Business Case Study OF Target's Brazil Operations

Target

Srisailam Jeripothula

Date:03/12/2024

Scaler DSML
SQL

Contents

Executive Summary:

This Case study focuses on the Target operations performance in Brazil during the years 2016,2017 and 2018 . These insights about the operations such as orders shipments , freight performance , reviews and payments etc were drawn by analysing the 100,000 rows of data by using SQL.from insights we can conclude that there is a potential to expand in Brazil as few states account for the large number of orders . This can only be done if we improve the under performing of other metrics such as delivery times , payments.

Introduction:

Target is a well-known American retailer and a well-known brand worldwide. By providing extraordinary pricing, imagination, innovation, and a guest experience that no other retailer can match, Target establishes itself as a preferred shopping destination.

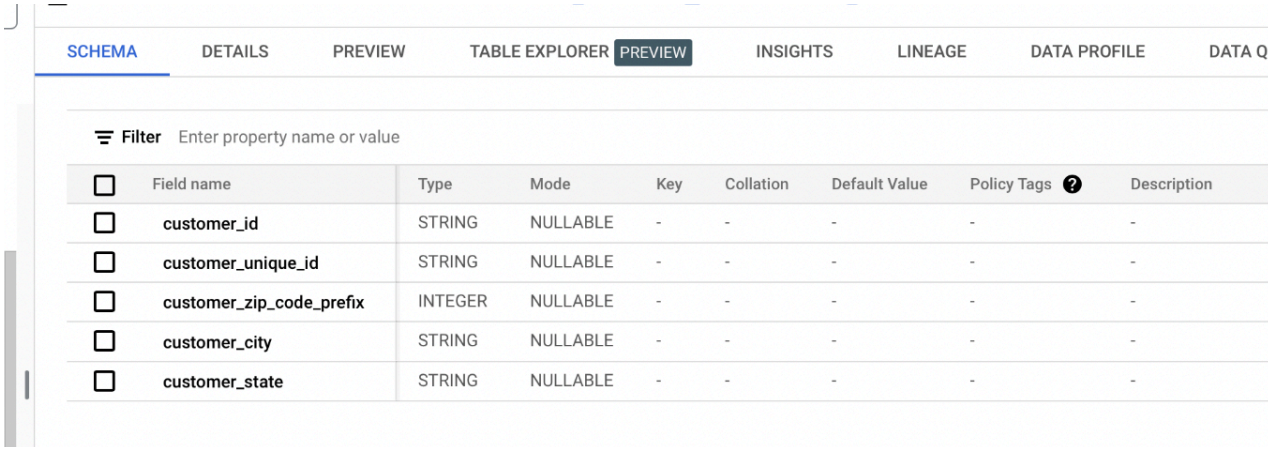
This specific business case offers useful details regarding 100,000 orders placed between 2016 and 2018 and focuses on Target's operations in Brazil. Order status, price, payment and freight performance, customer location, product features, and customer reviews are just a few of the dimensions that are fully visible in the information.

A number of business-related topics, including order processing, pricing , shipping and payments, client demographics, product attributes, and customer satisfaction levels, can be clarified by the data.

Analysis Using SQL:

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

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

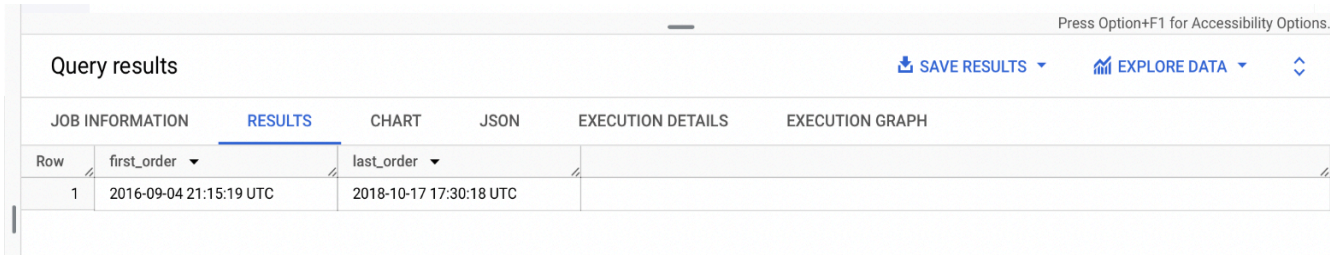


Field name	Type	Mode	Key	Collation	Default Value	Policy Tags	Description
customer_id	STRING	NULLABLE	-	-	-	-	-
customer_unique_id	STRING	NULLABLE	-	-	-	-	-
customer_zip_code_prefix	INTEGER	NULLABLE	-	-	-	-	-
customer_city	STRING	NULLABLE	-	-	-	-	-
customer_state	STRING	NULLABLE	-	-	-	-	-

Data types of customer details are strings except for the “customer_zip_code_prefix” is an integer data type and these values can be null.

1.2.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 `ace-charter-438705-d0.target.orders`;
```



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

The given data indeed accounts for the years 2016,2017 and 2018 but it does count for 3 years .It gives us slightly more than 2 years of data.

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

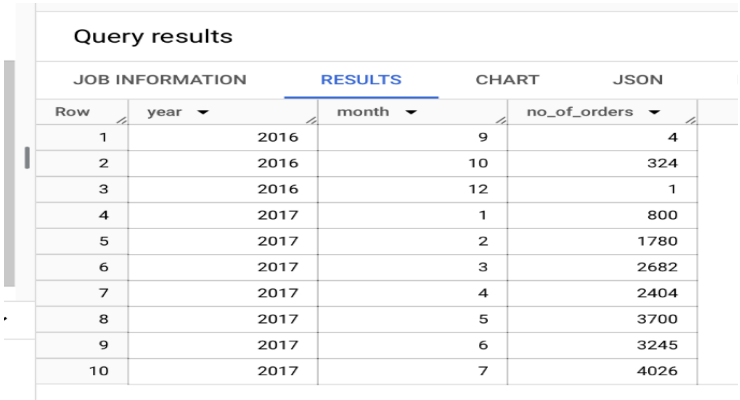
```
select count(distinct customer_city) as no_of_cities,
count(distinct customer_state) as no_of_states
from `ace-charter-438705-d0.target.orders` as o join `ace-charter-438705-d0.target.customers` as
c on o.customer_id=c.customer_id ;
```

All the states and customers which are registered in the customers table are ordering the products as the number of cities and states from the orders table is the same as the customer table as the question specifically asks for the number of cities and states for the given period . data of those is pulled from orders .

2.In-depth Exploration:

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

```
select extract(year from order_purchase_timestamp) as year
,  extract(month from order_purchase_timestamp)as month,count(*) as no_of_orders
from `ace-charter-438705-d0.target.orders`
group by  extract(year from order_purchase_timestamp)
,extract(month from order_purchase_timestamp) order by year,month;
```

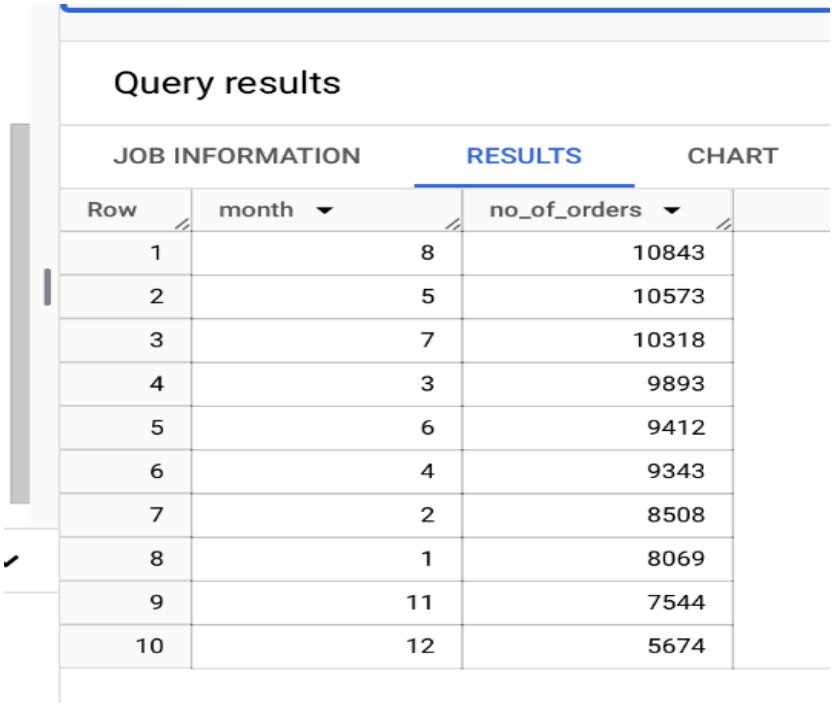


Row	year	month	no_of_orders
1	2016	9	4
2	2016	10	324
3	2016	12	1
4	2017	1	800
5	2017	2	1780
6	2017	3	2682
7	2017	4	2404
8	2017	5	3700
9	2017	6	3245
10	2017	7	4026

By observing the results there is gradual increase in orders from 2016-09 to end of 2017 but there is slight dip in 2018 over all orders were uniform in 2017 and 2018 but the data of 2017 november is missing.

2.2.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,count(*) as no_of_orders
from `ace-charter-438705-d0.target.orders`
group by
extract(month from order_purchase_timestamp) order by no_of_orders desc;
```



Query results

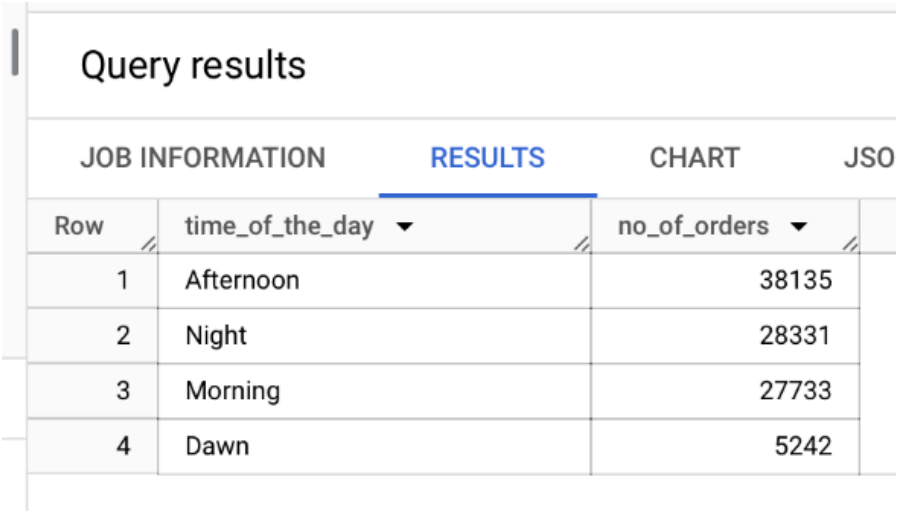
JOB INFORMATION | RESULTS | CHART

Row	month	no_of_orders
1	8	10843
2	5	10573
3	7	10318
4	3	9893
5	6	9412
6	4	9343
7	2	8508
8	1	8069
9	11	7544
10	12	5674

The orders peaked during May,September and May . For optimum results the preparations for the peak ideally start from the third month to meet the customer demand.it's better to prepare for the peak from March to meet the customer demand.

2.3.During what time of the day, do the Brazilian customers mostly place their orders? (Dawn(0-6 hrs), Morning(7-12 hrs), Afternoon(13-18 hrs) or Night(19-23 hrs))

```
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"
else "Night" end as time_of_the_day,count(*) as no_of_orders
from `ace-charter-438705-d0.target.orders` group by time_of_the_day order by no_of_orders
desc;
```



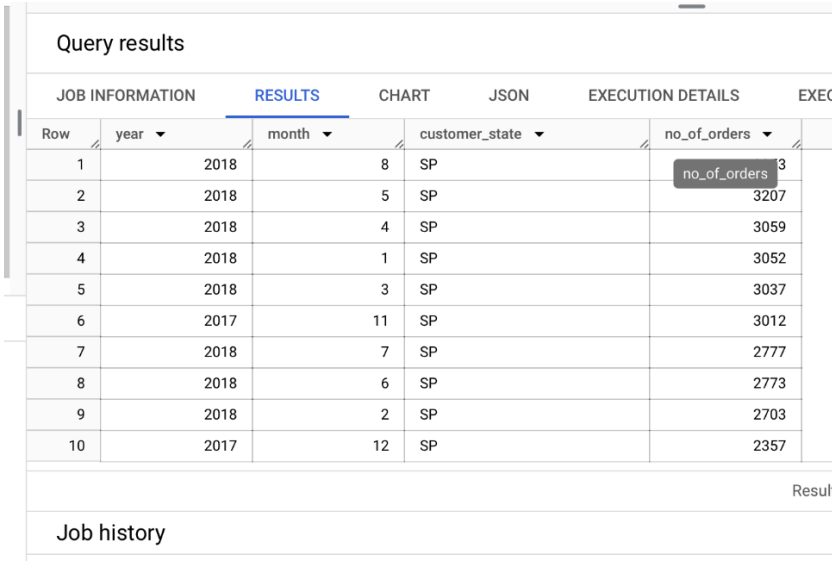
Row	time_of_the_day	no_of_orders
1	Afternoon	38135
2	Night	28331
3	Morning	27733
4	Dawn	5242

The more orders are ordered in the afternoon to better serve the customers it is better to have more people in the afternoon shift for customer query resolution .

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

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

```
select extract(year from order_purchase_timestamp)as year,
extract(month from order_purchase_timestamp)as month,customer_state,count(*) as
no_of_orders
from `ace-charter-438705-d0.target.orders` as o join`ace-charter-438705-d0.target.customers` as
c on o.customer_id=c.customer_id group by year,month,customer_state order by no_of_orders
desc;
```



Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXEC

Row	year	month	customer_state	no_of_orders
1	2018	8	SP	3
2	2018	5	SP	3207
3	2018	4	SP	3059
4	2018	1	SP	3052
5	2018	3	SP	3037
6	2017	11	SP	3012
7	2018	7	SP	2777
8	2018	6	SP	2773
9	2018	2	SP	2703
10	2017	12	SP	2357

no_of_orders

Resul

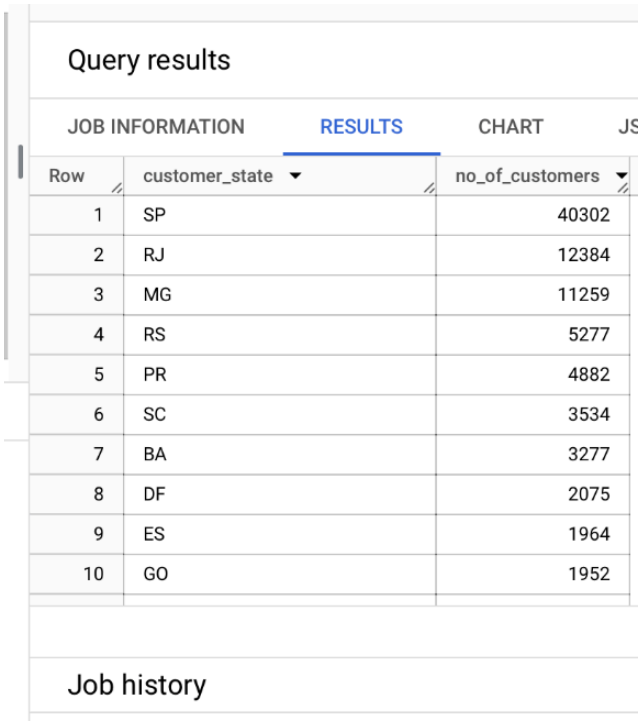
Job history

```
select year,month, count(distinct(customer_state)) from (select extract(year from
order_purchase_timestamp)as year,
extract(month from order_purchase_timestamp)as month,customer_state,count(*) as
no_of_orders
from `ace-charter-438705-d0.target.orders` as o join `ace-charter-438705-d0.target.customers` as
c on
o.customer_id=c.customer_id group by year,month,customer_state order by no_of_orders asc)as
t where no_of_orders <500 group by year,month;
```

More orders are from the state “SP” and only on average 3 states are ordering more than 500 orders out of 27 states when we analyse the monthly data . There is huge scope in increasing the orders count by focusing on other states , there could be more factors involved in these orders .

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

```
select customer_state,count(distinct customer_unique_id) as no_of_customers
from `ace-charter-438705-d0.target.customers`
group by customer_state order by no_of_customers desc;
```



Query results

JOB INFORMATION | RESULTS | CHART | JS

Row	customer_state	no_of_customers
1	SP	40302
2	RJ	12384
3	MG	11259
4	RS	5277
5	PR	4882
6	SC	3534
7	BA	3277
8	DF	2075
9	ES	1964
10	GO	1952

Job history

Customers are not distributed uniformly , more number of customers were registered from the "SP" and least from "RR"

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.

```
select (((lags-sum_of_sales)/sum_of_sales )*100) as perincrease
from
(select sum_of_sales,lag(sum_of_sales)over(order by sum_of_sales desc ) as lags
from
(select distinct extract(year from order_purchase_timestamp) as year , sum(payment_value) over
(partition by (extract(year from order_purchase_timestamp))) as sum_of_sales
from `ace-charter-438705-d0.target.payments` as p join `ace-charter-438705-d0.target.orders` as
o on
p.order_id=o.order_id where extract(year from order_purchase_timestamp)in(2017,2018) and
extract(month from order_purchase_timestamp) between 1 and 8)as t ) as s  ;
```

Row	perincrease
1	null
2	136.9768716466...

There is a 140% increase .

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

```
select distinct customer_state, avg(payment_value)over(partition by customer_state) as
avg_value,sum(payment_value)over(
partition by customer_state
) as sum_of_sales
from `ace-charter-438705-d0.target.orders` as o join `ace-charter-438705-d0.target.customers` as
c on
c.customer_id=o.customer_id  join `ace-charter-438705-d0.target.payments` as p on
p.order_id=o.order_id
order by sum_of_sales desc,avg_value desc
```

;

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTI

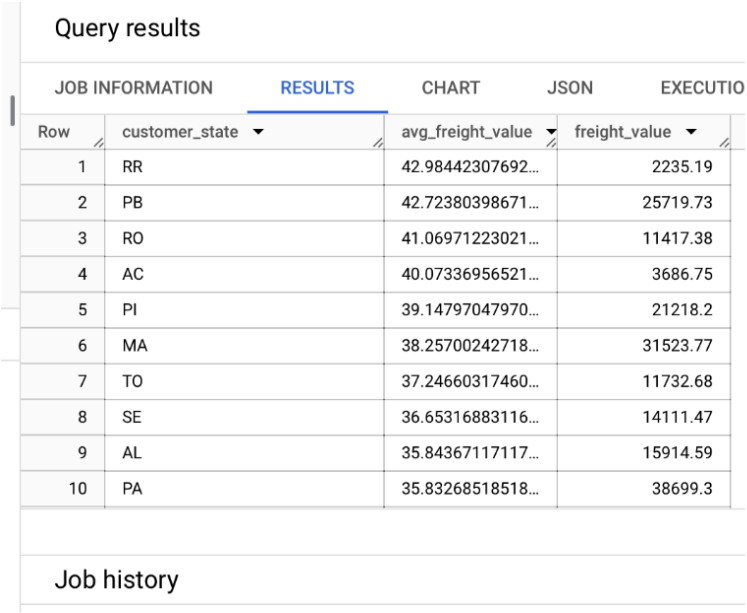
Row	customer_state	avg_value	sum_of_sales
1	SP	137.5046297739...	5998226.96
2	RJ	158.5258882235...	2144379.69
3	MG	154.7064336473...	1872257.26
4	RS	157.1804057868...	890898.54
5	PR	154.1536259977...	811156.38
6	SC	165.9793367075...	623086.43
7	BA	170.8160166204...	616645.82
8	DF	161.1347912885...	355141.08
9	GO	165.7634043560...	350092.31
10	ES	154.7069530137...	325967.55

Job history

The highest sales come from the state “SP” and we can also observe from above questions that this “SP” state also accounts for the highest number of orders . On other hand, “PB” has the highest average sales value .

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

```
select distinct customer_state, avg(freight_value)over(partition by customer_state) as
avg_freight_value,sum(freight_value)over(partition by customer_state) as freight_value
from `ace-charter-438705-d0.target.orders` as o join `ace-charter-438705-d0.target.customers` as
c on
c.customer_id=o.customer_id  join `ace-charter-438705-d0.target.order_items` as p on
p.order_id=o.order_id
order by avg_freight_value desc,freight_value desc;
```


Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTIO

Row	customer_state	avg_freight_value	freight_value
1	RR	42.98442307692...	2235.19
2	PB	42.72380398671...	25719.73
3	RO	41.06971223021...	11417.38
4	AC	40.07336956521...	3686.75
5	PI	39.14797047970...	21218.2
6	MA	38.25700242718...	31523.77
7	TO	37.24660317460...	11732.68
8	SE	36.65316883116...	14111.47
9	AL	35.84367117117...	15914.59
10	PA	35.83268518518...	38699.3

Job history

The “PB” which is having the highest average sales is also having the approximately higher average freight value .The state “SP” is having the maximum freight value in the given data .

5.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:

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

```
select order_id,datetime_diff(order_delivered_customer_date,order_purchase_timestamp,day)as time_to_deliver,datetime_diff(order_estimated_delivery_date,order_delivered_customer_date,day)as diff_estimated_delivery
from `ace-charter-438705-d0.target.orders`order by diff_estimated_delivery desc;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTI

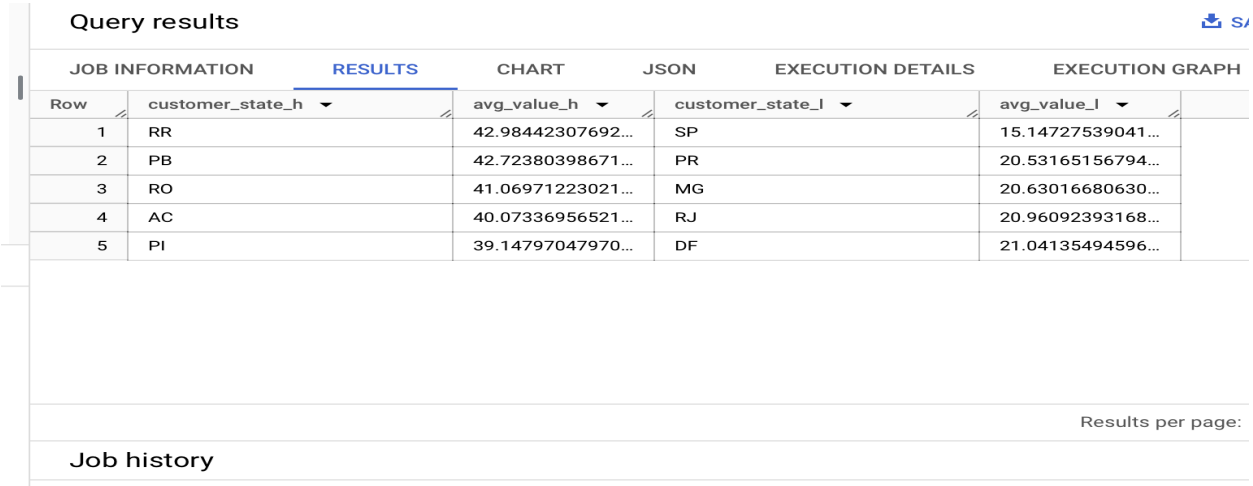
Row	order_id	time_to_deliver	diff_estimated_delive
1	0607f0efea4b566f1eb8f7d3c2...	3	146
2	c72727d29cde4cf870d569bf6...	6	139
3	eec7f369423b033e549c02f3c...	20	134
4	c2bb89b5c1dd978d507284be...	16	123
5	40dc2ba6f322a17626aac6244...	7	108
6	1a695d543b7302aa9446c8d5f...	12	83
7	39e0115911bf404857e14baa7...	11	82
8	38930f76efb00b138f4d632e4d...	11	77
9	c5132855100a12d63ed4e8ae0...	12	77
10	559eea5a72341a4c82dbce988...	13	77

Job history

By executing the above query we can find the time to deliver the order and difference between the estimated and actual delivery date , in the results we can see see the positive and negative values in diff_estimated_delivery positive value signifies the order has been delivered within the estimated delivery date and negative value indicated the order has been delivered late , there are a lot of order has been delivered after passing the estimated date which incur costs to the company and need to work on the scheduling of delivery date by working closing with supply chain and delivery partners .

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

```
with cte1 as (select distinct customer_state as customer_state_h ,avg(freight_value)over(partition
by customer_state) as avg_value_h,
from `ace-charter-438705-d0.target.orders` as o join `ace-charter-438705-d0.target.customers` as
c on c.customer_id=o.customer_id  join `ace-charter-438705-d0.target.order_items` as p on
p.order_id=o.order_id
order by avg_value_h desc limit 5),
 cte2 as (select distinct customer_state as customer_state_l, avg(freight_value)over(partition by
customer_state) as avg_value_l,
from `ace-charter-438705-d0.target.orders` as o join `ace-charter-438705-d0.target.customers` as
c on
c.customer_id=o.customer_id  join `ace-charter-438705-d0.target.order_items` as p on
p.order_id=o.order_id
order by avg_value_l  limit 5),
cte3 as ( select row_number()over(order by avg_value_h desc) as rh,
customer_state_h,avg_value_h from cte1),
cte4 as ( select row_number()over(order by avg_value_l )as rl, customer_state_l,avg_value_l
from cte2)
select customer_state_h,avg_value_h,customer_state_l,avg_value_l from cte3 as c3 join cte4 as
c4 on rh=rl;
```



The states with highest and lowest freight values were given above.

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

```
with cte1 as (select customer_state as customer_state_h
,round(avg(datetime_diff(order_delivered_customer_date,order_purchase_timestamp,day)))as
avg_time_to_deliver_h
from `ace-charter-438705-d0.target.orders` as o
join `ace-charter-438705-d0.target.customers` as c on
c.customer_id=o.customer_id group by customer_state order by avg_time_to_deliver_h desc
limit 5),
 cte2 as (select customer_state as customer_state_l
,round(avg(datetime_diff(order_delivered_customer_date,order_purchase_timestamp,day)))as
avg_time_to_deliver_l
from `ace-charter-438705-d0.target.orders` as o
join `ace-charter-438705-d0.target.customers` as c on
c.customer_id=o.customer_id group by customer_state order by avg_time_to_deliver_l
limit 5),
cte3 as ( select row_number()over(order by avg_time_to_deliver_h desc) as rh,
customer_state_h,avg_time_to_deliver_h from cte1),
cte4 as ( select row_number()over(order by avg_time_to_deliver_l )as rl,
customer_state_l,avg_time_to_deliver_l from cte2)

select customer_state_h,avg_time_to_deliver_h,customer_state_l,avg_time_to_deliver_l from
cte3 as c3 join cte4 as c4 on rh=rl;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GI

Row	customer_state_h	avg_time_to_deliver_	customer_state_l	avg_time_to_deliver_
1	RR	29.0	SP	8.0
2	AP	27.0	MG	12.0
3	AM	26.0	PR	12.0
4	AL	24.0	DF	13.0
5	PA	23.0	SC	14.0

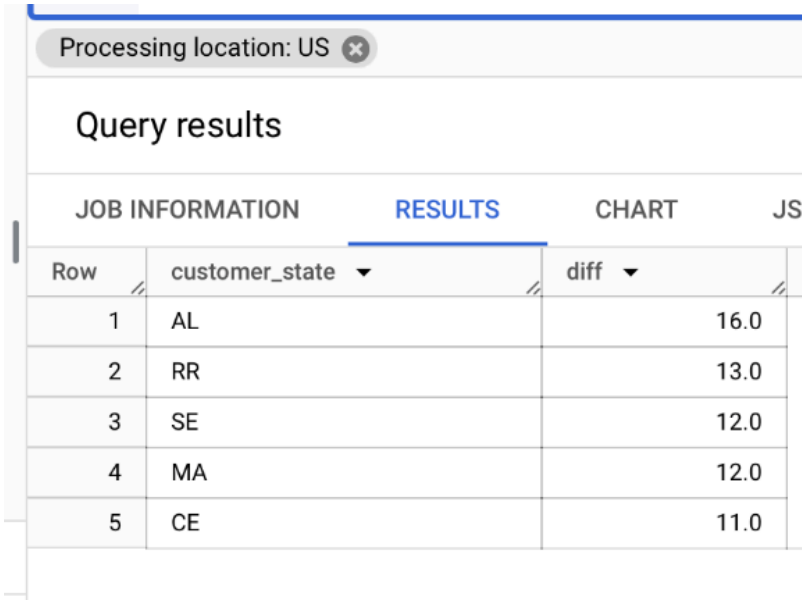
Results per

Job history

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.

```
select customer_state,round(t.avg_time_to_deliver-t.avg_diff_estimated_delivery) as diff
from(select customer_state,
avg(datetime_diff(order_delivered_customer_date,order_purchase_timestamp,day)) as
avg_time_to_deliver,avg
(datetime_diff(order_estimated_delivery_date,order_delivered_customer_date,day))as
avg_diff_estimated_delivery
from `ace-charter-438705-d0.target.orders` as o join `ace-charter-438705-d0.target.customers` as
c on c.customer_id=o.customer_id  group by customer_state)as t order by diff desc
limit 5 ;
```

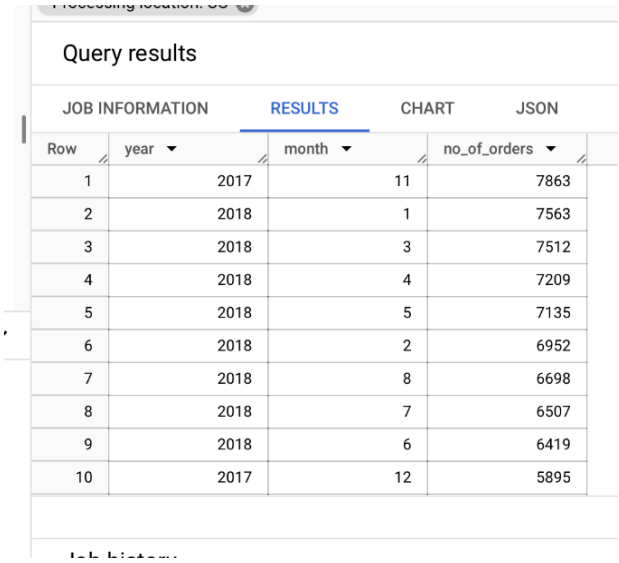


Row	customer_state	diff
1	AL	16.0
2	RR	13.0
3	SE	12.0
4	MA	12.0
5	CE	11.0

6.Analysis based on the payments:

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

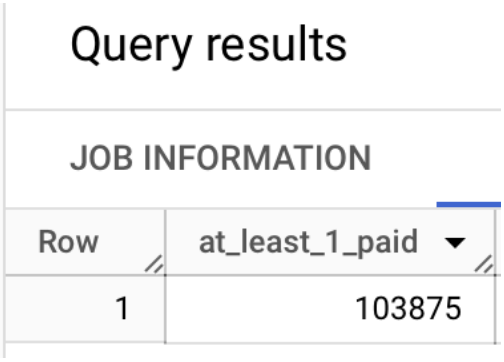
```
select extract(year from order_purchase_timestamp) as year,extract(month from
order_purchase_timestamp) as month ,count(*) as no_of_orders
 from `ace-charter-438705-d0.target.payments`as p
join `ace-charter-438705-d0.target.orders` as o on p.order_id=o.order_id
group by year,month order by no_of_orders desc;
```



Row	year	month	no_of_orders
1	2017	11	7863
2	2018	1	7563
3	2018	3	7512
4	2018	4	7209
5	2018	5	7135
6	2018	2	6952
7	2018	8	6698
8	2018	7	6507
9	2018	6	6419
10	2017	12	5895

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

```
select count(payment_value) as at_least_1_paid
 from `ace-charter-438705-d0.target.payments`
where payment_value!=0 and payment_installments!=0;
```



Lessons Learned :

Target is gaining in momentum in the Brazil Market . By the given data and time period it is hard to conclude how well the Target is performing as we don't have the full three years of data to estimate but . It would have been done better if we had more data . Target is performing very well in a few states particularly in "SP". There is a scope to enhance its sales by marketing I assume . The reason for this might be delayed deliveries , freight value,and being prepared for the peak periods and times . By allocating more resources in Brazil we can expect rapid growth in Brazil.

Conclusion:

Target can enhance its performance by concentrating on other states of Brazil by allocating more resources , by well preparing for the peak season in advance.