TARGET CORPORATION

In this case study I have used BIGQUERY by GOOGLE CLOUD to perform different operation and write queries to gain the desired results. There were total of 8 .csv files containing data of a company TARGET CORPORATION (Brazil chapter). The eight files were as following:

- customers.csv - In this we had all the details of customers such as their unique ID, ID's for the customers who made any purchase, customers location details such as zip code, city and state. All the fields data type were string except zip code which was integer.
- geolocation.csv - These values provide geographical information such as latitude, longitude, city, state and zip code.
- order_items.csv - In this we have information related to sellers like seller id and details of product such as product id and original price of the product and also the freight rate for each product. In this study most used field are price and freight values whose data type is float and order id which is a string.
- payments.csv - In this we have information regarding payments such as type of payment either using UPI, Credit card, Debit card, etc. , instalment option opt for and order id.
- reviews.csv - In this project this section is not used as it provides the feedback from the customers for their order if provided any.
- orders.csv - This is one of the main file containing all the information regarding orders which are placed by customers in the company such as order id, customer id, status of the order, timestamp when order was placed, date of delivery, estimated delivery and carrier date. Among these fields order id, customer id and order status are in string format and rest are in timestamp format.
- products.csv - Although this file isn't used in this study but this file provides all the physical information of the product such as its id and dimensions which is useful to mention on the website and also to provide it to delivery partner.
- sellers.csv - Same like customers.csv this file provides all the information regarding the sellers such as their id, city, state and zip code of their location.

1. **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**
        - **This question is answered above.**

2. Time period for which the data is given

```
SELECT
    min(date(order_purchase_timestamp)) Start_Date,
    max(date(order_purchase_timestamp)) End_Date,
    DATE_DIFF(max(date(order_purchase_timestamp)),
    min(date(order_purchase_timestamp)), day) Number_of_Days
FROM
    `kartik-scaler.ProjectSCALER.orders`;
```

Row	Start_Date	End_Date	Number_of_Days
1	2016-09-04	2018-10-17	773

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

```
SELECT
    distinct cust.customer_state, cust.customer_city
FROM
    `kartik-scaler.ProjectSCALER.customers` cust
JOIN
    `kartik-scaler.ProjectSCALER.orders` ord
ON cust.customer_id = ord.customer_id;
```

- Customers from around 4,310 distinct cities from 27 States have ordered products from Target Corporation from 04 Sep 2016 to 17 Oct 2018

Row	customer_state	customer_city
1	RN	acu
2	CE	ico
3	RS	ipe

2. **In-depth Exploration:**

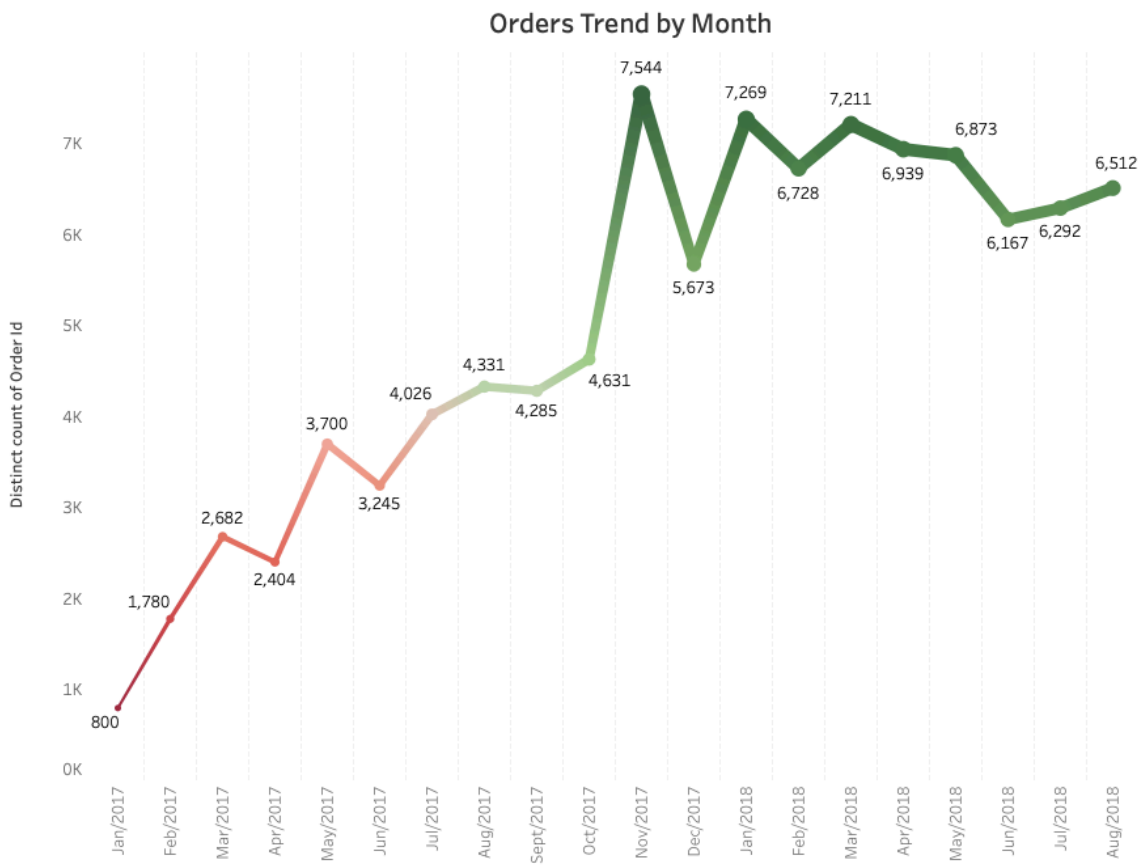
    1. **Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario? Can we see some seasonality with peaks at specific months?**

```
SELECT
        YEAR, MONTH, COUNT(*) MoM_Sales_Number
FROM
      (
      SELECT EXTRACT(Year FROM order_purchase_timestamp) YEAR,
      EXTRACT(Month FROM order_purchase_timestamp) MONTH
      FROM `kartik-scaler.ProjectSCALER.orders`
      )
GROUP BY
      YEAR, MONTH
ORDER BY
      YEAR, MONTH;
```

```
--OR


SELECT
      EXTRACT ( Year FROM order_purchase_timestamp ) YEAR,
      EXTRACT ( Month FROM order_purchase_timestamp ) MONTH,
      COUNT (*)
FROM
       `kartik-scaler.ProjectSCALER.orders`
GROUP BY
      YEAR, MONTH
ORDER BY
      YEAR, MONTH;
```

- It's odd to have very low numbers in Sep 2016, Dec 2016 and 1-2 months of 2018, maybe during this time data was not collected efficiently / Target Corporation had started its new chain due to which it took around 3 - 4 months to gain good no of customers in Brazil market. But after Feb 2017 till Aug 2018 their sales have been good in numbers.

Row	YEAR	MONTH	MoM_Sales_Number
1	2016	9	4
2	2016	10	324
3	2016	12	1
4	2017	1	800
5	2017	2	1780



- As per the given data there were no trends to be observed or seasonality at peaks of some specific month. But as its shown in graph the trend has grown gradually from Jan 2017 and went to the peak in November 2017 and there was sudden dip in December 2017 and did recovery in Jan 2018. The sudden boom toward November 2017 was maybe due to Brazilian Proclamation Day. In Brazil November is the second and only month in which there is 6 holidays after January. And there is a Festival called "The Carnival of Brazil" which is one of the most famous festival in Brazil in this there is one week long party around the nation. This festival was around 08 Feb 2018 which might have caused increase in sales around January 2018.
  Target corporation can promote some offers one month before festival season to increase their sales and gain maximum profit around this time of the year.

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

```
SELECT Time_of_Day, COUNT(*) Total_Order

FROM ( SELECT

        CASE

            WHEN EXTRACT ( hour FROM order_purchase_timestamp ) between 18 and 23
```

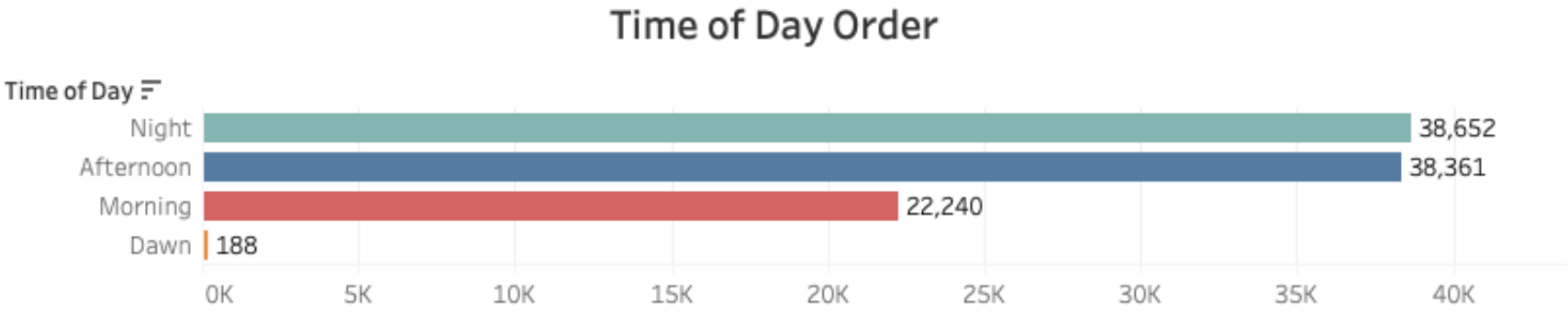
```
            THEN 'Night'
                WHEN EXTRACT ( hour FROM order_purchase_timestamp ) between 12 and 17
            THEN 'Afternoon'
                WHEN EXTRACT ( hour FROM order_purchase_timestamp ) between 6 and 11
            THEN 'Morning'
                WHEN EXTRACT ( hour FROM order_purchase_timestamp ) = 5
            THEN 'Dawn'
                ELSE 'Night'
            END Time_of_Day
        FROM
            `kartik-scaler.ProjectSCALER.orders` )
GROUP BY Time_of_Day;
```

- Brazilian public tend to by more products during Night time.

In this we have considered following time slots as :

- 6 pm to 5 am - Night
- 5 am to 6 am - Dawn
- 6 am to 12 noon - Morning
- 12 noon to 6 pm - Afternoon

Row	Time_of_Day	Total_Order
1	Morning	22240
2	Night	38652
3	Afternoon	38361
4	Dawn	188



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

1. Get month on month orders by states

```
SELECT

    cust.customer_state,

    EXTRACT ( Month FROM odr.order_purchase_timestamp ) Month,

    EXTRACT ( Year FROM odr.order_purchase_timestamp ) Year,

    COUNT (odr.order_purchase_timestamp) No_of_Orders

FROM

    `kartik-scaler.ProjectSCALER.customers` cust

JOIN

     `kartik-scaler.ProjectSCALER.orders` odr

ON cust.customer_id = odr.customer_id

GROUP BY

    cust.customer_state , Month, Year

ORDER BY

    cust.customer_state, Month, Year, No_of_Orders;
```

- In this we got an output for each 27 states comparing their performance from 2017 to 2018 month wise and also with the next month. This provides better readability to compare the No. Of orders in a particular month with another or with the same but different year.

Row	customer_state	Month	Year	No_of_Orders
1	AC	1	2017	2
2	AC	1	2018	6
3	AC	2	2017	3
4	AC	2	2018	3

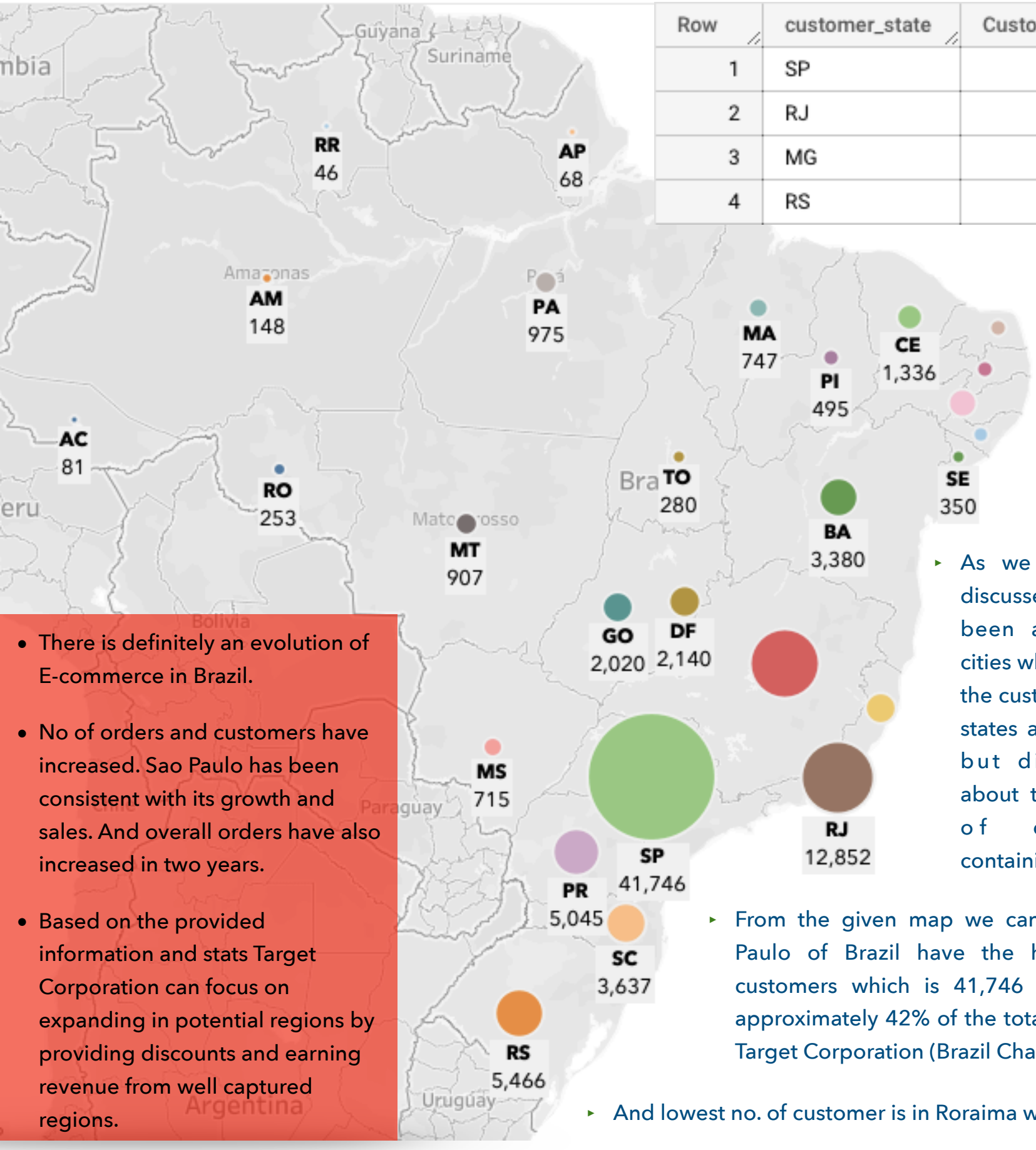
2. Distribution of customers across the states in Brazil

```
SELECT

    distinct customer_state, COUNT(*) Customers_in_State

FROM
```

```
    `kartik-scaler.ProjectSCALER.customers`
GROUP BY
    customer_state
ORDER BY
    Customers_in_State desc;
```

Distribution of customers across the states

Row	customer_state	Customers_in_State
1	SP	41746
2	RJ	12852
3	MG	11635
4	RS	5466



- There is definitely an evolution of E-commerce in Brazil.
- No of orders and customers have increased. Sao Paulo has been consistent with its growth and sales. And overall orders have also increased in two years.
- Based on the provided information and stats Target Corporation can focus on expanding in potential regions by providing discounts and earning revenue from well captured regions.

- As we have already discussed, there had been around 4,310 cities which are having the customers from 27 states all across Brazil but didn't talked about the highest no of customers containing state.
- From the given map we can see that Sao Paulo of Brazil have the highest no of customers which is 41,746 not exact but approximately 42% of the total customers of Target Corporation (Brazil Chapter).
- And lowest no. of customer is in Roraima with 46 of them.

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**

```
SELECT

    tb1.Month_2017 Month_of_Year, ROUND ( tb1.Cost_of_Order,2 ) Cost_of_Order_2017,

    ROUND ( tb2.Cost_of_Order, 2 ) Cost_of_Order_2018, ROUND ( tb2.Cost_of_Order

    tb1.Cost_of_Order, 2 )

    Percent_Increase_in_Cost

FROM

    (SELECT EXTRACT(month FROM odr.order_purchase_timestamp) Month_2017,

    sum(py.payment_value) -

    sum(odrit.price + odrit.freight_value) Cost_of_Order

FROM

    `kartik-scaler.ProjectSCALER.orders` odr

JOIN

    `kartik-scaler.ProjectSCALER.payments` py

ON odr.order_id = py.order_id

JOIN

    `kartik-scaler.ProjectSCALER.order_items` odrit

ON py.order_id = odrit.order_id

WHERE

    EXTRACT ( month FROM odr.order_purchase_timestamp ) between 1 and 8 and EXTRACT ( Year
FROM

    odr.order_purchase_timestamp ) = 2017

GROUP BY

    Month_2017

ORDER BY
```

```
        Month_2017) tb1
JOIN
        ( SELECT EXTRACT ( month FROM odr.order_purchase_timestamp ) Month_2018, sum
        ( py.payment_value )  - sum(odrit.price + odrit.freight_value) Cost_of_Order
FROM
        `kartik-scaler.ProjectSCALER.orders` odr
JOIN
        `kartik-scaler.ProjectSCALER.payments` py
ON odr.order_id = py.order_id
JOIN
        `kartik-scaler.ProjectSCALER.order_items` odrit
ON py.order_id = odrit.order_id
WHERE
        EXTRACT ( month FROM odr.order_purchase_timestamp ) between 1 and 8 and EXTRACT ( Year
FROM
        odr.order_purchase_timestamp ) = 2018
GROUP BY
        Month_2018
ORDER BY
        Month_2018) tb2
ON tb1.Month_2017 = tb2.Month_2018
ORDER BY
        Month_of_Year;
```

- We can clearly observe that there is an increase in cost from the previous year.
- This is possibly due to an inflation or tax policies of the Brazilian government.

Row	Month_of_Year	Cost_of_Order_2017	Cost_of_Order_2018	Percent_Increase_in_Cost
1	1	41323.87	257228.34	6.22
2	2	41904.42	278265.94	6.64
3	3	69179.37	271056.81	3.92
4	4	56922.75	298238.49	5.24

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

```
SELECT
    cust.Customer_State, ROUND (sum(ordit.freight_value),2) Total_Freight, ROUND
    (avg(ordit.freight_value),2) Mean_Freight, ROUND (sum(ordit.price),2)Total_Price,
     ROUND (avg(ordit.price),2) Mean_Price
FROM
    `kartik-scaler.ProjectSCALER.orders` ord
LEFT  JOIN
    `kartik-scaler.ProjectSCALER.order_items` ordit
ON ord.order_id = ordit.order_id
JOIN
    `kartik-scaler.ProjectSCALER.customers` cust
ON ord.customer_id = cust.customer_id
GROUP BY
    cust.customer_state
ORDER BY
    Mean_Price desc;
```

- The table shows the Total freight cost, total products cost, mean freight amount and mean product price amount per state.
- This helps to understand and compare the cost of freight for different states of Brazil which will help to make the decision regarding reducing the freight price by changing the delivery partner of location.

Row	Customer_State	Total_Freight	Mean_Freight	Total_Price	Mean_Price
1	PB	25719.73	42.72	115268.08	191.48
2	AL	15914.59	35.84	80314.81	180.89
3	AC	3686.75	40.07	15982.95	173.73
4	RO	11417.38	41.07	46140.64	165.97

- Above two tables shows that there has been an increase in the cost of products and freight charges in two years. This can be good as we can assume that the Brazilian economy is growing which can be beneficial for the E-Commerce platform like Target Corporation
- But at the same time Target should consider the increase in the cost as concern too as it can reduce their profit margin. For this they should consider multiple options for reducing their product cost and freight cost so that they can increase their profitability by maintaining the trust among their customers.

5. Analysis on sales, freight and delivery time

1. Calculate days between purchasing, delivering and estimated delivery

```
SELECT
    Date(order_purchase_timestamp) Purchase_Date,
    DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY)
    Actual_Delivery_Days,
    DATE_DIFF(order_estimated_delivery_date,order_purchase_timestamp, DAY)
    Estimated_Delivery_Days
FROM
    `kartik-scaler.ProjectSCALER.orders`
WHERE
    order_delivered_customer_date is Not null
ORDER BY
    Purchase_Date;
```

Row	Purchase_Date	Actual_Delivery_Days	Estimated_Delivery_Days
1	2016-09-15	54	18
2	2016-10-03	35	52
3	2016-10-03	24	34
4	2016-10-03	30	58

2. Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:

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

```
SELECT
    Date(order_purchase_timestamp) Purchase_Date,
    DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) Days_to_Delivery,
    DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date, DAY)
    Difference_Estimated_and_Delivery_Days
```

```
FROM
    `kartik-scaler.ProjectSCALER.orders`
WHERE
    order_delivered_customer_date is Not null
ORDER BY
    Purchase_Date;
```

Row	Purchase_Date	Days_to_Delivery	Difference_Estimated_and_Delivery_Days
1	2016-09-15	54	-36
2	2016-10-03	35	16
3	2016-10-03	24	10
4	2016-10-03	30	27

3. Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

VIEW is used in this identification as we need to drive multiple results in following statements from 5.4 onward.
Also the formula provided for the calculation of Time to deliver and difference between estimated delivery days and delivery days is used in the query.

***Also statement 5.4 was incorrectly numbered so the revised numbering has been provided.**

Formula provided and used:

time_to_delivery = order_purchase_timestamp-order_delivered_customer_date

diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date

Solution:

```
Create VIEW ProjectSCALER.cust_group
    (cust, freight, delivery_time, estimeated_delivery_time) as
    SELECT odr.customer_id cust, avg(odrit.freight_value) freight,
```

```
        avg(date_diff(odr.order_delivered_customer_date, odr.order_purchase_timestamp, DAY))
delivery_time,

        avg(date_diff(odr.order_estimated_delivery_date, odr.order_delivered_customer_date, DAY))

        estimeated_delivery_time

FROM

        `kartik-scaler.ProjectSCALER.orders` odr

LEFT JOIN

        `kartik-scaler.ProjectSCALER.order_items` odrit

ON odr.order_id = odrit.order_id

GROUP BY

        odr.customer_id;



SELECT

        c.customer_state State, ROUND (avg(cg.freight),2) Avg_Freight_Price,

        ROUND (avg(cg.delivery_time),2) Avg_Delivery_Time,

        ROUND (avg(cg.estimeated_delivery_time),2) Avg_Est_Delivery_time

FROM

        `kartik-scaler.ProjectSCALER.customers` c

LEFT JOIN

        `kartik-scaler.ProjectSCALER.cust_group` cg

ON c.customer_id = cg.cust

GROUP BY

        c.customer_state

ORDER BY

        Avg_Freight_Price desc, Avg_Delivery_Time desc, Avg_Est_Delivery_time desc;
```

Row	State	Avg_Freight_Price	Avg_Delivery_Time	Avg_Est_Delivery_time
1	RO	42.38	18.91	19.13
2	RR	42.26	28.98	16.41
3	PB	41.66	19.95	12.37
4	AC	41.52	20.64	19.76

- Above table displays the average freight price for the state including average delivery time of the product and average estimated delivery time for the same.
- This table can help us to understand where delivery services need to be improved and where estimated delivery time is to be corrected or updated.

4. Sort the data to get the following:

1. (5.5) Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

--- Top 5 States with Highest Freight Value ---

```
SELECT
   c.customer_state State, ROUND (avg(cg.freight),2) Avg_Freight_Price
FROM
   `kartik-scaler.ProjectSCALER.customers` c
LEFT JOIN
   `kartik-scaler.ProjectSCALER.cust_group` cg
ON
   c.customer_id = cg.cust
GROUP BY
   c.customer_state
ORDER BY
   Avg_Freight_Price desc
LIMIT 5;
```

Row	State	Avg_Freight_Price
1	RO	42.38
2	RR	42.26
3	PB	41.66
4	AC	41.52
5	PI	39.04

- Above table shows top 5 states with highest Freight price by an average

```
--- Top 5 States with Lowest Freight Value ---
SELECT
  c.customer_state State, ROUND (avg(cg.freight),2) Avg_Freight_Price
FROM
  `kartik-scaler.ProjectSCALER.customers` c
LEFT JOIN
  `kartik-scaler.ProjectSCALER.cust_group` cg
ON c.customer_id = cg.cust
GROUP BY
  c.customer_state
ORDER BY
  Avg_Freight_Price
LIMIT 5;
```

Row	State	Avg_Freight_Price
1	SP	15.3
2	PR	20.45
3	MG	20.8
4	RJ	21.15
5	DF	21.34

- Above table shows top 5 states with lowest Freight price by an average

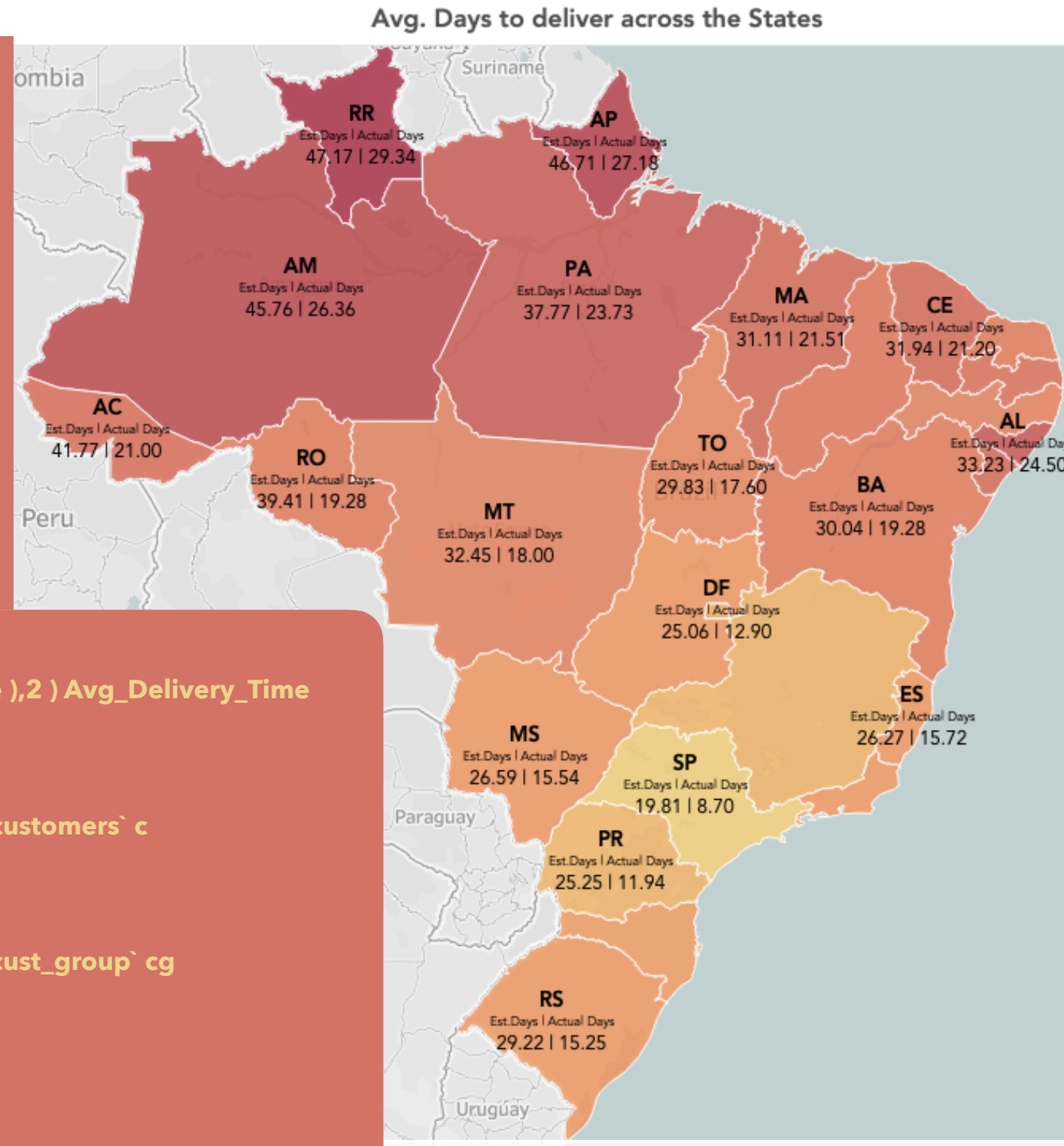
2. (5.6)Top 5 states with highest/lowest average time to delivery

```
--- Top 5 States with Maximum Average Time to Delivery ---
SELECT
  c.customer_state State,
ROUND ( avg ( cg.delivery_time ), 2 ) Avg_Delivery_Time
FROM
  `kartik-scaler.ProjectSCALER.customers` c
LEFT JOIN
  `kartik-scaler.ProjectSCALER.cust_group` cg
ON c.customer_id = cg.cust
GROUP BY
  c.customer_state
```

Row	State	Avg_Delivery_Time
1	RR	28.98
2	AP	26.73
3	AM	25.99
4	AL	24.04
5	PA	23.32

- Above table shows top 5 states with Highest delivery days by an average

```
ORDER BY

    Avg_Delivery_Time desc

LIMIT 5;


--- Top 5 States with
Minimum Average Time to
Delivery ---

SELECT

    c.customer_state State,

ROUND ( avg ( cg.delivery_time ),2 ) Avg_Delivery_Time

FROM

    `kartik-scaler.ProjectSCALER.customers` c

LEFT JOIN

    `kartik-scaler.ProjectSCALER.cust_group` cg

ON c.customer_id = cg.cust

GROUP BY
```



Row	State	Avg_Delivery_Time
1	SP	8.3
2	PR	11.53
3	MG	11.54
4	DF	12.51
5	SC	14.48

- Side table shows top 5 states with Lowest delivery days by an average

3. (5.7) Top 5 states where delivery is really fast/ not so fast compared to estimated date

--- Top 5 states with fast delivery compared to estimated date, delivery is really fast/ not so fast compared to estimated date —-

```
SELECT
    c.customer_state State_with_Fast_Delivery,
ROUND ( avg ( cg.delivery_time ),2 ) Avg_Delivery_Time,
ROUND ( avg ( cg.estimeated_delivery_time ),2 ) Avg_Est_Delivery_time
FROM
    `kartik-scaler.ProjectSCALER.customers` c
LEFT JOIN
     `kartik-scaler.ProjectSCALER.cust_group` cg
ON c.customer_id = cg.cust
GROUP BY
    c.customer_state
ORDER BY
    ( ROUND ( avg ( cg.estimeated_delivery_time ),2 ) - ROUND ( avg ( cg.delivery_time ), 2 )) desc
LIMIT 5;
```

Row	State_with_Fast_Delivery	Avg_Delivery_Time	Avg_Est_Delivery_time
1	SP	8.3	10.14
2	PR	11.53	12.36
3	MG	11.54	12.3
4	RO	18.91	19.13
5	AC	20.64	19.76

- Side table shows top 5 states with fast delivery service by an average

```
 --- Top 5 states with not as fast delivery compared to estimated date --
SELECT
    c.customer_state State_with_Slow_Delivery,
ROUND ( avg ( cg.delivery_time ), 2 ) Avg_Delivery_Time,
ROUND ( avg ( cg.estimeated_delivery_time ), 2 ) Avg_Est_Delivery_time
FROM
    `kartik-scaler.ProjectSCALER.customers` c
LEFT JOIN
```

```
    `kartik-scaler.ProjectSCALER.cust_group` cg
ON c.customer_id = cg.cust
GROUP BY
    c.customer_state
ORDER BY
    ( ROUND ( avg ( cg.estimeated_delivery_time ), 2 ) - ROUND ( avg ( cg.delivery_time ), 2 )) asc
LIMIT 5;
```

- Side table shows top 5 states with Not so fast/ very slow delivery service by an average

Row	State_with_Slow_Delivery	Avg_Delivery_Time	Avg_Est_Delivery_time
1	AL	24.04	7.95
2	RR	28.98	16.41
3	MA	21.12	8.77
4	SE	21.03	9.17
5	CE	20.82	9.96

6. Payment type analysis:

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

```
SELECT
    distinct payment_type
FROM
    `kartik-scaler.ProjectSCALER.payments`;
```

Row	payment_type
1	credit_card
2	voucher
3	not_defined
4	debit_card
5	UPI

- Identifying the unique payment types from the Payments file

```
SELECT
    EXTRACT ( YEAR FROM o.order_purchase_timestamp ) Year,
    EXTRACT ( MONTH FROM o.order_purchase_timestamp ) Month,
    COUNT ( case WHEN p.payment_type = 'credit_card' THEN 1 end ) Credit_Card,
    COUNT ( case WHEN p.payment_type = 'voucher' THEN 2 end ) Voucher,
    COUNT( case WHEN p.payment_type = 'debit_card' THEN 3 end ) Debit_Card,
```

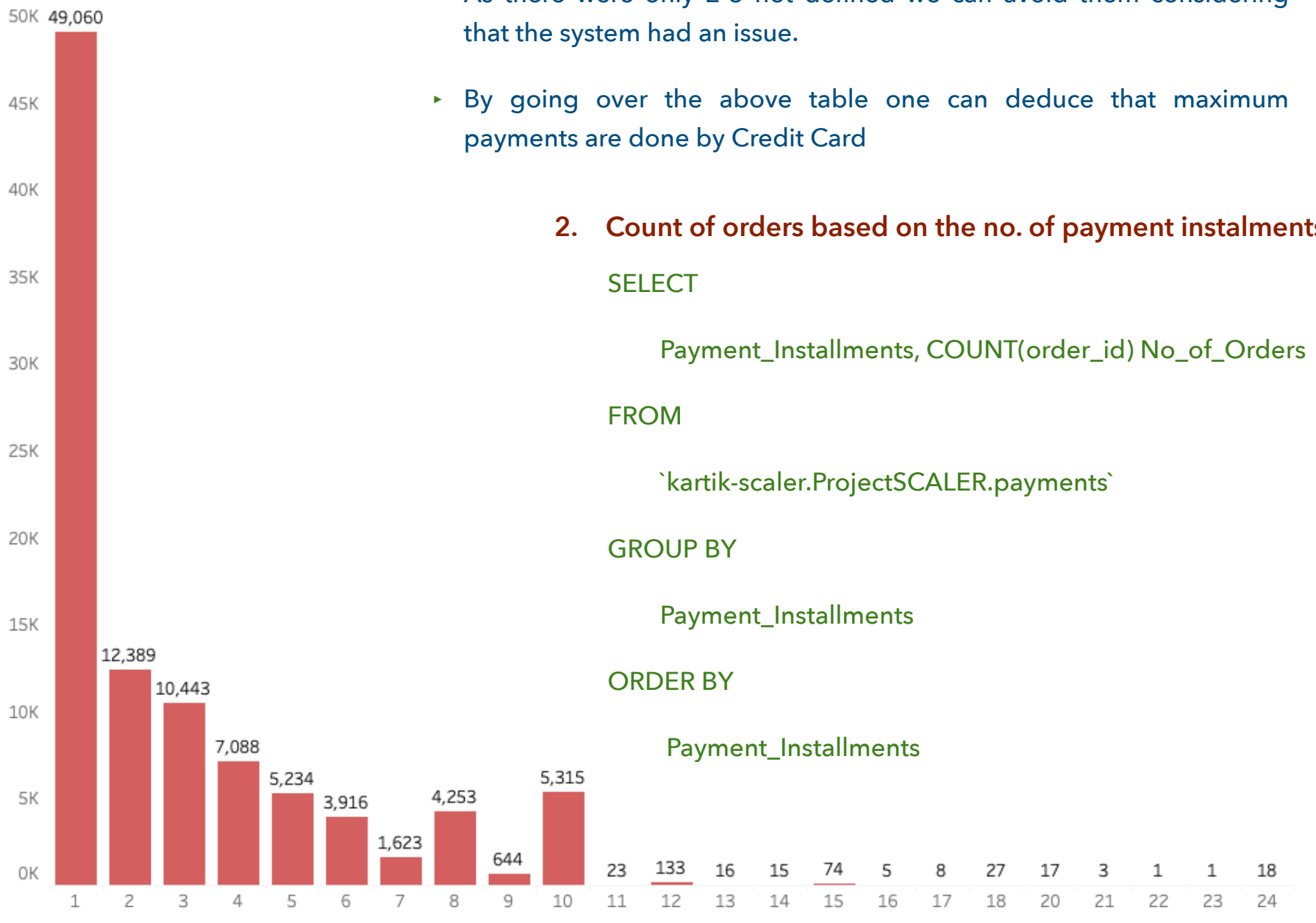
```
        COUNT ( case WHEN p.payment_type = 'UPI' THEN 4 end ) UPI,
        COUNT ( case WHEN p.payment_type = 'not_defined' THEN 5 end ) Not_Defined
FROM
        `kartik-scaler.ProjectSCALER.orders` o
LEFT JOIN
        `kartik-scaler.ProjectSCALER.payments` p
ON o.order_id = p.order_id
GROUP BY
        Year, Month
ORDER BY
        Year, Month;
```

Row	Year	Month	Credit_Card	Voucher	Debit_Card	UPI	Not_Defined
1	2016	9	3	0	0	0	0
2	2016	10	254	23	2	63	0
3	2016	12	1	0	0	0	0
4	2017	1	583	61	9	197	0

- The above table displays the month on month payments done by different payment methods which are by Credit Card, Vouchers, Debit Card, UPI and Not Defined.
- As there were only 2-3 not defined we can avoid them considering that the system had an issue.
- By going over the above table one can deduce that maximum payments are done by Credit Card

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

```
SELECT
        Payment_Installments, COUNT(order_id) No_of_Orders
FROM
        `kartik-scaler.ProjectSCALER.payments`
GROUP BY
        Payment_Installments
ORDER BY
        Payment_Installments
```



- Above bar graph Y-Axis represents the numbers of customers opted for payment instalments and X-Axis represent the No. Of instalments they opted for from 1 to 24 while paying.
- Side table displays the number of customers opted for different instalments for their purchase and the number of instalment they opted for.
- But according to above table it's clear that maximum people are purchasing by paying completely in their first instalment, means they are being low cost products.

Row	Payment_Installments	No_of_Orders
1	0	2
2	1	52546
3	2	12413
4	3	10461

- From above two tables and one graph we can understand that customers are tend to pay using Cred Cards for the shopping.
- Its also observed that people prefer to pay at once rather than instalments the Target Corporation should come out with flexible or multiple options of payment methods to increase the sales as many customers might be using third party payment option which is not an option with Target Corporation
- As maximum people use their credit card The Target Corporation can get in tie with companies offering credit card and create an offers for customers to promote more shopping and increase the customer sample even more in the Brazil.

Study by - KARTIK
EmailID - parkar2619@gmail.com