Target Business Case Study

Description:

In this Business case study, we are looking into Company Name Target only in the Brazil Location and the data is collected between the Years 2016 to 2018.

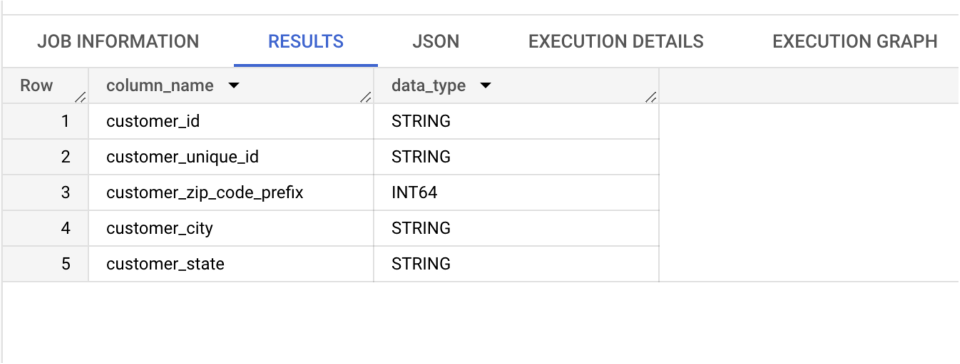
The analysis is performed to provide you the meaningful insights about Order Processing, Pricing strategies, Customer Demographics, Product characteristics and many more

1)1) Data type of all columns in the "customers" table.

Query:

```
SELECT
          column_name,
          data_type
          FROM Target.INFORMATION_SCHEMA.COLUMNS
          WHERE table_name = 'customers'
```

Output:



Row	column_name	data_type
1	customer_id	STRING
2	customer_unique_id	STRING
3	customer_zip_code_prefix	INT64
4	customer_city	STRING
5	customer_state	STRING

Insights:

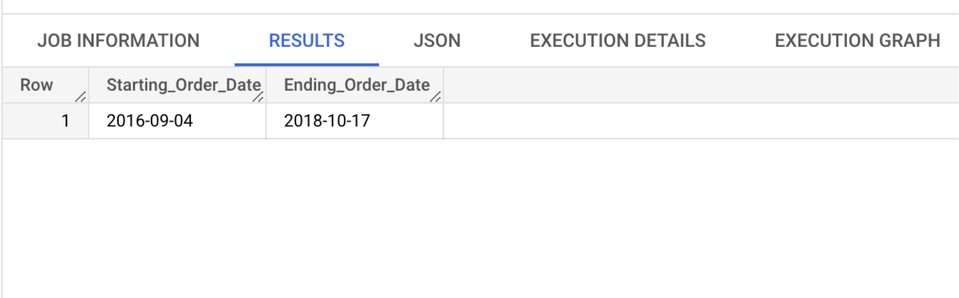
>> This Query helps us to understand the each and every table and the columns it holds and the data types of the columns.

1)2) Get the time range between which the orders were placed.

Query:

```
        Select
                  cast(min(order_purchase_timestamp) as date) as Starting_Order_Date,
                  cast(max(order_purchase_timestamp) as date) as Ending_Order_Date
        from `Target.orders`
```

Output:



Row	Starting_Order_Date	Ending_Order_Date
1	2016-09-04	2018-10-17

Insights:

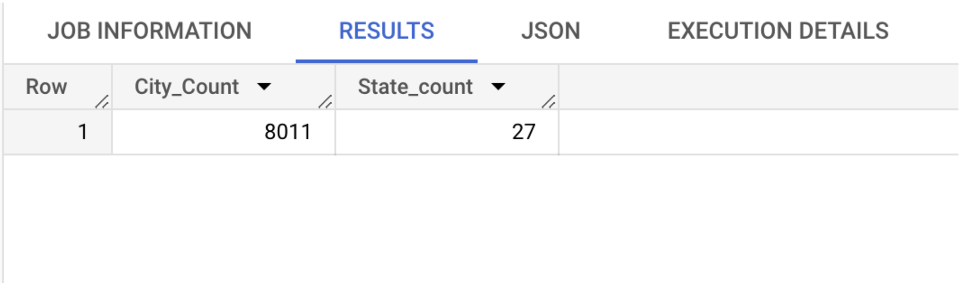
We can see what is the earliest and the latest order date given in the data set.

1)3) Count the number of Cities and States in our dataset.

Query:

```
select
                 count (distinct geolocation_city) as City_Count,
                 count (distinct geolocation_state) as State_count
        from `Target.geolocation`
```

Output:



Row	City_Count	State_count
1	8011	27

Insights:

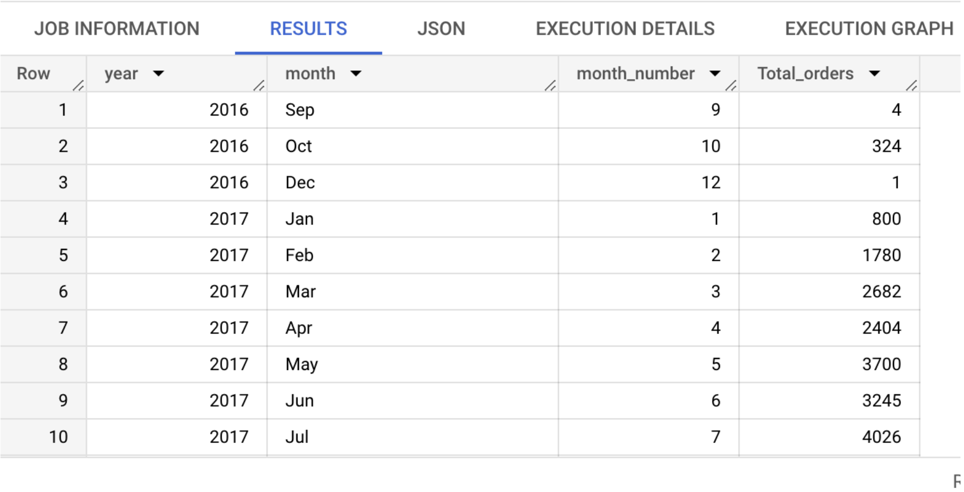
We can see the total number of City and states in the given Data set

2)1) Is there a growing trend in the no. of orders placed over the past years?

Query:

```
select
                    extract (year from order_purchase_timestamp) as year,
                    format_date('%b', order_purchase_timestamp) as month,
                    extract (month from order_purchase_timestamp) as month_number,
                    count(order_purchase_timestamp) as Total_orders
          from `Target.orders`
          group by month,year,month_number
          order by 1,3
```

Output:



Row	year	month	month_number	Total_orders
1	2016	Sep	9	4
2	2016	Oct	10	324
3	2016	Dec	12	1
4	2017	Jan	1	800
5	2017	Feb	2	1780
6	2017	Mar	3	2682
7	2017	Apr	4	2404
8	2017	May	5	3700
9	2017	Jun	6	3245
10	2017	Jul	7	4026

Insights:

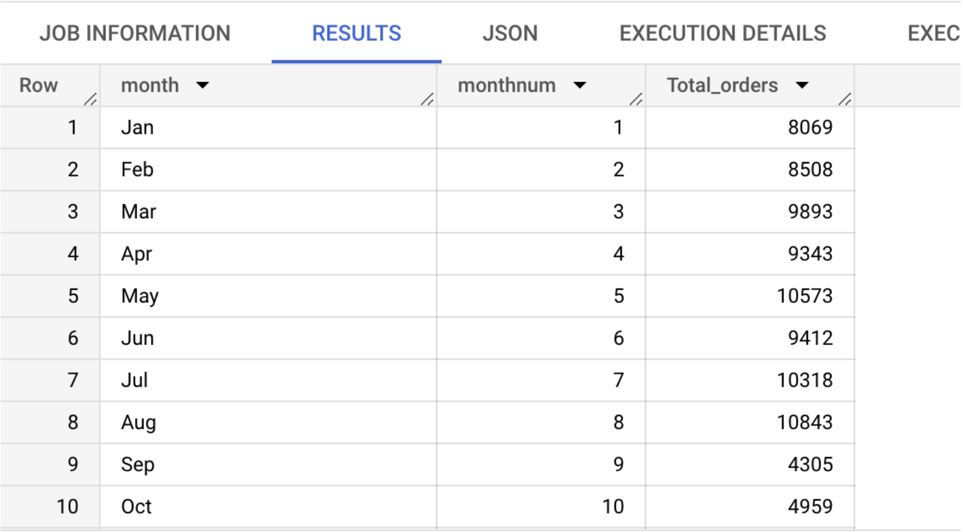
We can see the order count by year and by month. And we can find out about what kind of trend it is following.

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

Query:

```
        select
                format_date('%b', order_purchase_timestamp) as month,
                extract (month from order_purchase_timestamp) as monthnum,
                count(order_purchase_timestamp) as Total_orders
        from `Target.orders`
        group by month,monthnum
        order by monthnum
```

Output:



Row	month	monthnum	Total_orders
1	Jan	1	8069
2	Feb	2	8508
3	Mar	3	9893
4	Apr	4	9343
5	May	5	10573
6	Jun	6	9412
7	Jul	7	10318
8	Aug	8	10843
9	Sep	9	4305
10	Oct	10	4959

Insights:

We can see that after the first Quarter the order count has been increased until 2 and 3 Quarters but in the 4th Q again the order has been decreased.

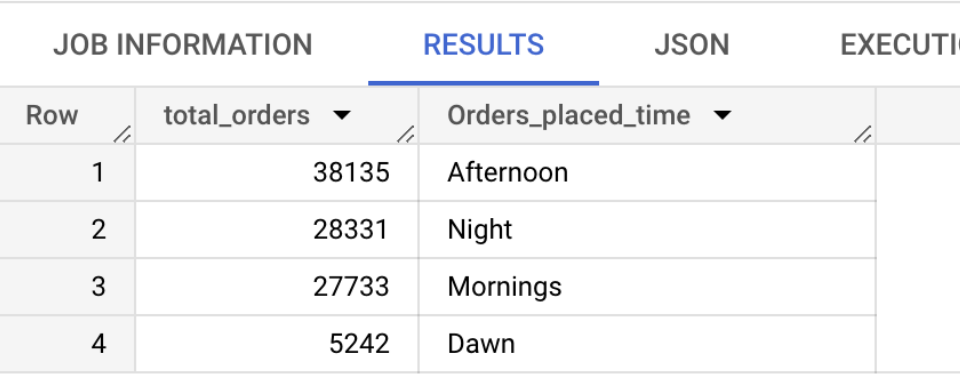
2)3 During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

Query:

```
select
        count(order_id) as total_orders,
        Orders_placed_time
    from
        (
            select
                order_id,
                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 Orders_placed_time
                from `Target.orders`
                order by 1 desc
        ) as tab
group by tab.Orders_Placed_time
order by 1 desc
```

Output:



Row	total_orders	Orders_placed_time
1	38135	Afternoon
2	28331	Night
3	27733	Mornings
4	5242	Dawn

Insights:

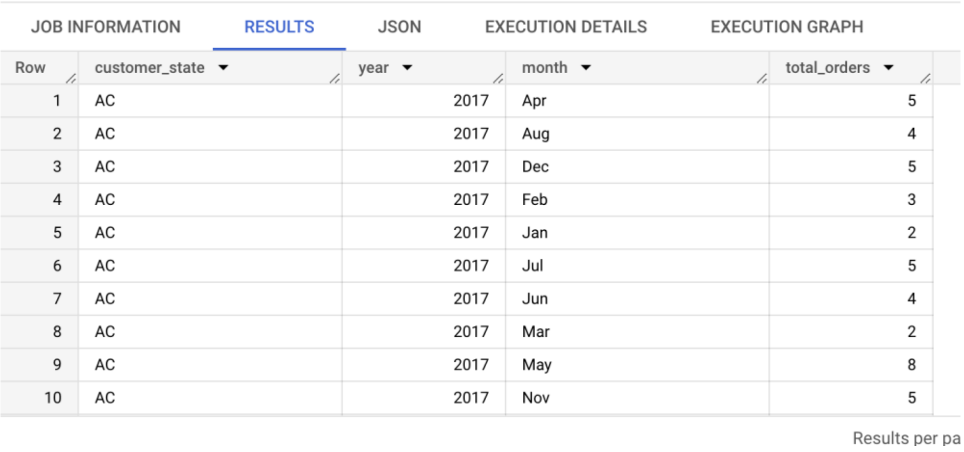
We can see that most of the orders are placed in Afternoon time and the Dawn is the time where people are less interested to order.

3)1 Get the month on month no. of orders placed in each state.

Query:

```
select
                    c.customer_state,
                    extract (year from order_purchase_timestamp) as year,
                    format_date('%b',order_purchase_timestamp) as month,
                    count(order_purchase_timestamp) as total_orders
          from
          `Target.customers` c
          inner join `Target.orders` o on o.customer_id = c.customer_id
          group by 1,2,3
          order by 1,2,3
```

Output:



Row	customer_state	year	month	total_orders
1	AC	2017	Apr	5
2	AC	2017	Aug	4
3	AC	2017	Dec	5
4	AC	2017	Feb	3
5	AC	2017	Jan	2
6	AC	2017	Jul	5
7	AC	2017	Jun	4
8	AC	2017	Mar	2
9	AC	2017	May	8
10	AC	2017	Nov	5

Insights:

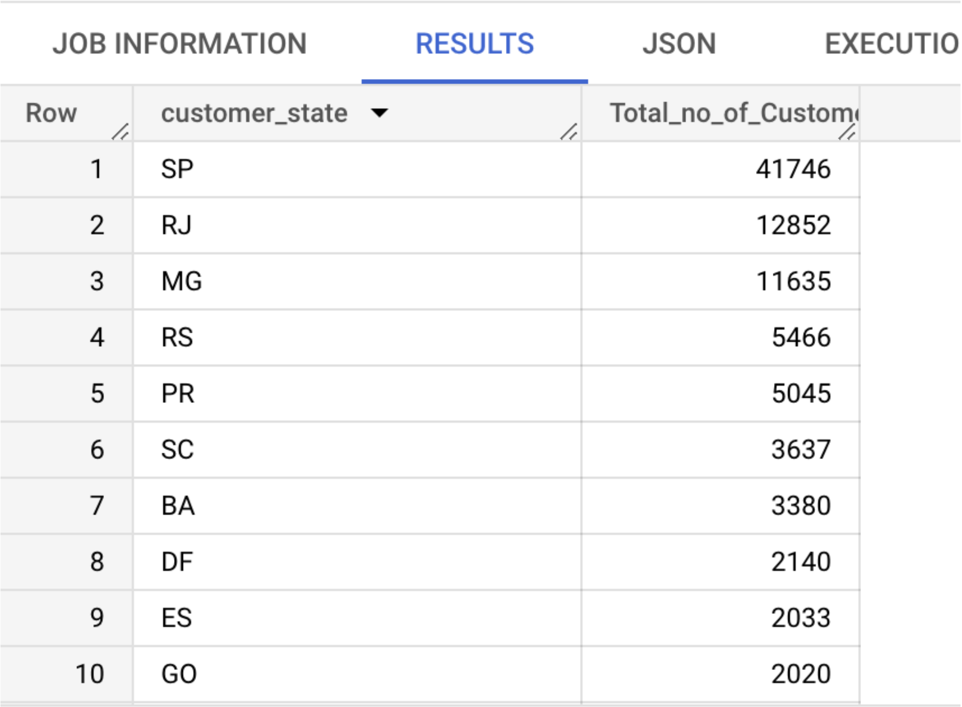
With this we can see the total number of orders that are placed in each month and in each year.

3)2 How are the customers distributed across all the states?

Query :

```
select
                customer_state,
                count(customer_id) as Total_no_of_Customers
        from
        `Target.customers`
        group by 1
        order by 2 desc
```

Output:



Row	customer_state	Total_no_of_Custom
1	SP	41746
2	RJ	12852
3	MG	11635
4	RS	5466
5	PR	5045
6	SC	3637
7	BA	3380
8	DF	2140
9	ES	2033
10	GO	2020

Insights:

With this we can see most of our customers lies in the SP,RJ,MG states.

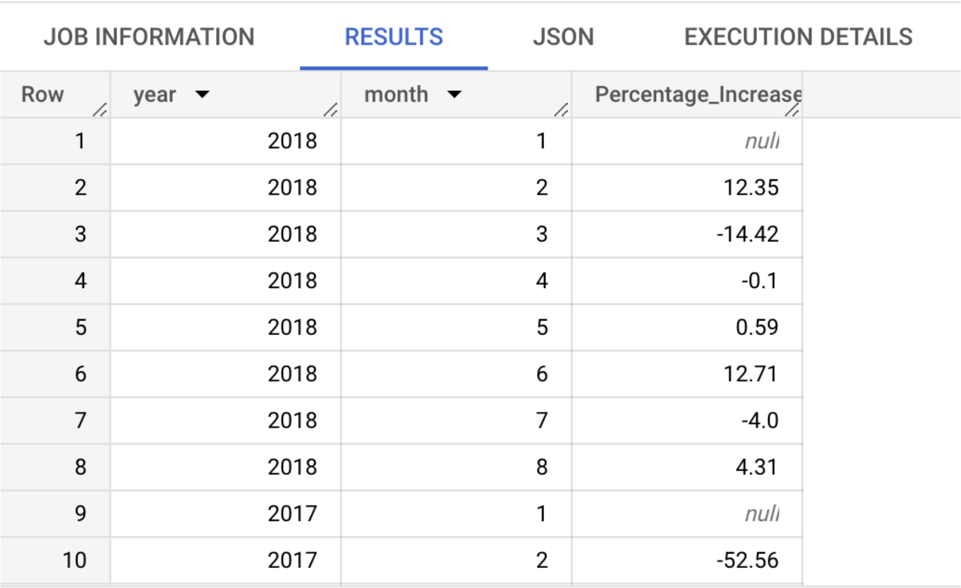
And our focus should be on stares that are in the below.

4)1) Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only).

Query:

```
select
                year,
                month,
                round(safe_divide((lag(tab.tot_value) over (partition by year order by year,month)-
                tab.tot_value),tab.tot_value) * 100,2) as Percentage_Increase
        from
                (
                select
                round(sum(payment_value),2) as tot_value,
                extract (year from order_purchase_timestamp) year,
                extract (month from order_purchase_timestamp) month
                from
                `Target.orders` o
                inner join `Target.payments` p on p.order_id = o.order_id
                group by 2,3
                ) as tab
where year in (2017,2018) and
month between 1 and 8
order by year , month
```

Output:



Row	year	month	Percentage_Increase
1	2018	1	*null*
2	2018	2	12.35
3	2018	3	-14.42
4	2018	4	-0.1
5	2018	5	0.59
6	2018	6	12.71
7	2018	7	-4.0
8	2018	8	4.31
9	2017	1	*null*
10	2017	2	-52.56

Insights:

We can the % increase of payment_value by year and by month

4)2) Calculate the Total & Average value of order price for each state.

Query:

```
select
             c.customer_state,
             round(sum(price),2) as Total_Price,
             round(avg(price),2) as Average_Price
      from `Target.order_items` oi
      inner join `Target.orders` o on oi.order_id = o.order_id
      inner join `Target.customers` c on o.customer_id = c.customer_id
      group by 1
      order by 2 desc, 3 desc
```

Output:

Insights:
We can see the total price value and average price value of orders for each state.

4)3 Calculate the Total & Average value of order freight for each state.

Query:

```
select
             c.customer_state,
             round(sum(freight_value),2) as Total_freight_value,
             round(avg(freight_value),2) as Average_Freight_value
      from `Target.order_items` oi
      inner join `Target.orders` o on oi.order_id = o.order_id
      inner join `Target.customers` c on o.customer_id = c.customer_id
      group by 1
      order by 2 desc, 3 desc
```

Output:

Insights:
We can see total freight and average freight for each state.

5)1) Find the no. of days taken to deliver each order from the order’s purchase date as delivery time.

Query:

```
select order_id,customer_id,
             date_diff(order_delivered_customer_date,order_purchase_timestamp,Day)
             time_to_deliver,
             date_diff(order_estimated_delivery_date,order_delivered_customer_date,Day
             ) Diff_estimated_Delivery
      from `Target.orders`
      order by 3 desc,4
```

```
Output:
```

Insights:
We can see how many days an order took to deliver the product and along with the estimated delivery date.

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

Query:
---- for the Top 5--------

```
select
             c.customer_state,
             round(avg(freight_value) ,2) as avg_freight_value
      from `Target.order_items` oi
      inner join `Target.orders` o on o.order_id = oi.order_id
      inner join `Target.customers` c on o.customer_id = c.customer_id
      group by 1
      order by 2 desc
      limit 5

      ------- for the bottom 5 -------
      select
             c.customer_state,
             round(avg(freight_value),2) as avg_freight_value
      from `Target.order_items` oi
      inner join `Target.orders` o on o.order_id = oi.order_id
      inner join `Target.customers` c on o.customer_id = c.customer_id
      group by 1
      order by 2
      limit 5
```

Output:

Insights:

We can see the Top 5 and Bottom 5 average freight values for each state.

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

Query:

----------- Bottom 5-----------------

```
        select
                c.customer_state,
                round(avg(date_diff(order_delivered_customer_date,order_purchase_timestamp,Day))) as avg_time
        from `Target.orders` o
        inner join `Target.customers` c on c.customer_id = o.customer_id
        where order_status = 'delivered'
        group by 1
        order by 2 desc
        limit 5

-------------------- Top 5 ------------------

select
                c.customer_state,
                round(avg(date_diff(order_delivered_customer_date,order_purchase_timestamp,Day))) as avg_time
        from `Target.orders` o
        inner join `Target.customers` c on c.customer_id = o.customer_id
        where order_status = 'delivered'
        group by 1
        order by 2
        limit 5



Output:
```

Insights:

We can see the Top 5 states with average delivery time and
Bottom 5 states with least delivery time

5)4 Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery.

Query:

```
select
                c.customer_state,
                round(avg(date_diff(order_estimated_delivery_date,order_delivered_custome
                r_date,Day))) as avg_time
        from `Target.orders` o
        inner join `Target.customers` c on c.customer_id = o.customer_id
        where order_status = 'delivered'
        group by 1
        order by 2 desc
        limit 5
```

```
Output ;
```

Insights:

We can these are the Top 5 states which are given fast deliveries

6)1 Find the month on month no. of orders placed using different payment types.

Query:

```
select
        payment_type,
        extract(year from order_purchase_timestamp) as Year,
        extract(month from order_purchase_timestamp) as month,
        count(o.order_id) as total_orders
from
`Target.payments` p
inner join `Target.orders` o on o.order_id = p.order_id
group by 1,2,3
order by 4 desc
```

Output:

Insights:

We can see which payment_type method was the most used by the customers.
We can dig more by looking into Month and Year data

6)2) Find the no. of orders placed on the basis of the payment installments that have been paid.

Query:

```
select
              payment_installments,
              count(o.order_id) as total_orders
       from `Target.payments` p
       inner join `Target.orders` o on o.order_id = p.order_id and payment_value <> 0
       group by 1
       order by 2 desc
```

Output:

Insights:
Here we can see that on the installment 1 there are more orders placed.

Recommendations:

>> We see that at the Dawn the orders are less if possible we can just start considering at that time and see how we can make customers order in that time.

>> In Q1 and Q4 the month on month sales are less this is the gap where we can make c
ustomers order in those months.

>>There are few states that are taking so much of time to deliver products to them we better look into those states and see what's causing the delay