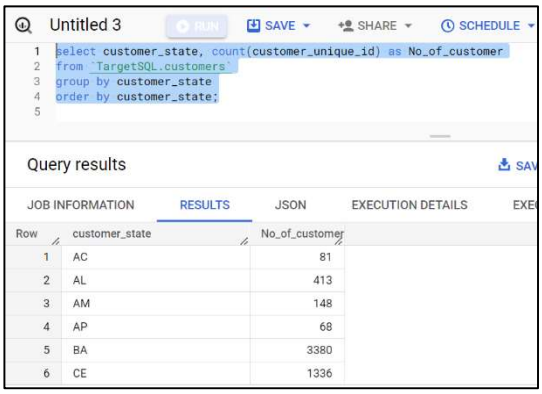


Figure 3.3

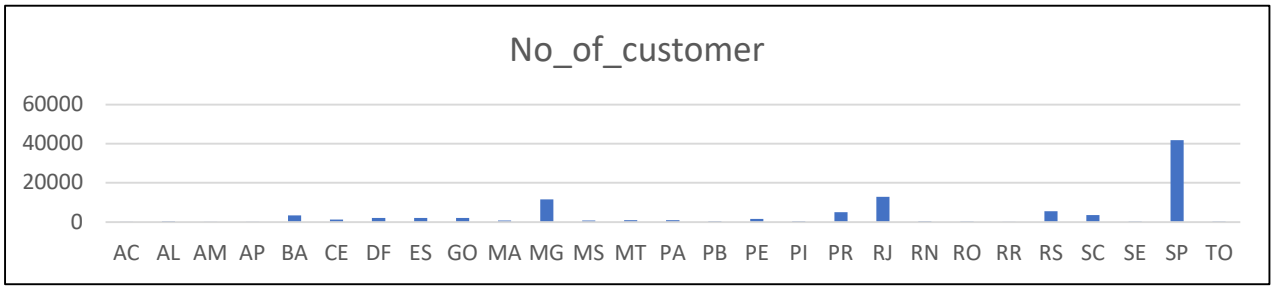


Figure 3.4

4. Impact on Economy:

To find the percentage increase in cost of orders from 2017 to 2018 (include months between Jan and Aug only) below query was written,

- For Year 2017:

```
select round(sum(payment_value),2)as Total_payment_2017
from `TargetSQL.payments`
where order_id in
(select order_id
from `TargetSQL.orders`
where order_purchase_timestamp between '2017-01-01' and '2017-08-31');
```

Total payment value in Year 2017 = **3645107.27**

- For Year 2018:

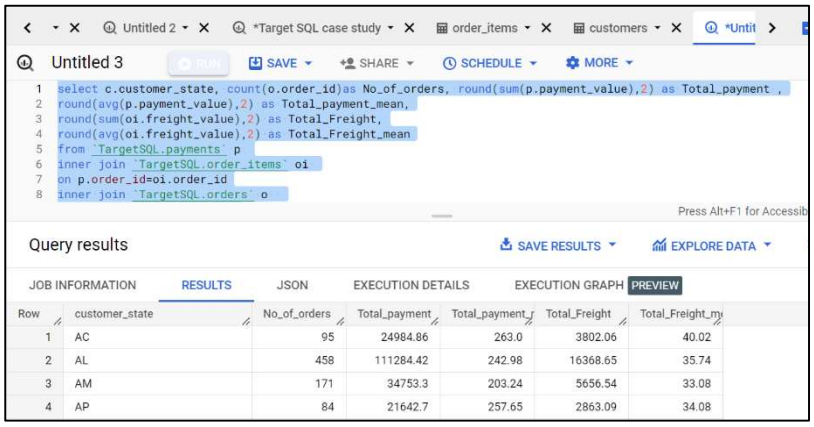
```
select round(sum(payment_value),2)as Total_payment_2018
from `TargetSQL.payments`
where order_id in
(select order_id
from `TargetSQL.orders`
where order_purchase_timestamp between '2018-01-01' and '2018-08-31');
```

Total payment value in Year 2018 = **8694669.95**

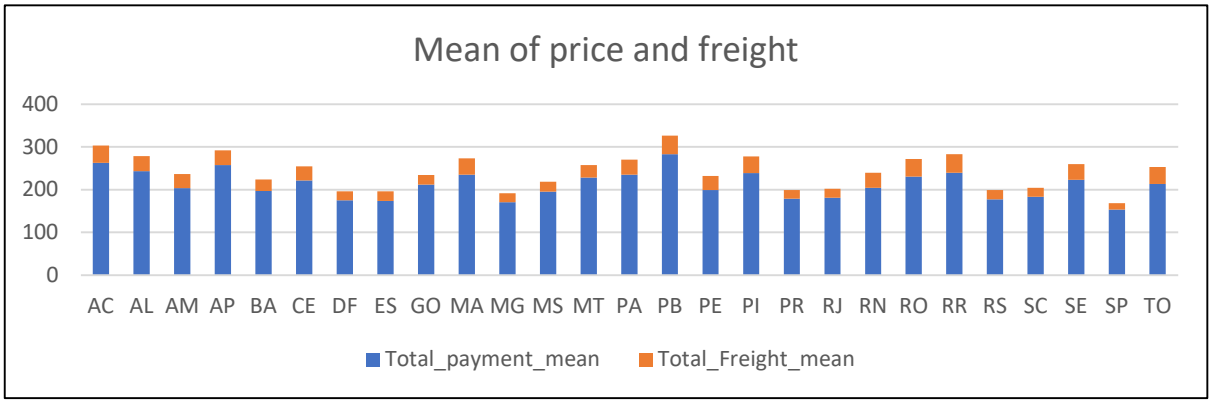
With the help of above data, the percentage increase in cost of orders from 2017 to 2018 (include months between Jan and Aug only) was calculated as **138.53%**

To calculate mean and sum of price and freight value by customer state below query was executed,

```
select c.customer_state, count(o.order_id)as No_of_orders, round(sum(p.payment_valu
e),2) as Total_payment ,
round(avg(p.payment_value),2) as Total_payment_mean,
round(sum(oi.freight_value),2) as Total_Freight,
round(avg(oi.freight_value),2) as Total_Freight_mean
from `TargetSQL.payments` p
inner join `TargetSQL.order_items` oi
on p.order_id=oi.order_id
inner join `TargetSQL.orders` o
on o.order_id= oi.order_id
inner join `TargetSQL.customers` c
on c.customer_id=o.customer_id
group by c.customer_state
order by c.customer_state;
```



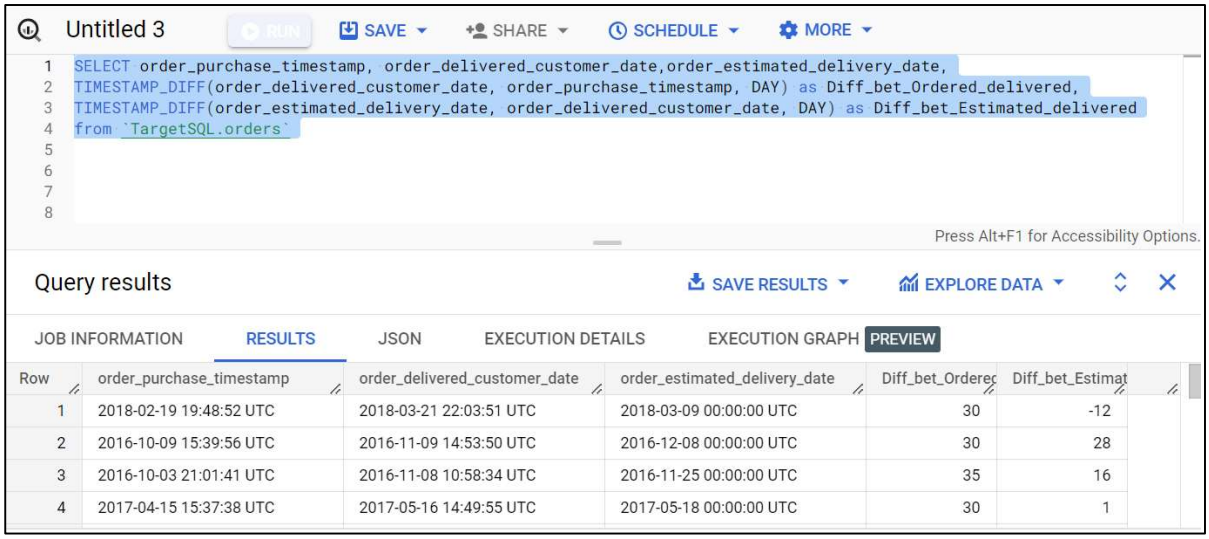
Row	customer_state	No_of_orders	Total_payment	Total_payment_r	Total_Freight	Total_Freight_m
1	AC	95	24984.86	263.0	3802.06	40.02
2	AL	458	111284.42	242.98	16368.65	35.74
3	AM	171	34753.3	203.24	5656.54	33.08
4	AP	84	21642.7	257.65	2863.09	34.08



5. **Analysis on sales, freight and delivery time:**

To find out if orders are delivered timely and difference between estimated delivery date and actual delivery date below query was executed,

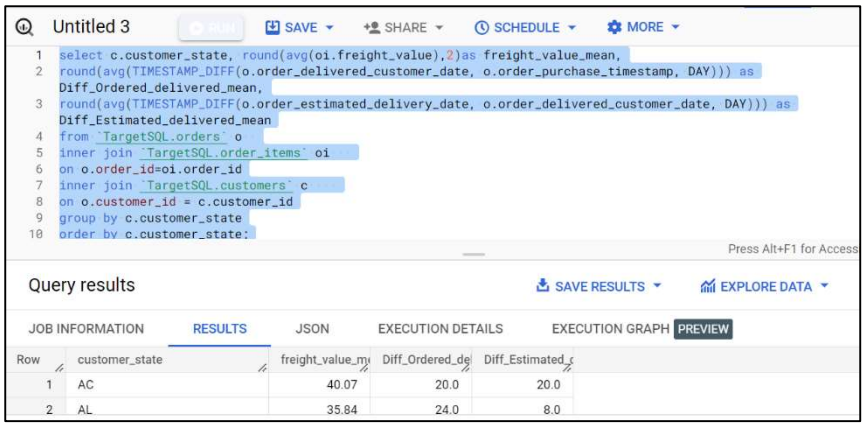
```
SELECT order_purchase_timestamp, order_delivered_customer_date,order_estimated_deli
very_date,
TIMESTAMP_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) as Dif
f_Ordered_delivered,
TIMESTAMP_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY) a
s Diff_Estimated_delivered
from `Business_Case_Target_SQL.orders`
```



Row	order_purchase_timestamp	order_delivered_customer_date	order_estimated_delivery_date	Diff_bet_Orderec	Diff_bet_Estimat
1	2018-02-19 19:48:52 UTC	2018-03-21 22:03:51 UTC	2018-03-09 00:00:00 UTC	30	-12
2	2016-10-09 15:39:56 UTC	2016-11-09 14:53:50 UTC	2016-12-08 00:00:00 UTC	30	28
3	2016-10-03 21:01:41 UTC	2016-11-08 10:58:34 UTC	2016-11-25 00:00:00 UTC	35	16
4	2017-04-15 15:37:38 UTC	2017-05-16 14:49:55 UTC	2017-05-18 00:00:00 UTC	30	1

To find mean freight, time to delivery and difference between estimated and actual delivery date by state, below query was written;

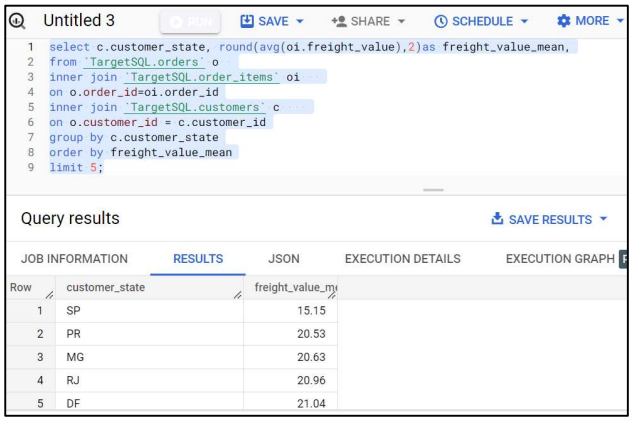
```
select c.customer_state, round(avg(oi.freight_value),2)as freight_value_mean,
round(avg(TIMESTAMP_DIFF(o.order_delivered_customer_date, o.order_purchase_timestam
p, DAY))) as Diff_Ordered_delivered_mean,
round(avg(TIMESTAMP_DIFF(o.order_estimated_delivery_date, o.order_delivered_custome
r_date, DAY))) as Diff_Estimated_delivered_mean
from `TargetSQL.orders` o
inner join `TargetSQL.order_items` oi
on o.order_id=oi.order_id
inner join `TargetSQL.customers` c
on o.customer_id = c.customer_id
group by c.customer_state
order by c.customer_state;
```



Row	customer_state	freight_value_my	Diff_Ordered_de	Diff_Estimated_c
1	AC	40.07	20.0	20.0
2	AL	35.84	24.0	8.0

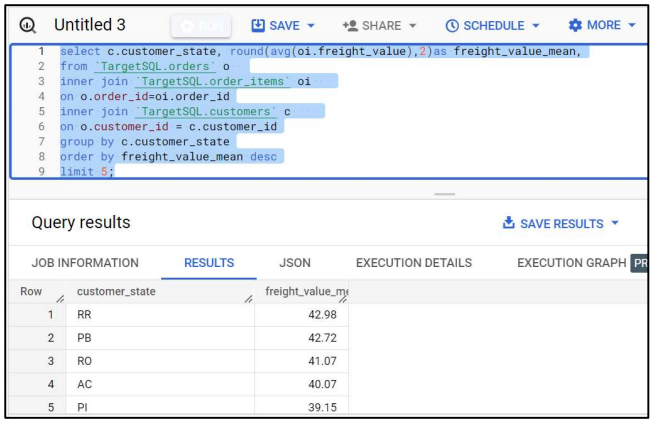
To find top 5 states with lowest average freight value below query was written and results were analysed,

```
select c.customer_state, round(avg(oi.freight_value),2)as freight_value_mean,
from `Business_Case_Target_SQL.orders` o
inner join `Business_Case_Target_SQL.order_items` oi
on o.order_id=oi.order_id
inner join `Business_Case_Target_SQL.customers` c
on o.customer_id = c.customer_id
group by c.customer_state
order by freight_value_mean
limit 5;
```



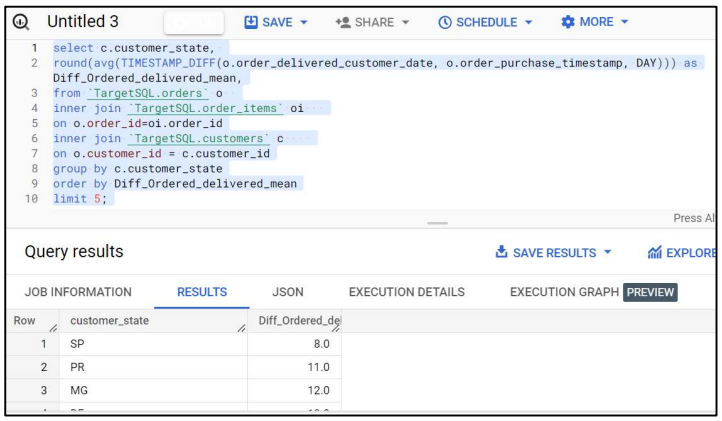
To find top 5 states with highest average freight value below query was written and results were analysed,

```
select c.customer_state, round(avg(oi.freight_value),2)as freight_value_mean,
from `TargetSQL.orders` o
inner join `TargetSQL.order_items` oi
on o.order_id=oi.order_id
inner join `TargetSQL.customers` c
on o.customer_id = c.customer_id
group by c.customer_state
order by freight_value_mean desc
limit 5;
```



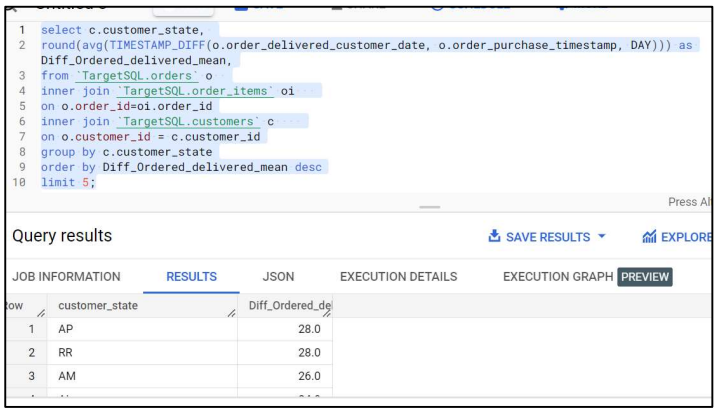
To find top 5 states with lowest average time to delivery below query was written and results were analysed,

```
select c.customer_state,
round(avg(TIMESTAMP_DIFF(o.order_delivered_customer_date, o.order_purchase_timestam
p, DAY))) as Diff_Ordered_delivered_mean,
from `TargetSQL.orders` o
inner join `TargetSQL.order_items` oi
on o.order_id=oi.order_id
inner join `TargetSQL.customers` c
on o.customer_id = c.customer_id
group by c.customer_state
order by Diff_Ordered_delivered_mean
limit 5;
```



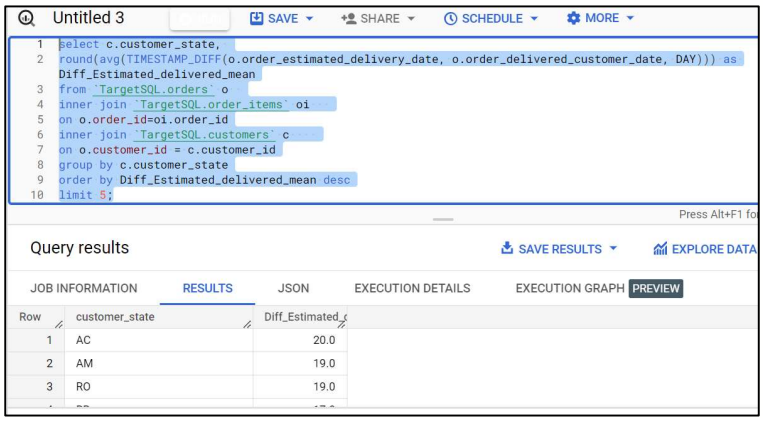
To find top 5 states with highest average time to delivery below query was written and results were analysed,

```
select c.customer_state,
round(avg(TIMESTAMP_DIFF(o.order_delivered_customer_date, o.order_purchase_timestam
p, DAY))) as Diff_Ordered_delivered_mean,
from `TargetSQL.orders` o
inner join `TargetSQL.order_items` oi
on o.order_id=oi.order_id
inner join `TargetSQL.customers` c
on o.customer_id = c.customer_id
group by c.customer_state
order by Diff_Ordered_delivered_mean desc
limit 5;
```



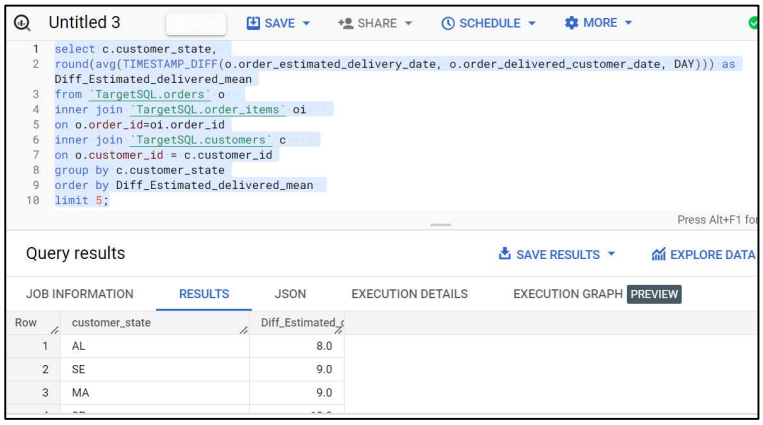
To find top 5 states with slowest delivery, below query was written and results were analysed,

```
select c.customer_state,
round(avg(TIMESTAMP_DIFF(o.order_estimated_delivery_date, o.order_delivered_custome
r_date, DAY))) as Diff_Estimated_delivered_mean
from `TargetSQL.orders` o
inner join `TargetSQL.order_items` oi
on o.order_id=oi.order_id
inner join `TargetSQL.customers` c
on o.customer_id = c.customer_id
group by c.customer_state
order by Diff_Estimated_delivered_mean desc
limit 5;
```



To find top 5 states with fastest delivery, below query was written and results were analysed,

```
select c.customer_state,
round(avg(TIMESTAMP_DIFF(o.order_estimated_delivery_date, o.order_delivered_custome
r_date, DAY))) as Diff_Estimated_delivered_mean
from `TargetSQL.orders` o
inner join `TargetSQL.order_items` oi
on o.order_id=oi.order_id
inner join `TargetSQL.customers` c
on o.customer_id = c.customer_id
group by c.customer_state
order by Diff_Estimated_delivered_mean
limit 5;
```



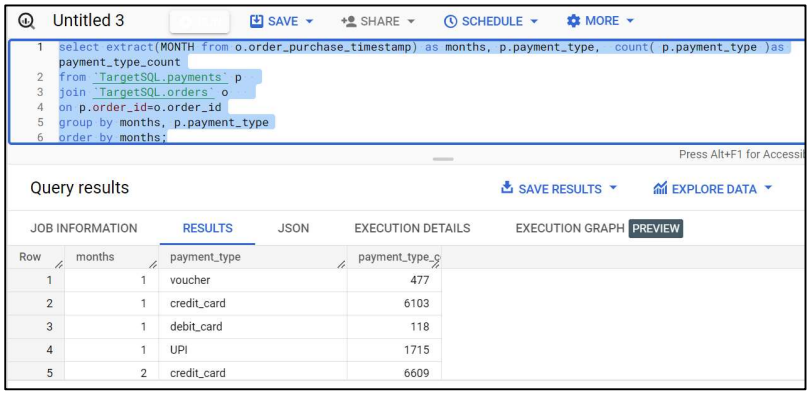
6. Payment type analysis:

There are total four different payment types viz, credit card, voucher, debit card, UPI

```
select payment_type
from `TargetSQL.payments`
group by payment_type;
```

To find out how different payment methods were used by customers, month over month count of orders for different payments fetched from dataset using following query,

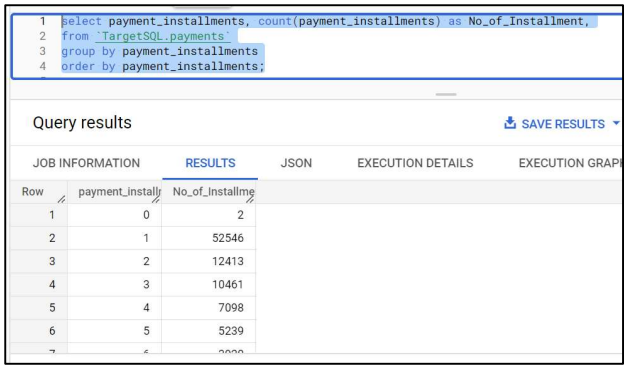
```
select extract(MONTH from o.order_purchase_timestamp) as months, p.payment_type,  c
ount( p.payment_type )as payment_type_count
from `TargetSQL.payments` p
join `TargetSQL.orders` o
on p.order_id=o.order_id
group by months, p.payment_type
order by months;
```



Row	months	payment_type	payment_type_c
1	1	voucher	477
2	1	credit_card	6103
3	1	debit_card	118
4	1	UPI	1715
5	2	credit_card	6609

To find the count of orders based on the number of payment installments, below query was executed.

```
select payment_installments, count(payment_installments) as No_of_Installment,
from `TargetSQL.payments`
group by payment_installments
order by payment_installments;
```



Row	payment_installr	No_of_Installme
1	0	2
2	1	52546
3	2	12413
4	3	10461
5	4	7098
6	5	5239

7. **Actionable Insights:**

I. As per given data, e-commerce business of Target in Brazil is growing, number of orders and total purchase value shows up trend.

II. Customers are distributed all over 27 states of brasil. SP state gave highest numbers of customers with highest amount orders placed.

III. Customers in Brazil tend to place orders in afternoon period of the day.

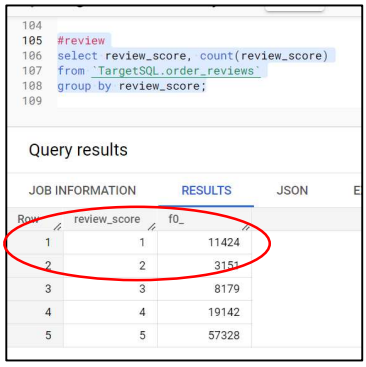
IV. In the few states delivery time is very high, it may harm user experience.

8. Recommendations:

I. Target can reduce delivery time in the states where delivery time is higher than estimated delivery time and improve user experience.

II. Target can reduce freight charges in the state where freight is higher that average freight charges.

III. With the help of reviews of customer where review score is less than or equal to 2, Target can look into the matter and solve the customer issues to improve customer user experience

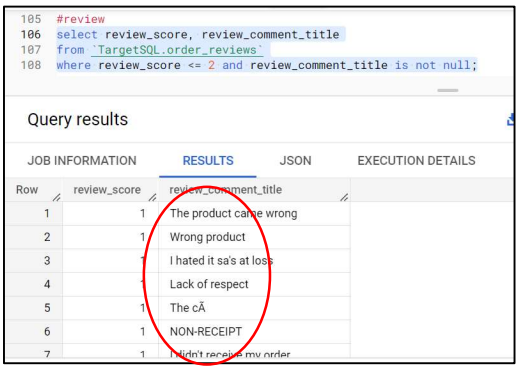


```
#review
select review_score, count(review_score)
from `TargetSQL.order_reviews`
group by review_score;
```

Query results

JOB INFORMATION | RESULTS | JSON

Row	review_score	f0_
1	1	11424
2	2	3151
3	3	8179
4	4	19142
5	5	57328



```
#review
select review_score, review_comment_title
from `TargetSQL.order_reviews`
where review_score <= 2 and review_comment_title is not null;
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	review_score	review_comment_title
1	1	The product came wrong
2	1	Wrong product
3	1	I hated it sa's at loss
4	1	Lack of respect
5	1	The cÃ
6	1	NON-RECEIPT
7	1	I didn't receive my order