Business Case: Target SQL

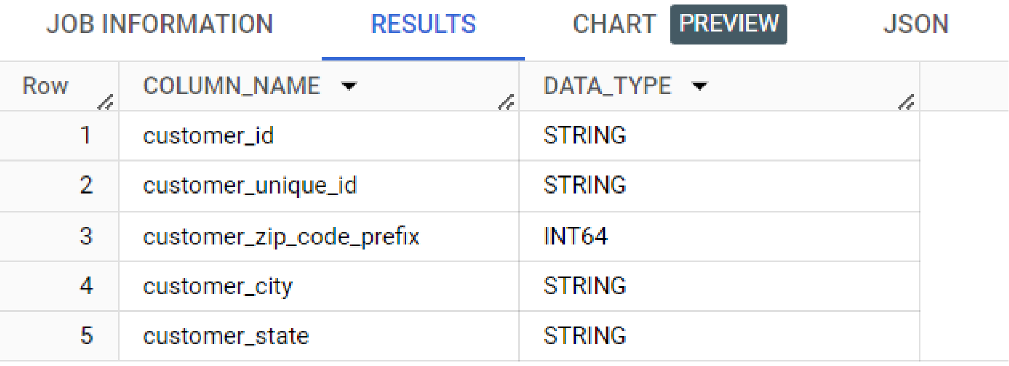
Answer

Q1: **Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset:**

```
#Q1.1: Data type of all columns in the "customers" table.

SELECT
  COLUMN_NAME, DATA_TYPE
FROM
  industrial-keep-408812.target_datas.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:

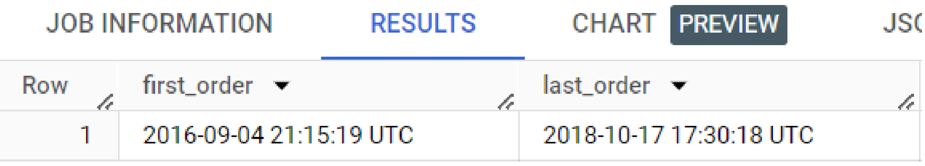
- Here most of the columns are saved as "STRING" except customer_zip_code_prefix. Which is saved as "INTEGER".

Recommendation: NA

```
#Q1.2: 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
  `industrial-keep-408812.target_datas.orders`
```

Output:



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

Insights:

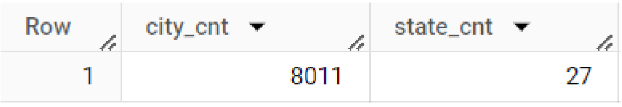
- Target company orders are was placed between these two timestamps. These two orders came in consecutive months but different years.
- To analyse the trends and overall pattens, it can be more helpful to know the time range

Recommendation: NA

```
#Q1.3: Count the Cities & States of customers who ordered during the
given period

SELECT
  COUNT(DISTINCT geolocation_city) as city_cnt,
  COUNT(DISTINCT geolocation_state) as state_cnt
FROM
  `industrial-keep-408812.target_datas.geolocation`
```

Output:



Row	city_cnt	state_cnt
1	8011	27

Insights:

- Here there are 27 distinct states and 8011 distinct cities which placed between this time period, by observing the data which state that Target company is reaching most of state in Brazil.

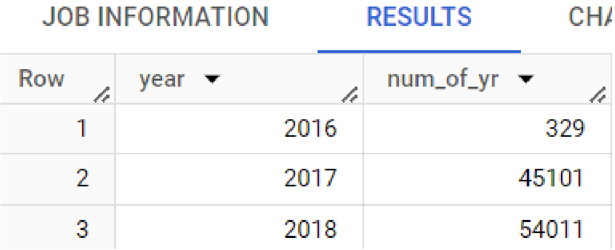
Recommendation:

- This company needs to focus on other areas specially on other cities which are left over and need to do more advertisements to attract more customers.

Q2: In-depth Exploration:

```
#Q2.1: Is there a growing trend in the no. of orders placed over the
past years?
SELECT
  EXTRACT(YEAR FROM order_purchase_timestamp) as year,
  count(order_id) as num_of_yr
from
  `industrial-keep-408812.target_datas.orders`
group by
  year
order by
  year
```

Output:



Row	year	num_of_yr
1	2016	329
2	2017	45101
3	2018	54011

Insights:

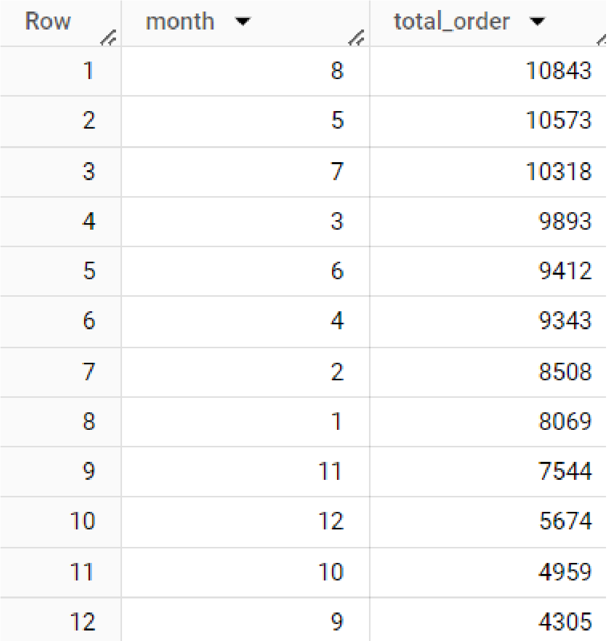
- After examining the result, we can say that there has been an upward trend in the number of orders over past years. The company doing fabulous job.

Recommendation:

- To continue this upward trend, company need to increase its employees and staff to manage this high traffic

```
#Q2.2: Can we see some kind of monthly seasonality in terms of the
no. of orders being placed
SELECT
  month,
  count(order_id) as total_order
FROM
    (select
        *,
        EXTRACT(MONTH FROM order_purchase_timestamp) as month
      from `industrial-keep-408812.target_datas.orders`
    ) as n_t
group by month
order by total_order DESC
```

Output:



Row	month	total_order
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
11	10	4959
12	9	4305

Insights:

- Here between May and August there is a huge increase in the orders placed as it's a period of festivals and in the first quarter the number of order places is quite good but in the last quarter of year the orders placed is less than other quarters. So, company needs to give more effort in this time periods.

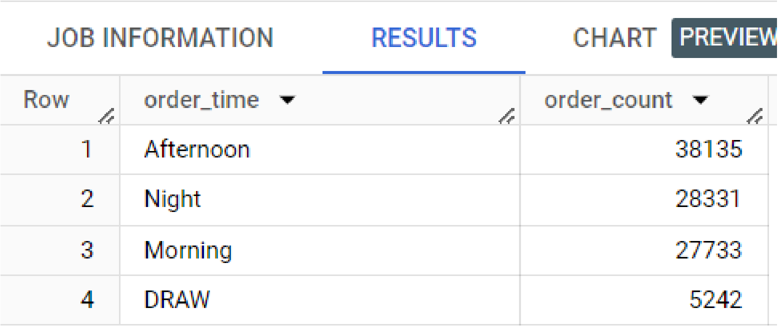
Recommendation:

- By keeping this result in view, company need to focus on seasonality products in stock which will play vital role in increasing sales. Company also needs to take advantage of this situation to increase the non-seasonality products by giving combo pack with seasonality products by providing some discounts on combo packs and company needs to do more advertisements in the last quarter.

```
/*Q2.3:
During what time of the day,do the Brazilian customers mostly place
their orders?
(Dawn, Morning, Afternoon,Night)
*/

SELECT
  CASE WHEN EXTRACT(HOUR FROM order_purchase_timestamp) between 0
and 6 then "DRAW"
       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 order_time,
  count(order_id) as order_count
from
  `industrial-keep-408812.target_datas.orders`
group by order_time
order by order_count desc
```

Output:



Row	order_time	order_count
1	Afternoon	38135
2	Night	28331
3	Morning	27733
4	DRAW	5242

Insights:

- By analyzing the data, we can say that Brazilian customers mostly placed their orders in Afternoon followed by night and morning where there is a rapid decrease in Dawn because most of the people were in sleep and inactive or doing some other daily activities.

Recommendation:

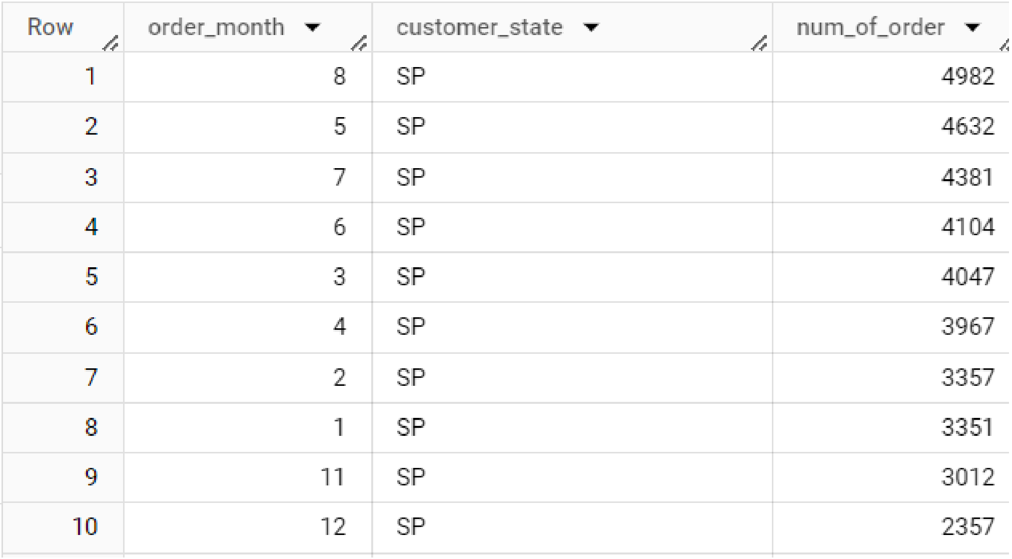
- By lunching forced marketing efforts during the busiest ordering period, it can help to increase numbers of order placed and Target company needs to make availability of more staff in company at afternoon to manage the heavy traffic.

Q3: Evolution of E-commerce orders in the Brazil region:

```
#Q3.1: Get the month on month no. of orders placed in each state.

SELECT
  EXTRACT(month from o.order_purchase_timestamp) as order_month,
  c.customer_state, count(*) as num_of_order
FROM
  `target_datas.customers` as c inner join
  `target_datas.orders` as o
  on c.customer_id = o.customer_id
group by
  order_month,c.customer_state
order by
  num_of_order DESC
```

Output:



Row	order_month	customer_state	num_of_order
1	8	SP	4982
2	5	SP	4632
3	7	SP	4381
4	6	SP	4104
5	3	SP	4047
6	4	SP	3967
7	2	SP	3357
8	1	SP	3351
9	11	SP	3012
10	12	SP	2357

Insights:

- From above, state called SP has the highest number of orders followed by RJ state and MG state with second and third highest RR state has lowest order rate when comparing to other states.

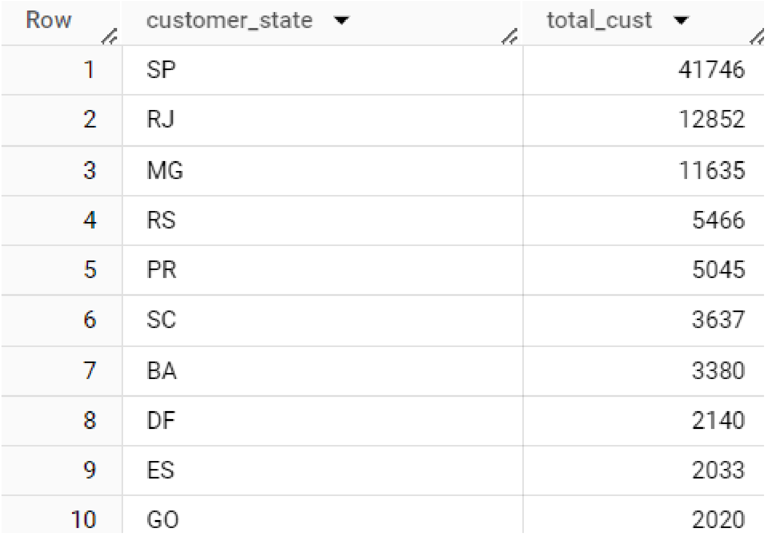
Recommendation:

- Company needs to focus on low order placed areas like RR. So, company needs to make some good marketing strategy like make good offers, give combo product and do some weekly sale to attract more customer.

#Q3.2: How are the customers distributed across all the states?

```
SELECT
  customer_state,
  COUNT(DISTINCT customer_id) as total_cust
from
  `target_datas.customers`
group by
  customer_state
order by
  total_cust desc
```

Output:



Row	customer_state	total_cust
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:

- Here state called SP has the highest customers and state called RR has the fewest customers and there is a huge difference between first highest number of customers and second highest number of customers. So, we can consider that SP state is the most important state for this company.

Recommendation:

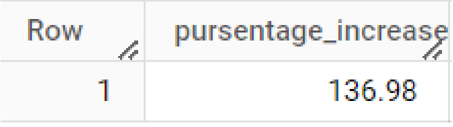
- Target needs to give more focus on least number of customers state like RR, AP, AC as in the following state the customer number is less that 85. Company needs to conduct some kind of survey to know more about the peoples and their requirements.

Q4: Impact on Economy: Analyse the money movement by e-commerce by looking at order prices, freight and others:

```
/*
Q4.1: 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
  ROUND((((total_cost_2018-total_cost_2017)/total_cost_2017)*100),2)
  as pursentage_increase
FROM
  (SELECT
   sum(
       case when extract(year from o.order_purchase_timestamp)=2017
       and
       extract(month from o.order_purchase_timestamp) between 1 and 8
       then payment_value else 0 end
      ) as total_cost_2017,
   sum(
       case when extract(year from o.order_purchase_timestamp)=2018
       and
       extract(month from o.order_purchase_timestamp) between 1 and 8
       then payment_value else 0 end
      ) as total_cost_2018
   FROM
     target_datas.orders` as o

      inner join
     `target_datas.payments` as p
      on o.order_id = p.order_id
   ) as T
```

Output:



Row	pursentage_increase
1	136.98

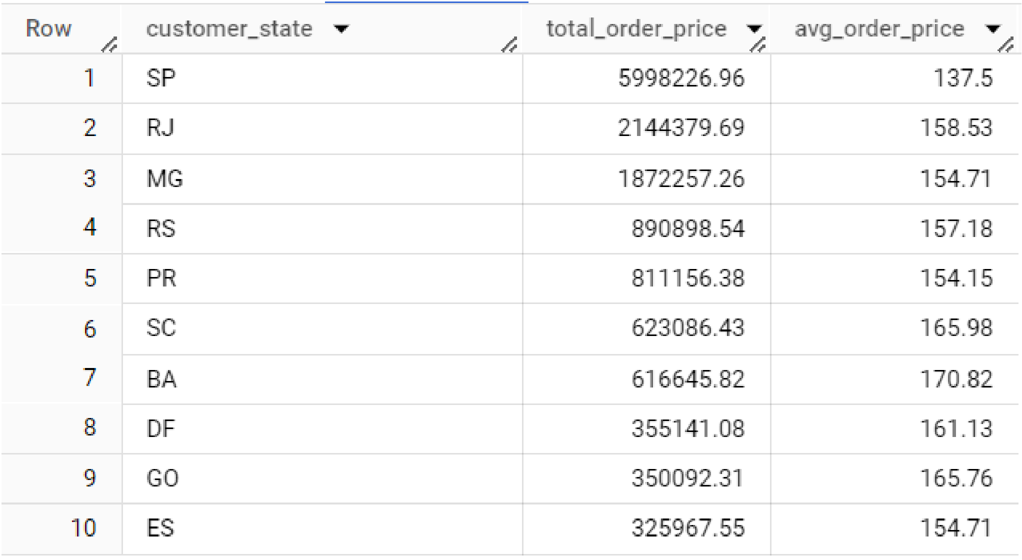
Insights:

- By observing above results there is a growth of approximately 137% from year 2017 and 2018 including months between Jan to Aug. This is a tremendous growth in the year 2018 compare to 2017. Target is doing tremendous work by creating a massive growth in just one year of span. These results not even include last four months but these eight months made a big difference in profit percentage.

Recommendation: NA

```
#Q4.2: Calculate the Total & Average value of order price for each
state:
SELECT
  c.customer_state,
  ROUND(sum(p.payment_value),2) as total_order_price,
  ROUND(avg(p.payment_value),2) as avg_order_price
FROM
  `target_datas.orders` as o
  inner join
  `target_datas.payments` as p
  on o.order_id = p.order_id
  inner join
  `target_datas.customers` as c
  on o.customer_id = c.customer_id
group by
  c.customer_state
order by
  total_order_price desc
```

Output:



Row	customer_state	total_order_price	avg_order_price
1	SP	5998226.96	137.5
2	RJ	2144379.69	158.53
3	MG	1872257.26	154.71
4	RS	890898.54	157.18
5	PR	811156.38	154.15
6	SC	623086.43	165.98
7	BA	616645.82	170.82
8	DF	355141.08	161.13
9	GO	350092.31	165.76
10	ES	325967.55	154.71

Insights:

- Here total value refers sum of the prices of all orders by no of orders. From above results state called SP has highest total order price but it has 137.5 average order price and state called PB has highest average order price. This shows the SP state contributing in large value where RR state has lowest value in terms of total order price.

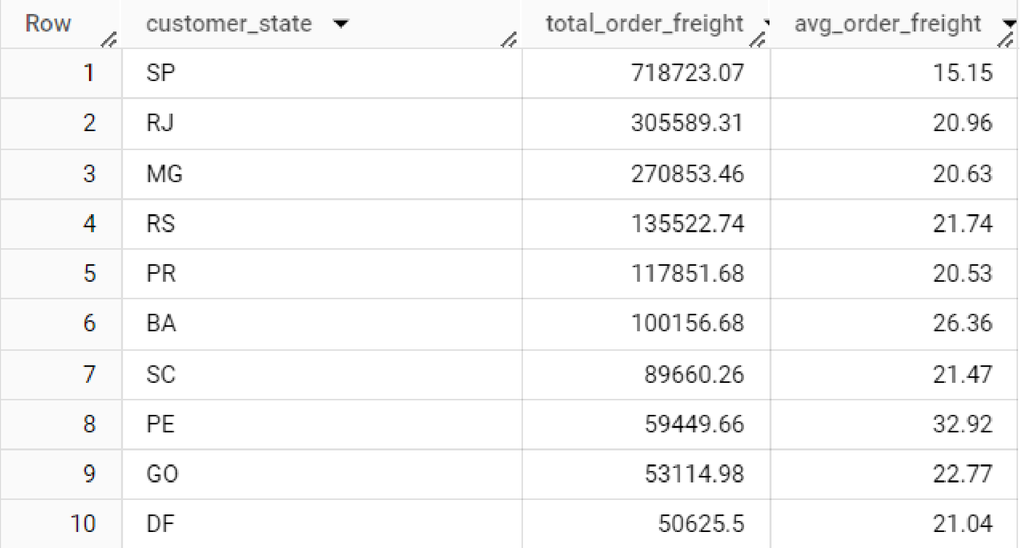
Recommendation:

- Company needs to focus on low level states to generate new customer from these states. It needs to implement some marketing strategies by providing discounts on occasions.

```
#Q4.3: Calculate the Total & Average value of order freight for each
state

SELECT
  c.customer_state,
  ROUND(sum(o_i.freight_value),2) as total_order_freight,
  ROUND(avg(o_i.freight_value),2) as avg_order_freight
FROM
  `target_datas.orders` as o
  inner join
  `target_datas.order_items` as o_i
  on o.order_id = o_i.order_id
  inner join
  `target_datas.customers` as c
  on o.customer_id = c.customer_id
group by
  c.customer_state
order by
  total_order_freight desc
```

Output:



Row	customer_state	total_order_freight	avg_order_freight
1	SP	718723.07	15.15
2	RJ	305589.31	20.96
3	MG	270853.46	20.63
4	RS	135522.74	21.74
5	PR	117851.68	20.53
6	BA	100156.68	26.36
7	SC	89660.26	21.47
8	PE	59449.66	32.92
9	GO	53114.98	22.77
10	DF	50625.5	21.04

Insights:

- Here we can find states with highest total freight costs (freight cost means price for transporting item from one place to other it is like delivery charges).
- Here in this a state called SP having less average order freight but RR has highest average order freight. By this we can say that SP state is near from this company for which freight

cost is less where RR state is very far from this company it leads to increase in cost of freight price.

- Over all we can say that increase freight price is directly proportional to decrease in number of orders.

Recommendation:

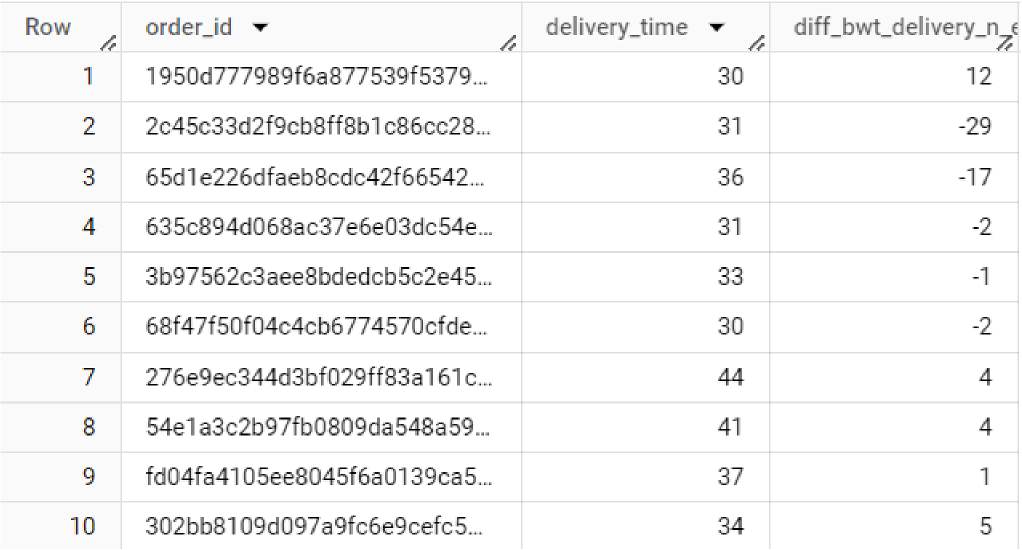
- Company needs to extend its services by expanding their branches in some far away states which leads to increase customers in these states and it will also reduce the long delivering period to short period it will also decrease in average freight values and increase in total freight values.

Q5: Analysis based on sales, freight and delivery time:

```
/* Q5.1:
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,
  DATE_DIFF(DATE(order_delivered_customer_date),
  DATE(order_purchase_timestamp),DAY) AS delivery_time,
  DATE_DIFF(DATE(order_delivered_customer_date),
  DATE(order_estimated_delivery_date),DAY)
    AS diff_bwt_delivery_n_estimate
FROM
  `target_datas.orders`
```

Output:



Row	order_id	delivery_time	diff_bwt_delivery_n_e
1	1950d777989f6a877539f5379...	30	12
2	2c45c33d2f9cb8ff8b1c86cc28...	31	-29
3	65d1e226dfaeb8cdc42f66542...	36	-17
4	635c894d068ac37e6e03dc54e...	31	-2
5	3b97562c3aee8bdedcb5c2e45...	33	-1
6	68f47f50f04c4cb6774570cfde...	30	-2
7	276e9ec344d3bf029ff83a161c...	44	4
8	54e1a3c2b97fb0809da548a59...	41	4
9	fd04fa4105ee8045f6a0139ca5...	37	1
10	302bb8109d097a9fc6e9cefc5...	34	5

Insights:

- By observing the results, from above table the second column indicates the number of days taken to deliver the order and third column indicates that difference between order delivered date and order estimated date. We can see both positive and negative values occurs in third column, positive values states that order is delivered after the estimated date and negative values state that order is delivered before to the estimated value.

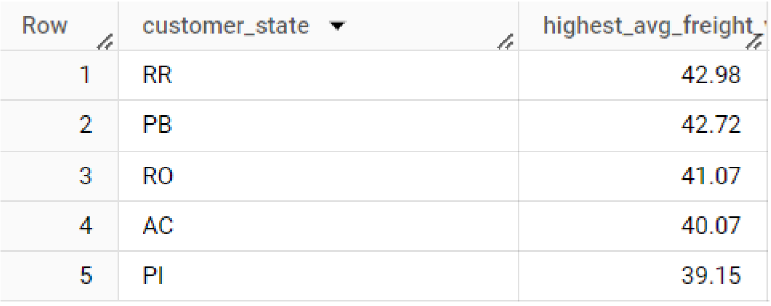
Recommendation:

- Here as we can see some orders time taken to deliver the order is very long which might cause for lose of customer. So, company needs do fast delivery and if for some random reason delivery is delay then company give some voucher kind of things by which customer interest should not decrease.

```
#Q5.2.1: Find out the top 5 states with the highest average freight
value

SELECT
  c.customer_state,
  ROUND(AVG(oi.freight_value),2) as highest_avg_freight_value
FROM
  `target_datas.orders` as o
  join `target_datas.order_items` as oi
  on o.order_id=oi.order_id
  join `target_datas.customers` as c
  on o.customer_id = c.customer_id
group by
  c.customer_state
order by
  highest_avg_freight_value desc
limit
  5
```

Output:

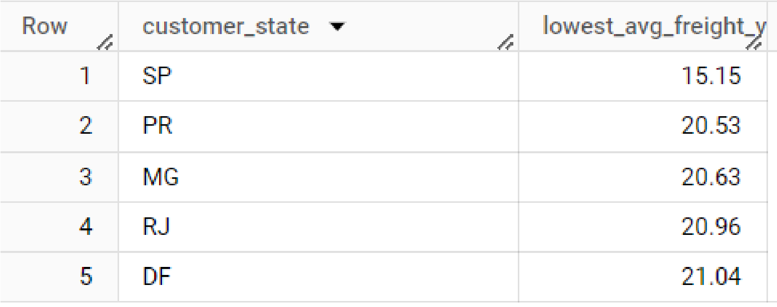


Row	customer_state	highest_avg_freight_
1	RR	42.98
2	PB	42.72
3	RO	41.07
4	AC	40.07
5	PI	39.15

```
 #Q5.2.2: Find out the top 5 states with the lowest average freight
value

SELECT
  c.customer_state,
  ROUND(AVG(oi.freight_value),2) as lowest_avg_freight_value
FROM
  `target_datas.orders` as o
  join `target_datas.order_items` as oi
  on o.order_id=oi.order_id
  join `target_datas.customers` as c
  on o.customer_id = c.customer_id
group by
  c.customer_state
order by
  lowest_avg_freight_value
limit
  5
```

Output:



Row	customer_state	lowest_avg_freight_y
1	SP	15.15
2	PR	20.53
3	MG	20.63
4	RJ	20.96
5	DF	21.04

Insights:

- By combining Q5.2.1 and Q5.2.2, we have a table that represents top five highest average freight value and top five lowest average freight values. By the results RR state has the highest average value and SP state has lowest average freight value.

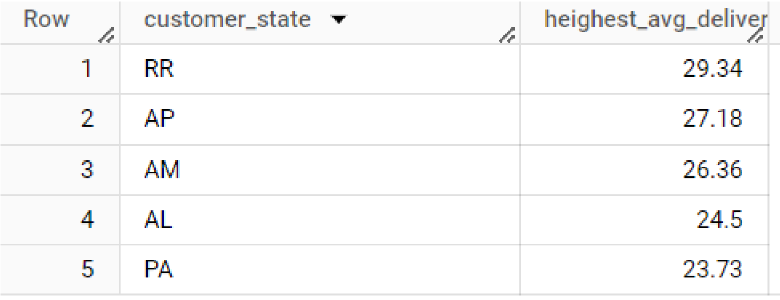
Recommendation:

- By observing above resulting table, the states which has highest average freight value are the states which are far away from the orders pickup place. Which will be the major cause for increasing freight values this indirectly affecting the customers who place want to place order but not placing any order. Where lowest freight values states are short distance from order pickup place this will not affect the customers. Company should focus on this issue because there are very a smaller number of customers from highest average freight values, so company should look alternative to overcome this by providing services from short distances by expanding their branches.

#Q5.3.1: Find out the top 5 states with the highest average delivery time

```
SELECT
  c.customer_state,
  ROUND(AVG(t.delivery_time),2) as heighest_avg_delivery_time
FROM
  (
    SELECT
      *,
      DATE_DIFF(DATE(order_delivered_customer_date),
      DATE(order_purchase_timestamp),DAY) AS delivery_time
    FROM
      `target_datas.orders`
    WHERE
      order_status = "delivered" and
      order_delivered_customer_date is not null
    order by
      order_purchase_timestamp
  ) as t
  JOIN
  `target_datas.customers` as c
  on t.customer_id=c.customer_id
group by
  c.customer_state
order by
  heighest_avg_delivery_time desc
limit
  5
```

Output:

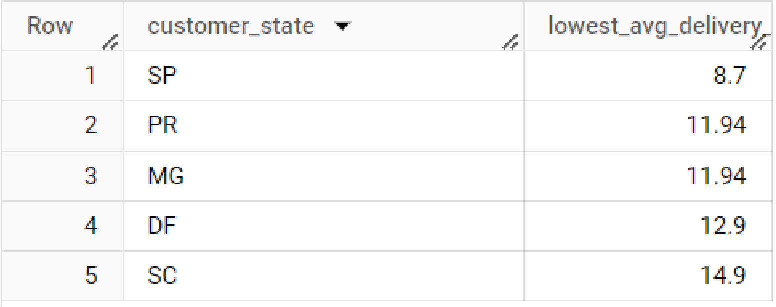


Row	customer_state	heighest_avg_deliver
1	RR	29.34
2	AP	27.18
3	AM	26.36
4	AL	24.5
5	PA	23.73

#Q5.3.2: Find out the top 5 states with the highest average delivery time

```
SELECT
  c.customer_state,
  ROUND(AVG(t.delivery_time),2) as lowest_avg_delivery_time
FROM
  (
    SELECT
      *,
      DATE_DIFF(DATE(order_delivered_customer_date),
      DATE(order_purchase_timestamp),DAY) AS delivery_time
    FROM
      `target_datas.orders`
    WHERE
      order_status = "delivered" and
      order_delivered_customer_date is not null
    order by
      order_purchase_timestamp
  ) as t
  JOIN
  `target_datas.customers` as c
  on t.customer_id=c.customer_id
group by
  c.customer_state
order by
  lowest_avg_delivery_time
limit
  5
```

Output:



Row	customer_state	lowest_avg_delivery_
1	SP	8.7
2	PR	11.94
3	MG	11.94
4	DF	12.9
5	SC	14.9

Insights:

- By combining Q5.3.1 and Q5.3.2, we have a table that represents top five highest average delivery time and top five lowest average delivery time. So, from this we can say that states like SP, PR and MG are those states where company is taking less time to delivered the product where some states like RR, AP and AM are those states where company is taking long time to delivered the product.

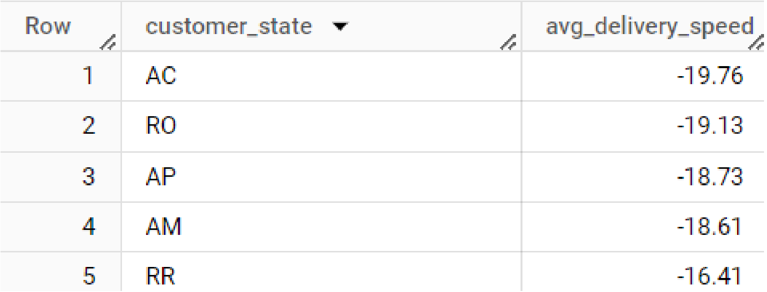
Recommendation:

- By observing above data, delivery delay is indirectly affecting the number of customers due to the long delivery time customers may looking into alternative options this might decrease in customer numbers. So, company need should look into this issue and need to work on on-or-before the estimate delivery date.

```sql
/* Q5.4:
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(avg_delivery_speed,2) as avg_delivery_speed
FROM
  (
    SELECT
      c.customer_state,
      avg(DATE_DIFF(o.order_delivered_customer_date,
      o.order_estimated_delivery_date,DAY)) as avg_delivery_speed
    from
      `target_datas.orders` as o
      join `target_datas.customers` as c
      on o.customer_id = c.customer_id
    where
      o.order_delivered_customer_date is not null and
      o.order_estimated_delivery_date is not null
    group by
      c.customer_state
  ) AS T
ORDER BY
  avg_delivery_speed
LIMIT
  5
```

Output:



Row	customer_state	avg_delivery_speed
1	AC	-19.76
2	RO	-19.13
3	AP	-18.73
4	AM	-18.61
5	RR	-16.41

Insights:

- The company may be operating in these state called AC, RO, AP and AM where the average speed to delivery is highest. Here negative symbol refers that order is delivered prior to the estimated date.

Recommendation:

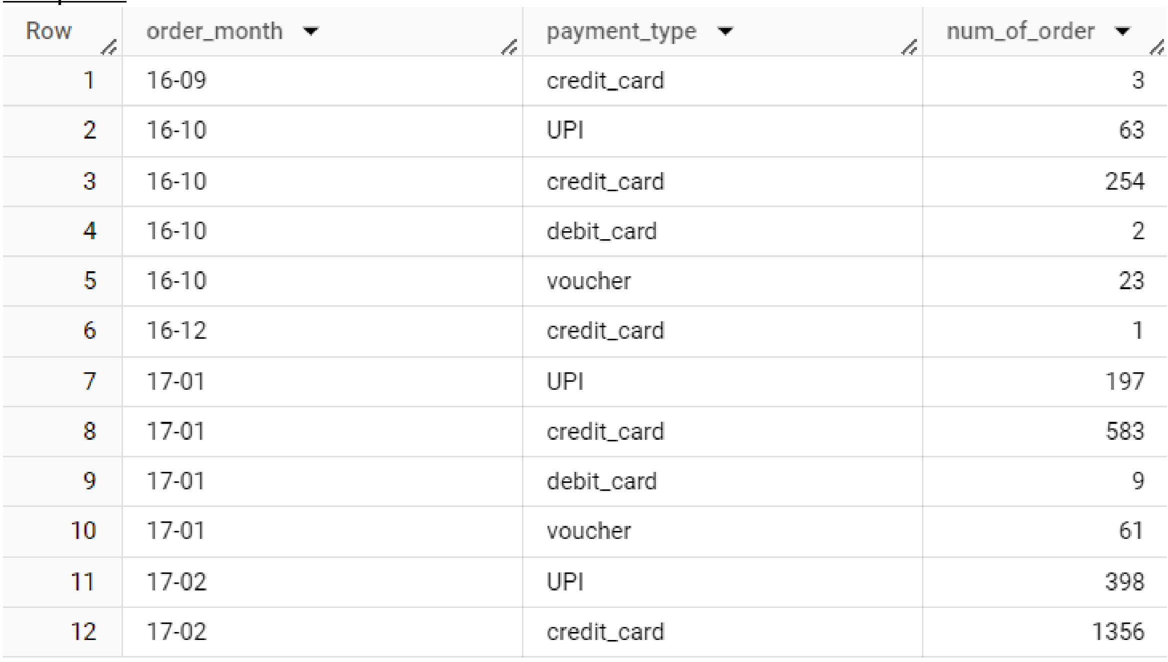
- Target giving its best by delivering its orders prior to the estimated dates, but when we compare individual orders, some orders are getting delay in delivery time. Target company needs to give that much of efforts to delivery to those state where average delivery date is too late.

Q6: Analysis based on the payments:

```
#Q6.1: Find the month on month no. of orders placed using different
payment types

SELECT
  format_timestamp("%y-%m" , o.order_purchase_timestamp) as
order_month,
  p.payment_type,count(*) num_of_order
FROM
  `target_datas.payments` as p
  join `target_datas.orders` as o
  on p.order_id = o.order_id
group by
  order_month,payment_type
order by
  order_month,payment_type
```

Output:



Row	order_month	payment_type	num_of_order
1	16-09	credit_card	3
2	16-10	UPI	63
3	16-10	credit_card	254
4	16-10	debit_card	2
5	16-10	voucher	23
6	16-12	credit_card	1
7	17-01	UPI	197
8	17-01	credit_card	583
9	17-01	debit_card	9
10	17-01	voucher	61
11	17-02	UPI	398
12	17-02	credit_card	1356

Insights:

- By observing above results on month on month most of the orders, here we can see that credit card as a payment method was the most used payment method for purchasing and less peoples used UPI mode as payment method for purchasing.

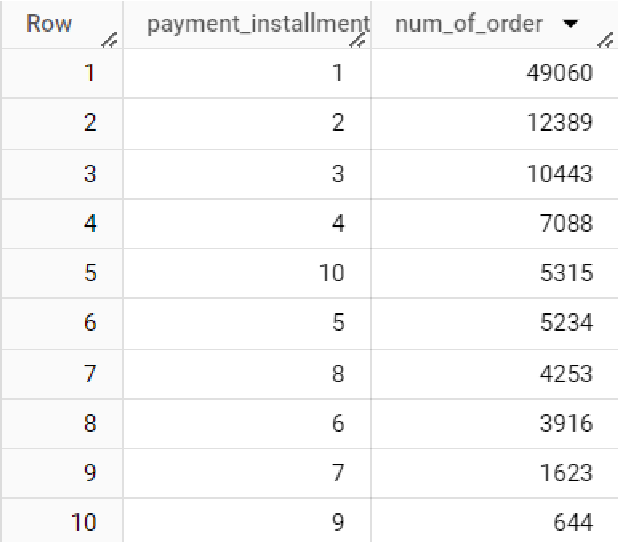
Recommendation:

- The target company need focus on the data because they need to make arrangements for the customers according to their convenient payment methods it makes easy and time saving process for customers, target should provide discounts on particular credit cards by collaborating with that banks which will generate the additional revenue.

```
#Q6.2: Find the no. of orders placed on the basis of the payment
installments that have been paid

SELECT
  payment_installments,count(distinct order_id) as num_of_order
FROM
  `target_datas.payments`
group by
  payment_installments
order by
  num_of_order desc
```

Output:



Row	payment_installment	num_of_order
1	1	49060
2	2	12389
3	3	10443
4	4	7088
5	10	5315
6	5	5234
7	8	4253
8	6	3916
9	7	1623
10	9	644

Insights:

- By observing the above results 49060 orders were placed where payment instalments were 1, only few of them went to two to twenty-four months of instalments process because to avoid extra or additional charges people preferring single payment method.

Recommendation:

- Target should also focus on giving discounts on instalments which will attract more and more customers because most of the people are also prefer instalments when they want buy some big products. By this company may be increase their customers.