Business case study | Target

The given database has the following tables with count of rows in each table.

```
SELECT table_name, table_rows
  FROM INFORMATION_SCHEMA.TABLES
  WHERE TABLE_SCHEMA = 'target';
```

TABLE_NAME	TABLE_ROWS
customers	98949
geolocation	1008495
order_items	111600
order_reviews	98582
orders	98750
payments	103534
products	32085
sellers	3095

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

As a part of 1.1: Data type of columns in a table

The structure of the tables can be found using as shown below.

```
SELECT *
  FROM INFORMATION_SCHEMA.columns
  WHERE TABLE_SCHEMA = 'target';
```

TABLE_NAME	COLUMN_NAME	ORDINAL_POSITION	COLUMN_DEFAULT	IS_NULLABLE	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH
customers	customer_id	1	NULL	NO	varchar	50
customers	customer_unique_id	2	NULL	YES	varchar	50
customers	customer_zip_code_prefix	3	NULL	YES	int	NULL
customers	customer_city	4	NULL	YES	varchar	50
customers	customer_state	5	NULL	YES	varchar	10
geolocation	geolocation_zip_code_prefix	1	NULL	YES	bigint	NULL
geolocation	geolocation_lat	2	NULL	YES	double	NULL
geolocation	geolocation_lng	3	NULL	YES	double	NULL
geolocation	geolocation_city	4	NULL	YES	text	65535
geolocation	geolocation_state	5	NULL	YES	text	65535
order_items	order_id	1	NULL	NO	varchar	50
order_items	order_item_id	2	NULL	YES	int	NULL
order_items	product_id	3	NULL	YES	varchar	50
order_items	seller_id	4	NULL	YES	varchar	50
order_items	shipping_limit_date	5	NULL	YES	datetime	NULL
order_items	price	6	NULL	YES	double	NULL

Some basic exploratory data analysis can be done on these tables like

a.

```
Select count(distinct customer_unique_id) 'Unique_cusotmers_cnt' from customers
```

Unique_cusotmers_cnt
96096

So far target reached has 96096 costumers.

b.

```
Select count(distinct seller_id) 'Unique_sellers_cnt' from sellers
```

Unique_sellers_cnt
3095

So far target has 3095 registered sellers.

c.

```
Select round(sum(payment_value),2) 'Orders_cnt' from payments;
```

Orders_cnt
16008872.12

The total amount of the Target marketplace is 16M $

1.2 Time period for which the data is given

The overall dataset we are given of target purchases can be found in the following way using order table which has information of all the orders according to the date.

```
Select min(order_purchase_timestamp) 'From', max(order_purchase_timestamp) 'To' from orders;
```

1.3 Cities and States covered in the dataset

The customers and sellers of target in brazil are distributed as following.

```
Select 'Customers', (Select count(distinct customer_state) from customers) 'State/province'
        ,(Select count(distinct customer_city) from customers) 'City'
        ,(Select count(distinct customer_zip_code_prefix) from customers) 'zip_code'
union all
Select 'Sellers', (Select count(distinct seller_state) from sellers) 'State'
        ,(Select count(distinct seller_city) from sellers) 'City'
        ,(Select count(distinct seller_zip_code_prefix) from sellers) 'zip_code';
```

	Customers	State/province	City	zip_code
▶	Customers	27	4119	14994
	Sellers	23	610	2246

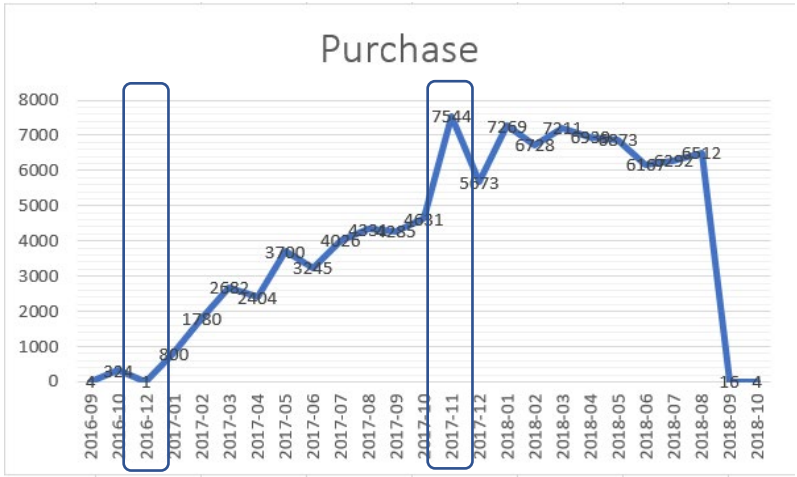
1. In-depth exploration

2.1 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?

we see the trend using the below query

```
with cte as (Select * , extract(year from order_purchase_timestamp) 'year'
  , extract(month from order_purchase_timestamp) 'month'  from orders)
Select  year, month, concat(year, '-', lpad(month, 2, '0')) 'date', count(1) 'Purchases' from cte
group by year, month order by year, month;
```

year	month	date	Puchases
2016	9	2016-09	4
2016	10	2016-10	324
2016	12	2016-12	1
2017	1	2017-01	800
2017	2	2017-02	1780
2017	3	2017-03	2682
2017	4	2017-04	2404
2017	5	2017-05	3700
2017	6	2017-06	3245
2017	7	2017-07	4026
2017	8	2017-08	4331
2017	9	2017-09	4285
2017	10	2017-10	4631
2017	11	2017-11	7544
2017	12	2017-12	5673
2018	1	2018-01	7269
2018	2	2018-02	6728
2018	3	2018-03	7211
2018	4	2018-04	6939
2018	5	2018-05	6873
2018	6	2018-06	6167
2018	7	2018-07	6292
2018	8	2018-08	6512
2018	9	2018-09	16
2018	10	2018-10	4

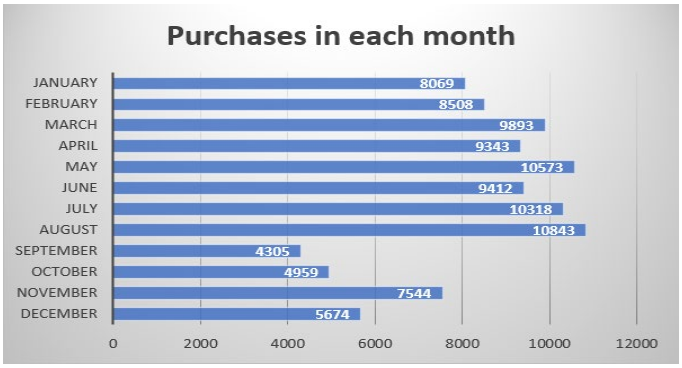


The highest purchase is recorded in 2017-11 with 7544 total purchases in that month and the lowest in 2016-12 with only a single purchase.

Also, we can see in which month the purchase are high irrespective of the year as shown below.

```
with cte as (Select *,   extract(month from order_purchase_timestamp) 'Month_of_purchase' from orders)

Select Month_of_purchase, count(1) 'frequency' from cte

group by Month_of_purchase

order by Month_of_purchase;
```

Month_of_purchase	frequency
1	8069
2	8508
3	9893
4	9343
5	10573
6	9412
7	10318
8	10843
9	4305
10	4959
11	7544
12	5674



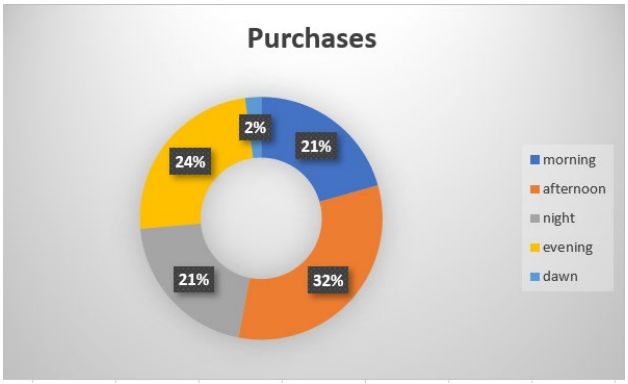
Insights: We can see that in the months of march to august the purchases are high in the target market and from September to December the purchases counts are comparatively less.

Recommendations: Assuming people tend to shop the things in the months of march to august (may be it could be a festival season, year-end bonus time etc.) target can make marketing and advertising with attractive offers to draw the attention of the customer. Thereby scaling up the business.

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

```
with cte as (Select * , case when extract(hour from order_purchase_timestamp) between 4 and 7 then 'dawn'
                            when extract(hour from order_purchase_timestamp) between 8 and 11 then 'morning'
                            when extract(hour from order_purchase_timestamp) between 12 and 16 then 'afternoon'
                            when extract(hour from order_purchase_timestamp) between 17 and 20 then 'evening'
                                 else 'night' end as 'Time'  from orders)
Select Time, count(time) 'Purchases' from cte
group by Time
```

	Time	Purchases
▶	morning	20507
	afternoon	32211
	night	20502
	evening	24094
	dawn	2127



Insights: We can see that in the afternoon, the purchases are comparatively high and the purchases are least at dawn.

Recommendations: Since the traffic of customers will be high from morning to evening more resources like supply chain and other back-end people should be ready to process the order and reduce the lag that might cause in the process.

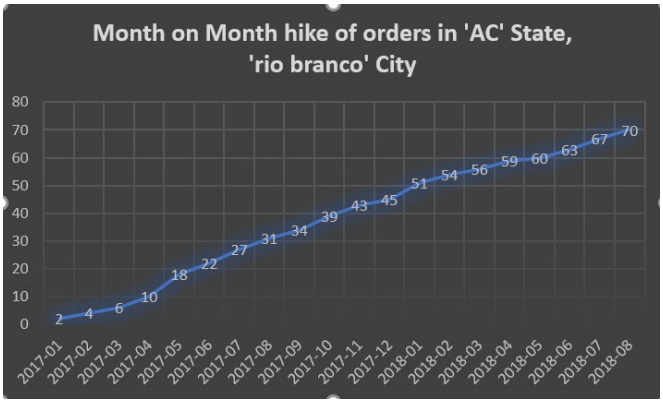
3. Evolution of E-commerce orders in the Brazil region:

3.1 The month on month sales in each city of a state can be obtained as shown below.

The count of target company customers of each state in Brazil can be found using the below queries

```
with cte as(

Select O.order_id, O.order_purchase_timestamp, C.customer_city, C.customer_state

        , extract(year from order_purchase_timestamp) 'year'

        , extract(month from order_purchase_timestamp) 'month'

        from customers C join orders O on C.customer_id = O.customer_id)

 Select distinct customer_state, customer_city, concat(year, '-', lpad(month, 2, '0')) 'date', count(order_id) over(partition by customer_state, customer_city  order by year, month) 'Month_on_Month_order' from cte order by customer_state, customer_city, year, month;
```

customer_state	customer_city	date	Month_on_Month_order
AC	epitaciolandia	2017-10	1
AC	manoel urbano	2017-09	1
AC	porto acre	2017-04	1
AC	rio branco	2017-01	2
AC	rio branco	2017-02	4
AC	rio branco	2017-03	6
AC	rio branco	2017-04	10
AC	rio branco	2017-05	18
AC	rio branco	2017-06	22
AC	rio branco	2017-07	27
AC	rio branco	2017-08	31
AC	rio branco	2017-09	34



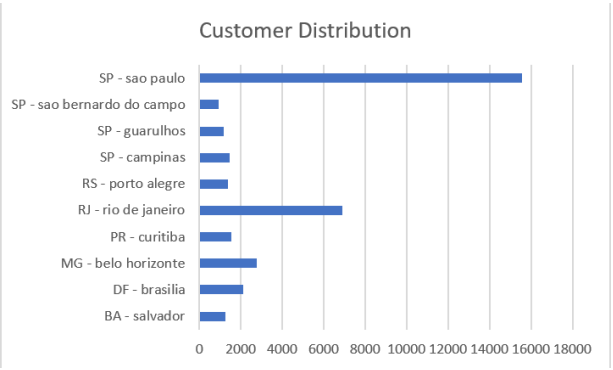
Insights: by modifying the query by ordering based on Month_on_Month_order. We can conclude that people from AC, AL, AM, AP, PB etc it is evident that the orders in these county is very less and states like SP, RJ are high

Recommendations: In those states the target should come up with business strategy to reach them with advertising and marketing. Also in those states where orders are high, the target should focus on continuous engagement with the customers with better customer experience and scale the business. Also supply chain and delivery should be taken care, because higher the incoming orders there could be of some breakpoints of the business.

3.2 How are customers distributed in Brazil

```
Select customer_state 'State', customer_city 'City', concat(customer_state ,  ' - ',customer_city ) 'Location'
, count(customer_id) 'Cnt' from customers group by customer_state, customer_city  order by Cnt desc;
-- limit 10;
```

	State	City	Location	Cnt
▶	SP	sao paulo	SP - sao paulo	15540
	RJ	rio de janeiro	RJ - rio de janeiro	6882
	MG	belo horizonte	MG - belo horizonte	2773
	DF	brasilia	DF - brasilia	2131
	PR	curitiba	PR - curitiba	1521
	SP	campinas	SP - campinas	1444
	RS	porto alegre	RS - porto alegre	1379
	BA	salvador	BA - salvador	1245
	SP	guarulhos	SP - guarulhos	1189
	SP	sao bernardo do campo	SP - sao bernardo do campo	938
	RJ	niteroi	RJ - niteroi	849



Insights: we can see that city ‘sao paulo’ from state ‘SP’ has the highest customer count. Also ‘rio de Janeiro’ from ‘RJ’ state etc.

Recommendations: Target should focus on deploying more on delivery network, warehousing, resources and supply as there would be high demands in those cities. Also this reduces the time required to deliver the customer which is liking factor for the customers.

Similarly for customers, the same way for distribution of sellers of the target market can also be found

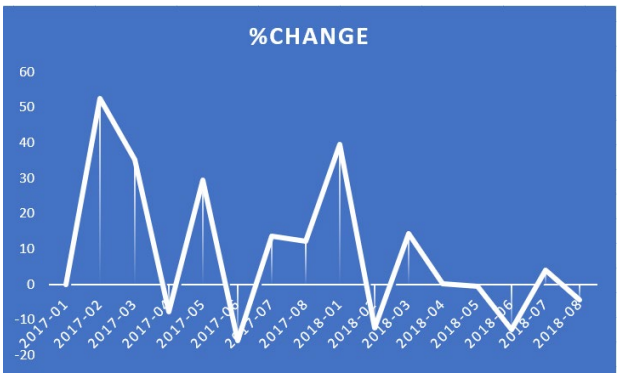
4. Impact on Economy: Analyze the money movemented by e-commerce by looking at order prices, freight and others.

4.1 Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only)

Change of percentage given by

```
with cte as( Select P.payment_value, month(order_purchase_timestamp) 'month_'   , year(order_purchase_timestamp) 'year_'
from orders O join Payments P on P.order_id = O.order_id)
, cte_final as( Select year_ 'Year', month_ 'Month', concat( year_, '-',lpad(month_, 2, '0')) 'Date'
                ,round(sum(payment_value),2) 'Tot_turnover' from cte where month_ between 1 and 8
                group by year_ , month_)
, cte_final2 as( Select *, Tot_turnover - lag(Tot_turnover,1) over (order by year, month) 'diff' from cte_final)
Select *, round(diff/Tot_turnover*100, 2) '%Change' from cte_final2 order by Year, Month
```

Year	Month	Date	Tot_turnover	diff	%Change
2017	1	2017-01	138488.04	NULL	NULL
2017	2	2017-02	291908.01	153419.97	52.56
2017	3	2017-03	449863.6	157955.58999999997	35.11
2017	4	2017-04	417788.03	-32075.56999999995	-7.68
2017	5	2017-05	592918.82	175130.78999999992	29.54
2017	6	2017-06	511276.38	-81642.43999999994	-15.97
2017	7	2017-07	592382.92	81106.54000000004	13.69
2017	8	2017-08	674396.32	82013.3999999999	12.16
2018	1	2018-01	1115004.18	440607.86	39.52
2018	2	2018-02	992463.34	-122540.83999999997	-12.35
2018	3	2018-03	1159652.12	167188.78000000014	14.42
2018	4	2018-04	1160785.48	1133.3599999998696	0.1
2018	5	2018-05	1153982.15	-6803.3300000000745	-0.59
2018	6	2018-06	1023880.5	-130101.6499999999	-12.71



Insights: currently the %change of cost of orders are in downtrend

Recommendations: Target should look into the trend and take the corresponding actions because this can not go un-addressed.

4.2 Mean & Sum of price and freight value by customer state

```
Select C.Customer_state, round(sum(OI.price + freight_value),2) 'Sum', round(avg(OI.price + freight_value),2) 'Mean'
from orders O join order_items OI on OI.order_id = O.order_id join customers C on O.customer_id = C.customer_id
group by C.customer_state order by sum desc;
```

Customer_state	Sum	Mean
SP	5921678.12	124.8
RJ	2129681.98	146.08
MG	1856161.49	141.38
RS	885826.76	142.07
PR	800935.44	139.54
BA	611506.67	160.97
SC	610213.6	146.12
DF	353229.44	146.81
GO	347706.93	149.04

Insights: The total sum of the orders in SP, RJ, MJ states are very high.

Recommendations: Target can deploy the warehouse nearby to all these cities so thereby reducing the costs of delivery.

5. Analysis on sales, freight and delivery time

```
Select 'order_purchase_timestamp' as 'Order Table', count(*) Null_cnt from orders where order_purchase_timestamp = 0
union all
Select 'order_delivered_carrier_date' as 'Order Table', count(*) Null_cnt from orders where order_delivered_carrier_date = 0
union all
Select 'order_delivered_customer_date' as 'Order Table', count(*) Null_cnt from orders where order_delivered_customer_date = 0
union all
Select 'order_estimated_delivery_date' as 'Order Table', count(*) Null_cnt from orders where order_estimated_delivery_date = 0
```

Order Table	Null_cnt
order_purchase_timestamp	0
order_delivered_carrier_date	1783
order_delivered_customer_date	2965
order_estimated_delivery_date	0

In the order table order_delivere_carrier_date and order_delivered_customer_date has null values entries of 1783 and 2965 respectively.

5.1 Calculate days between purchasing, delivering and estimated delivery

```
with cte as( Select *, datediff(order_estimated_delivery_date, order_purchase_timestamp) 'est_del_days'
    , datediff(order_delivered_customer_date, order_estimated_delivery_date) 'act_del_days'
    from orders where order_estimated_delivery_date != 0 and order_delivered_customer_date  != 0)
Select order_id, order_purchase_timestamp, est_del_days, act_del_days, est_del_days+ act_del_days  tot_days_took from cte
-- where act_del_days > 10 -- considering 10 as threshold
order by act_del_days desc
```

id	order_id	order_purchase_timestamp	est_del_days	act_del_days	tot_days_took
1	1b3190b2dfa9d789e1f14c05b647a14a	2018-02-23 14:57:35	20	188	208
2	ca07593549f1816d26a572e06dc1eab6	2017-02-21 23:31:27	29	181	210
3	47b40429ed8cce3aee9199792275433f	2018-01-03 09:44:01	16	175	191
4	2fe324febf907e3ea3f2aa9650869fa5	2017-03-13 20:17:10	23	167	190
5	285ab9426d6982034523a855f55a885e	2017-03-08 22:47:40	29	166	195
6	440d0d17af552815d15a9e41abe49359	2017-03-07 23:59:51	31	165	196
7	c27815f7e3dd0b926b58552628481575	2017-03-15 23:23:17	26	162	188
8	0f4519c5f1c541ddec9f21b3bddd533a	2017-03-09 13:26:57	33	161	194

Understanding : The serial id 1 -> this order ordered on 2018-02-23, the estimated delivery date was 20 days from date of purchase, 188 days was the difference between estimated delivery date and actual delivery date. So it took 208 days to reach the customer. 188 days of delivery date difference is not at all a toleratable from the customer point of view.

Insights: for all these orders whose difference of date of delivery to the estimated date of delivery is more than 100.

Recommendations: Target should focus on reducing this gap by better management of delivery network.

5.2 Create columns:

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

```
with cte as( Select *, datediff(order_delivered_customer_date, order_purchase_timestamp) 'time_to_delivery'
                , datediff(order_delivered_customer_date, order_estimated_delivery_date) 'diff_estimated_delivery'
      from orders where order_estimated_delivery_date != 0 and order_delivered_customer_date  != 0)
Select row_number() over (order by time_to_delivery desc) 'id', order_id, order_purchase_timestamp, time_to_delivery,
diff_estimated_delivery  from cte
-- where act_del_days > 10 -- considering 10 as threshold
order by time_to_delivery desc
```

id	order_id	order_purchase_timestamp	time_to_delivery	diff_estimated_delivery
1	ca07593549f1816d26a572e06dc1eab6	2017-02-21 23:31:27	210	181
2	1b3190b2dfa9d789e1f14c05b647a14a	2018-02-23 14:57:35	208	188
3	440d0d17af552815d15a9e41abe49359	2017-03-07 23:59:51	196	165
4	285ab9426d6982034523a855f55a885e	2017-03-08 22:47:40	195	166
5	2fb597c2f772eca01b1f5c561bf6cc7b	2017-03-08 18:09:02	195	155
6	0f4519c5f1c541ddec9f21b3bddd533a	2017-03-09 13:26:57	194	161
7	47b40429ed8cce3aee9199792275433f	2018-01-03 09:44:01	191	175
8	2fe324febf907e3ea3f2aa9650869fa5	2017-03-13 20:17:10	190	167

Understanding: So the sl_id 1, it took 210 days to reach the customer, target should look into these orders and find what is the actual reason of delay to this extent and find the solutions to reduce this gap for betterment of customer satisfaction.

5.3 Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

customer_state	Mean_freight_value	Mean_time_to_delivery	Mean_diff_estimated_delivery
AC	40.05	21	-20
AL	35.87	25	-8
AM	33.31	27	-19
AP	34.16	29	-18
BA	26.49	20	-10
CE	32.73	21	-11
DF	21.07	13	-12
ES	22.03	16	-10
GO	22.56	16	-12
MA	38.49	22	-9

```
with cte as(Select O.order_id, O.order_purchase_timestamp, C.customer_state, OI.freight_value

                , datediff(O.order_delivered_customer_date, O.order_purchase_timestamp) 'time_to_delivery'

                , datediff(O.order_delivered_customer_date, O.order_estimated_delivery_date) 'diff_estimated_delivery'

   from orders O join  Order_items OI on OI.order_id = O.order_id  join customers C on C.customer_id = O.customer_id

where O.order_estimated_delivery_date != 0 and O.order_delivered_customer_date  != 0)

Select Customer_state, round(avg(freight_value),2) as 'Mean_freight_value'

   , ceil(round(avg(time_to_delivery),2)) as 'Mean_time_to_delivery', ceil(round(avg(diff_estimated_delivery),2))  as
'Mean_diff_estimated_delivery'  from cte group by customer_state order by customer_state
```

customer_state	Mean_freight_value	Mean_time_to_delivery	Mean_diff_estimated_delivery
AC	40.05	21	-20
AL	35.87	25	-8
AM	33.31	27	-19
AP	34.16	29	-18
BA	26.49	20	-10
CE	32.73	21	-11
DF	21.07	13	-12
ES	22.03	16	-10
GO	22.56	16	-12
MA	38.49	22	-9

To describe this resultset, the state ‘AC’ has mean freight value in delivering the products to the customers. 21 days of avg days to reach the customers and in avg within 20 days than the estimated delivery date the ordered items are reaching the customers.

5.4 Sort the data to get the following:

5.4.1

```
with cte as(Select O.order_id, O.order_purchase_timestamp, C.customer_state, OI.freight_value
                , datediff(O.order_delivered_customer_date, O.order_purchase_timestamp) 'time_to_delivery'
                , datediff(O.order_delivered_customer_date, O.order_estimated_delivery_date) 'diff_estimated_delivery'
   from orders O join  Order_items OI on OI.order_id = O.order_id  join customers C on C.customer_id = O.customer_id
where O.order_estimated_delivery_date != 0 and O.order_delivered_customer_date  != 0)
, cte_final as(Select Customer_state, round(avg(freight_value),2) as 'Mean_freight_value'
    , ceil(round(avg(time_to_delivery),2)) as 'Mean_time_to_delivery', ceil(round(avg(diff_estimated_delivery),2))  as
'Mean_diff_estimated_delivery'  from cte group by customer_state)
, cte_final2 as (Select *, dense_rank() over(order by Mean_freight_value desc) 'ranker' from cte_final)
Select * from cte_final2 where ranker <=5 order by ranker;
```

	customer_state	Mean_freight_value	Mean_time_to_delivery	Mean_diff_estimated_delivery	ranker
▶	PB	43.09	21	-13	1
	RR	43.09	29	-18	1
	RO	41.33	20	-20	2
	AC	40.05	21	-20	3
	PI	39.12	20	-11	4
	MA	38.49	22	-9	5

Insights: mean of freight value of these states are high

Recommendations: Target can deploy warehouses near to these cities in common. So delivery costs and other operational costs can be reduced

5.4.2

```
with cte as(Select O.order_id, O.order_purchase_timestamp, C.customer_state, OI.freight_value

            , datediff(O.order_delivered_customer_date, O.order_purchase_timestamp) 'time_to_delivery'

            , datediff(O.order_delivered_customer_date, O.order_estimated_delivery_date) 'diff_estimated_delivery'

  from orders O join  Order_items OI on OI.order_id = O.order_id      join customers C on C.customer_id =
O.customer_id

where O.order_estimated_delivery_date != 0 and O.order_delivered_customer_date  != 0)

, cte_final as(Select Customer_state, round(avg(freight_value),2) as 'Mean_freight_value'

   , ceil(round(avg(time_to_delivery),2)) as 'Mean_time_to_delivery', ceil(round(avg(diff_estimated_delivery),2))  as
'Mean_diff_estimated_delivery'  from cte group by customer_state)

, cte_final2 as (Select *, dense_rank() over(order by Mean_time_to_delivery desc) 'ranker' from cte_final)

Select * from cte_final2 where ranker <=5 order by ranker
```

customer_state	Mean_freight_value	Mean_time_to_delivery	Mean_diff_estimated_delivery	ranker
AP	34.16	29	-18	1
RR	43.09	29	-18	1
AM	33.31	27	-19	2
AL	35.87	25	-8	3
PA	35.63	24	-14	4
MA	38.49	22	-9	5
SE	36.57	22	-10	5

Insights: mean time to taken to deliver the ordered items into these states are bit high

Recommendations: Target can deploy warehouses near to these cities in common. So delivery costs and other operational costs can be reduced

5.4.3

```
with cte as(Select O.order_id, O.order_purchase_timestamp, C.customer_state, OI.freight_value
                , datediff(O.order_delivered_customer_date, O.order_purchase_timestamp) 'time_to_delivery'
                , datediff(O.order_delivered_customer_date, O.order_estimated_delivery_date) 'diff_estimated_delivery'
   from orders O join  Order_items OI on OI.order_id = O.order_id  join customers C on C.customer_id = O.customer_id
where O.order_estimated_delivery_date != 0 and O.order_delivered_customer_date  != 0)
, cte_final as( Select Customer_state, round(avg(freight_value),2) as 'Mean_freight_value'
    , ceil(round(avg(time_to_delivery),2)) as 'Mean_time_to_delivery', ceil(round(avg(diff_estimated_delivery),2))  as
'Mean_diff_estimated_delivery'  from cte group by customer_state)
, cte_final2 as ( Select *, dense_rank() over(order by Mean_diff_estimated_delivery) 'ranker' from cte_final)
Select * from cte_final2 where ranker <=5 order by ranker
```

	customer_state	Mean_freight_value	Mean_time_to_delivery	Mean_diff_estimated_delivery	ranker
▶	RO	41.33	20	-20	1
	AC	40.05	21	-20	1
	AM	33.31	27	-19	2
	AP	34.16	29	-18	3
	RR	43.09	29	-18	3
	RS	21.61	16	-14	4
	PA	35.63	24	-14	4
	MT	28	18	-14	4
	MG	20.63	12	-13	5
	PR	20.47	12	-13	5
	PE	32.69	19	-13	5
	PB	43.09	21	-13	5
	RN	35.72	20	-13	5

Insights: The delivery is really fast in these states compared to the estimated delivery date. So target is doing exceptionally well in delivering the ordered items to the customer

Recommendations: Target should consistently follow this in order to please the customer base and engage them

6. Payment type analysis:

6.1 Month over Month count of orders for different payment types

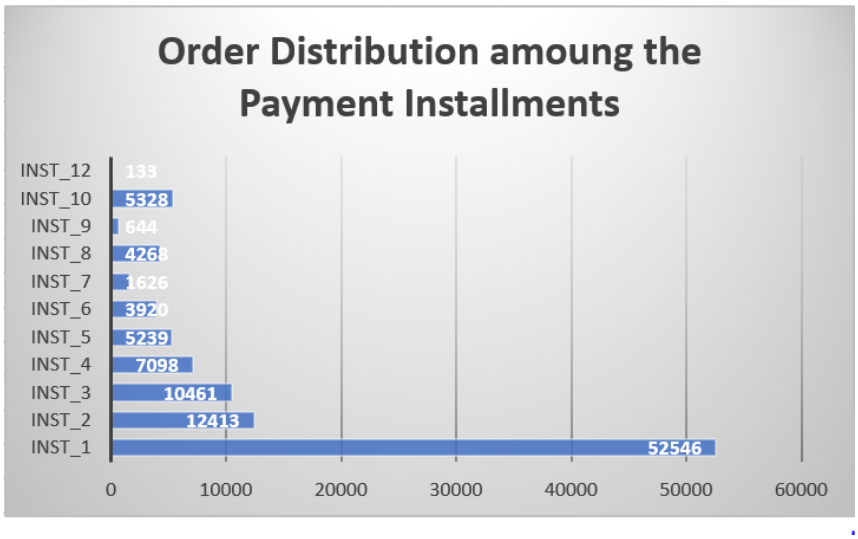
```
Select concat(extract(year from order_purchase_timestamp)
                        , '-', lpad(extract(month from order_purchase_timestamp),2, '0')) 'Purchase_date'
        ,P.Payment_type, count(*) Cnt
        from payments P join orders O on P.order_id = O.order_id
group by extract(year from order_purchase_timestamp), extract(month from order_purchase_timestamp), payment_type
order by extract(year from order_purchase_timestamp), extract(month from order_purchase_timestamp)
```

	Purchase_date	Payment_type	Cnt
▶	2016-09	credit_card	3
	2016-10	credit_card	254
	2016-10	debit_card	2
	2016-10	UPI	63
	2016-10	voucher	23
	2016-12	credit_card	1
	2017-01	credit_card	583
	2017-01	debit_card	9

6.2 Distribution of payment installments and count of orders

```
Select Payment_installments, count(1) Cnt from payments P
group by payment_installments order by Payment_installments desc;
```

	Payment_installments	Cnt
	13	16
	12	133
	11	23
	10	5328
	9	644
	8	4268
	7	1626
	6	3920
	5	5239
	4	7098
	3	10461
	2	12413
	1	52546



Insights: More customers have opted minimum number of installments like 1,2,3 and 4 to buy products

Recommendations: Lesser the number of installments offered to the customers, they tend to make more orders in the target market. So target should offer the installments in this mentioned window.