Target : Business Case Study

There are 8 tables in the dataset ‘target’, which are as follows

1. Customers
2. Orders
3. Payments
4. Products
5. geolocation
6. order_items
7. order_reviews
8. Sellers

```
SELECT * FROM target.INFORMATION_SCHEMA.TABLES
WHERE table_name = ‘order_items';
```

order_items table

Row	table_catalog	table_schema	table_name	table_type	ddl
1	targetsql-375812	target	order_items	BASE TABLE	CREATE TABLE `targetsql-375812.target.order_items` (order_id STRING, order_item_id INT64, product_id STRING, seller_id STRING, shipping_limit_date TIMESTAMP, price FLOAT64, freight_value FLOAT64) OPTIONS(expiration_timestamp=TIMESTAMP "2023-03-26T12:57:32.083Z");

Geolocation table

fullname	type
geolocation_zip_code_prefix	INTEGER
geolocation_lat	FLOAT
geolocation_lng	FLOAT
geolocation_city	STRING
geolocation_state	STRING

Customers table

fullname	type
customer_id	STRING
customer_unique_id	STRING
customer_zip_code_prefix	INTEGER

customer_city	STRING
customer_state	STRING

order_reviews table

fullname	type
review_id	STRING
order_id	STRING
review_score	INTEGER
review_comment_title	STRING
review_creation_date	TIMESTAMP
review_answer_timestamp	TIMESTAMP

Orders table

fullname	type
order_id	STRING
customer_id	STRING
order_status	STRING
order_purchase_timestamp	TIMESTAMP
order_approved_at	TIMESTAMP
order_delivered_carrier_date	TIMESTAMP
order_delivered_customer_date	TIMESTAMP
order_estimated_delivery_date	TIMESTAMP

Payments table

fullname	type
order_id	STRING
payment_sequential	INTEGER
payment_type	STRING
payment_installments	INTEGER
payment_value	FLOAT

Products table

fullname	type
product_id	STRING
product_category	STRING
product_name_length	INTEGER
product_description_length	INTEGER
product_photos_qty	INTEGER

product_weight_g	INTEGER
product_length_cm	INTEGER
product_height_cm	INTEGER
product_width_cm	INTEGER

Sellers table

fullname	type
seller_id	STRING
seller_zip_code_prefix	INTEGER
seller_city	STRING
seller_state	STRING

Points to be noted about datatypes used :

1. Latitude and longitude from geolocation table are of float datatype as they are of the form '-10.9105145187545'
2. All dates and timestamps are of the datatype 'timestamp' and is of the form 'yyyy-mm-dd hh:mm:ss'. For those values where only dates need to be mentioned, it is stored as '2018-09-23 00:00:00'.
3. Names of places, comments, reviews, categories, ids (order, seller, customer etc) are all string datatype.
4. Quantity, measurements, zip codes etc come under integer datatype.
5. Price and weights are in float datatype.

Note: Datatypes may vary according to the data. The above mentioned datatypes are valid only for target sql dataset as per its data. It is important to keep in mind of the kind of data we will input as the attributes/tables are created accordingly.

1. Time period for which the data is given

```
SELECT min(order_purchase_timestamp) as data_start_date,
       max(order_purchase_timestamp) as data_end_date
FROM target.orders;
```

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

- Our order data lies within timeframe given in the above table.

In the orders tables, we have 5 order statuses using

```
SELECT DISTINCT order_status FROM target.orders;
```

Row	order_status
1	created
2	shipped
3	approved
4	canceled

5	invoiced
6	delivered
7	processing
8	unavailable

2. In order_reviews table, we can check the reviews using the below query

```
SELECT order_id,
       review_comment_title
FROM target.order_reviews
WHERE review_comment_title is not null;
```

order_reviews table

Row	order_id	review_comment_title
2	65e004516d71c3c2e5e08910ab872a08	Wrong product
3	61c724e40941762318a52917a4acc55c	I hated it sa's at loss
4	574f906eedf477adaf3507222887d81f	Lack of respect
5	430bd310105f069ac72ef7758f40299b	The cÃ
6	50e46ab38a7ca83546cab50d4bca4835	NON-RECEIPT
7	474f2b1387a07ea0159ca7a996dd09e8	I didn't receive my order
8	72fb560d115ecf3b15de9b853c02e505	I received the product
9	d1aa2409cda0cbe698a489a0f9364c2a	Delay
10	37cd892c7a6275aecbe6dfdcee8de6b7	Wrong product

- Note: Out of 99224 records/orders, we have 11549 reviews only, which means only 11.6 percent of total orders in the period of 2016 to 2018 got reviews.

To check how many orders are there with issues in deliveries,

```
SELECT review_comment_title
FROM target.order_reviews
WHERE lower(review_comment_title) LIKE '%didn_t receive%' OR
      lower(review_comment_title) LIKE '%delay%' OR
      lower(review_comment_title) LIKE 'not%delivered' OR
      lower(review_comment_title) LIKE '%awaiting%' OR
      lower(review_comment_title) LIKE '%waiting%' OR
      lower(review_comment_title) LIKE ‘%delivery%'
      lower(review_comment_title) LIKE '%partial%' OR
      lower(review_comment_title) LIKE '%not arrived%' OR
      lower(review_comment_title) LIKE '%missing%' OR
      lower(review_comment_title) LIKE '%incomplete%' OR
      lower(review_comment_title) LIKE '%late%' OR
      lower(review_comment_title) LIKE '%different product%' OR
      lower(review_comment_title) LIKE '%wrong product%';
```

order_reviews table

Row	review_comment_title

1	I didn't receive my order
2	Delay
3	Delivery delay
4	delivery delay
5	Late delivery
6	Delay DMS
7	Delay delivery
8	I didn't receive
9	I didn't receive
10	delay

- Note: there are 979 such orders where there have been delays in delivery and 8.47% of total reviews have delivery issues.

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

```
SELECT
      DISTINCT c.customer_city,
      c.customer_state
FROM target.customers as c
JOIN target.orders as o
ON c.customer_id = o.customer_id;
```

Customers and orders

Row	customer_city	customer_state
1	acu	RN
2	ico	CE
3	ipe	RS
4	ipu	CE
5	ita	SC
6	itu	SP
7	jau	SP
8	luz	MG
9	poa	SP
10	uba	MG

- Brazil has a total of 27 states according to the data and Target gets orders from 4310 cities out of 5570 cities across the 27 states.

○ In-depth analysis

2.1 Growing trend on e-commerce in Brazil.

```
SELECT
      format_datetime('%Y-%m', order_purchase_timestamp) as month,
      count(order_id) as orders_count
FROM target.orders
GROUP BY month
order by month;
```

The below data shows the number of order received by Target on a. Monthly basis from September 2016 to October 2018.

Row	month	orders_count
1	2016-09	4
2	2016-10	324
3	2016-12	1
4	2017-01	800
5	2017-02	1780
6	2017-03	2682
7	2017-04	2404
8	2017-05	3700
9	2017-06	3245
10	2017-07	4026
11	2017-08	4331
12	2017-09	4285
13	2017-10	4631
14	2017-11	7544
15	2017-12	5673
16	2018-01	7269
17	2018-02	6728
18	2018-03	7211
19	2018-04	6939
20	2018-05	6873
21	2018-06	6167
22	2018-07	6292

23	2018-08	6512
24	2018-09	16
25	2018-10	4

- It can be noted that the sales/orders has improved in 2017 compared 2016 and the first 8 months of 2018 did better compared to the first 8 months in 2017 and hence there is indeed a growing trend on e-commerce in Brazil.

```
SELECT
      format_datetime('%Y-%m', order_purchase_timestamp) as month,
      count(order_id) as orders_count
FROM target.orders
GROUP BY month
order by orders_count desc;
```

Row	month	orders_count
1	2017-11	7544
2	2018-01	7269
3	2018-03	7211
4	2018-04	6939
5	2018-05	6873
6	2018-02	6728
7	2018-08	6512
8	2018-07	6292
9	2018-06	6167
10	2017-12	5673
11	2017-10	4631
12	2017-08	4331
13	2017-09	4285
14	2017-07	4026
15	2017-05	3700
16	2017-06	3245
17	2017-03	2682
18	2017-04	2404
19	2017-02	1780
20	2017-01	800
21	2016-10	324
22	2018-09	16
23	2016-09	4
24	2018-10	4

25	2016-12	1

- It can be clearly observed from the above data that the maximum sale has been done in the month of Nov 2017 and sales maintained trend in the first 5 months of 2018.
- It may be concluded that summer in Brazil seems good for business as that is where the peaks exist.

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

```
SELECT
  CASE
    WHEN FORMAT_TIMESTAMP("%H", order_purchase_timestamp) between '00' and '06'
    THEN 'Dawn'
    WHEN FORMAT_TIMESTAMP("%H", order_purchase_timestamp) between '07' and '11'
    THEN 'Morning'
    WHEN FORMAT_TIMESTAMP("%H", order_purchase_timestamp) between '12' and '16'
    THEN 'Afternoon'
    WHEN FORMAT_TIMESTAMP("%H", order_purchase_timestamp) between '17' and '23'
    THEN 'Night'
  END AS daily_order_traffic,
  COUNT(order_id) as order_count
FROM target.orders
GROUP BY daily_order_traffic
order by order_count desc ;
```

Row	daily_order_traffic	order_count
1	Night	40250
2	Afternoon	32211
3	Morning	21738
4	Dawn	5242

- From the above result se, we can conclude that most purchases are made in the night and the purchase rate increases as time progresses in the day i.e. dawn < morning < afternoon < night .

Evolution of E-commerce orders in the Brazil region

3.1 Get month on month orders by states

```
SELECT
      format_datetime('%Y-%m', o.order_purchase_timestamp) as month,
      c.customer_state,
      count(o.order_id) as orders_count
FROM target.orders as o
JOIN target.customers as c
ON c.customer_id = o.customer_id
GROUP BY c.customer_state,month
order by c.customer_state desc, month desc;
```

Row	month	customer_state	orders_count
1	2018-08	TO	13
2	2018-07	TO	22
3	2018-06	TO	18
4	2018-05	TO	16
5	2018-04	TO	19

21	2018-10	SP	2
22	2018-09	SP	8
23	2018-08	SP	3253
24	2018-07	SP	2777
25	2018-06	SP	2773

3.2 Distribution of customers across the states in Brazil

```
SELECT
      customer_state,
      count(customer_id) as num_of_customers
FROM target.customers
GROUP BY customer_state
order by num_of_customers desc;
```

Row	customer_state	num_of_customers
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
11	PE	1652
12	CE	1336
13	PA	975
14	MT	907
15	MA	747

- We arranged the number of customers column in each state in descending order to easily grasp where target's most number of customers are from. We can infer that the states São Paulo(SP), Rio de Janeiro(RJ) and Minas Gerais(MG) has 66.6% of Target's customer base.

○ Impact on Economy

4.1 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.

```
WITH table1 as
      (SELECT
            format_datetime('%m', o.order_purchase_timestamp) as month,
            format_datetime('%Y', o.order_purchase_timestamp) as year,
            sum(p.payment_value) as total_orders_cost
      FROM target.orders as o
      JOIN target.payments as p
      ON o.order_id = p.order_id
      AND format_datetime('%Y', o.order_purchase_timestamp) IN ('2017','2018')
      WHERE format_datetime('%m', o.order_purchase_timestamp) NOT IN
      ('09','10','11','12')
      GROUP BY month, year
      order by year, month
      )
SELECT
      t1.month,
      round(((t2.total_orders_cost - t1.total_orders_cost)/
      t1.total_orders_cost)*100,0) as percent_change_in_cost
FROM table1 as t1
JOIN  table1 as t2
ON t1.month = t2.month
AND t1.year = '2017' and t2.year = ‘2018'
order by t1.month;
```

Row	month	percent_change_in_cost
1	1	705,0
2	2	240,0
3	3	158,0
4	4	178,0
5	5	95,0
6	6	100,0
7	7	80,0
8	8	52,0

- The percentage increase in cost of orders/sales between the same months of years 2017 and 2018 has been computed and represented in the above table. It can be noted that the percent of change in the month of Jan 2017 and 2018 is the most significant with sales amount increasing by 705% in year 2018.
- The percent of change in sales/order costs is also significant, but is lower.

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

```
SELECT
      c.customer_state,
      round(sum(it.price),2) as total_price,
      round(sum(it.freight_value),2) as total_freight
FROM target.orders as o
JOIN target.order_items as it
ON it.order_id = o.order_id
JOIN target.customers as c
ON c.customer_id = o.customer_id
group by c.customer_state
order by total_price desc;
```

Row	customer_state	total_price	total_freight
1	SP	5202955,05	718723,07
2	RJ	1824092,67	305589,31
3	MG	1585308,03	270853,46
4	RS	750304,02	135522,74
5	PR	683083,76	117851,68
6	SC	520553,34	89660,26

7	BA	511349,99	100156,68
8	DF	302603,94	50625,5
9	GO	294591,95	53114,98
10	ES	275037,31	49764,6

- These values are similar to the customer distribution across states. The order value of São Paulo(SP), Rio de Janeiro(RJ) and Minas Gerais(MG) states is the highest. The total freight value of Sao Paulo is 13.8% of the total price.

○ Analysis on sales, freight and delivery time

5.1 Calculate days between purchasing, delivering and estimated delivery

```
SELECT
      order_id,
      DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as
      actual_delivery_time,
      DATE_DIFF(order_estimated_delivery_date,order_purchase_timestamp,DAY) as
      estimated_delivery_time
FROM target.orders
where order_delivered_customer_date is not null;
```

Row	order_id	actual_delivery_time	estimated_delivery_time
1	1950d777989f6a877539f53795b4c3c3	30	17
2	2c45c33d2f9cb8ff8b1c86cc28c11c30	30	59
3	65d1e226dfaeb8cdc42f665422522d14	35	52
4	635c894d068ac37e6e03dc54eccb6189	30	32
5	3b97562c3aee8bdedcb5c2e45a50d5e1	32	33
6	68f47f50f04c4cb6774570cfde3a9aa7	29	31
7	276e9ec344d3bf029ff83a161c6b3ce9	43	39
8	54e1a3c2b97fb0809da548a59f64c813	40	36
9	fd04fa4105ee8045f6a0139ca5b49f27	37	35
10	302bb8109d097a9fc6e9cefc5917d1f3	33	28

5.2 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_DIFF(order_purchase_timestamp,order_delivered_customer_date,DAY) as
      time_to_delivery,
      DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY)
      as diff_estimated_delivery
 FROM target.orders ;
```

Row	time_to_delivery	diff_estimated_delivery
1	-30	-12
2	-30	28
3	-35	16
4	-30	1
5	-32	0
6	-29	1
7	-43	-4
8	-40	-4
9	-37	-1
10	-33	-5

- A negative value in diff_estimated_delivery indicates that the product was delivered with a delay. A total of 6535 orders were delivered with a delay out of 99441 i.e 7% percent of total delivered orders were delayed.

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

```
SELECT
      c.customer_state,
      round(avg(freight_value)) as mean_freight_value,
      round(avg(DATE_DIFF(order_delivered_customer_date,
      order_purchase_timestamp,DAY))) as avgtime_to_delivery,
      round(avg(DATE_DIFF(order_estimated_delivery_date,
order_delivered_customer_date, DAY))) as avgdiff_estimated_delivery
 FROM target.orders as o
 JOIN target.customers as c
 ON o.customer_id = c.customer_id
 JOIN target.order_items as it
 ON it.order_id = o.order_id
 group by c.customer_state;
```

Row	customer_ state	mean_freight_ value	avgtime_to_ delivery	avgdiff_estimated_ delivery
1	MT	28,0	18	14,0
2	MA	38,0	21	9,0
3	AL	36,0	24	8,0
4	SP	15,0	8	10,0
5	MG	21,0	12	12,0
6	PE	33,0	18	13,0
7	RJ	21,0	15	11,0
8	DF	21,0	13	11,0
9	RS	22,0	15	13,0
10	SE	37,0	21	9,0

- We have an average freight cost of 33.4 across all states. Maranhao(MA) Alagoas(AL) and Sergipe(SE) are the states that have the highest freight charges. The no of order/customers per state seem to have a correlation with the freight charges levied. Rather, the states with most traffic have lower freight charges.

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

```
SELECT
      c.customer_state,
      round(avg(freight_value)) as mean_freight_value,
      round(avg(DATE_DIFF(order_purchase_timestamp,
      order_delivered_customer_date,DAY))) as avgtime_to_delivery,
      round(avg(DATE_DIFF(order_estimated_delivery_date,
      order_delivered_customer_date, DAY))) as avgdiff_estimated_delivery
 FROM target.orders as o
 JOIN target.customers as c
 ON o.customer_id = c.customer_id
 JOIN target.order_items as it
 ON it.order_id = o.order_id
 group by c.customer_state
 order by mean_freight_value asc limit 5;
```

Row	customer _state	mean_freight_ value	avgtime_to_ delivery	avgdiff_estimated_ delivery
1	SP	15,0	8	10,0
2	PR	21,0	11	13,0
3	RJ	21,0	15	11,0
4	DF	21,0	13	11,0
5	MG	21,0	12	12,0

```
SELECT
      c.customer_state,
      round(avg(freight_value)) as mean_freight_value,
      round(avg(DATE_DIFF(order_purchase_timestamp,
      order_delivered_customer_date,DAY))) as avgtime_to_delivery,
      round(avg(DATE_DIFF(order_estimated_delivery_date,
      order_delivered_customer_date, DAY))) as avgdiff_estimated_delivery
 FROM target.orders as o
 JOIN target.customers as c
 ON o.customer_id = c.customer_id
 JOIN target.order_items as it
 ON it.order_id = o.order_id
 group by c.customer_state
 order by mean_freight_value desc limit 5;
```

Row	customer_state	mean_freight _value	avgtime_to _delivery	avgdiff_estimated _delivery
1	PB	43,0	20	12,0
2	RR	43,0	28	17,0
3	RO	41,0	19	19,0
4	AC	40,0	20	20,0
5	PI	39,0	19	11,0

5.6 Top 5 states with highest/lowest average time to delivery

```
SELECT
      c.customer_state,
      round(avg(DATE_DIFF(order_purchase_timestamp,
      order_delivered_customer_date,DAY))) as avgtime_to_delivery,
      round(avg(DATE_DIFF(order_estimated_delivery_date,
      order_delivered_customer_date, DAY))) as avgdiff_estimated_delivery
 FROM target.orders as o
 JOIN target.customers as c
 ON o.customer_id = c.customer_id
 group by c.customer_state
 order by avgtime_to_delivery desc limit 5;
```

Row	customer_state	avgtime_to_delivery	avgdiff_estimated_delivery
1	SP	8	10,0
2	PR	12	12,0
3	MG	12	12,0
4	DF	13	11,0
5	SC	14	11,0

```
SELECT
      c.customer_state,
      round(avg(DATE_DIFF(order_delivered_customer_date,
            order_purchase_timestamp, DAY))) as avgtime_to_delivery,
      round(avg(DATE_DIFF(order_estimated_delivery_date,
            order_delivered_customer_date, DAY))) as avgdiff_estimated_delivery
 FROM target.orders as o
 JOIN target.customers as c
 ON o.customer_id = c.customer_id
 group by c.customer_state
 order by avgtime_to_delivery limit 5;
```

Row	customer_state	avgtime_to_ delivery	avgdiff_estimated_ delivery
1	RR	29	16,0
2	AP	27	19,0
3	AM	26	19,0
4	AL	24	8,0
5	PA	23	13,0

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

- <u>Top 5 states with really fast delivery speed</u>

```
SELECT
      c.customer_state,
      round(avg(DATE_DIFF(order_delivered_customer_date,
order_purchase_timestamp,DAY))) as avgtime_to_delivery,
      round(avg(DATE_DIFF(order_estimated_delivery_date,
order_delivered_customer_date, DAY))) as avgdiff_estimated_delivery
 FROM target.orders as o
 JOIN target.customers as c
 ON o.customer_id = c.customer_id
 group by c.customer_state
 order by avgdiff_estimated_delivery limit 5;
```

Row	customer _state	avgtime_to _delivery	avgdiff_estimated _delivery

1	RR	29,0	16,0
2	AP	27,0	19,0
3	AM	26,0	19,0
4	AL	24,0	8,0
5	PA	23,0	13,0

- Top 5 states with lower delivery speed with respect to estimated date.

```
SELECT
      c.customer_state,
      round(avg(DATE_DIFF(order_delivered_customer_date,
                  order_purchase_timestamp,DAY))) as avgtime_to_delivery,
      round(avg(DATE_DIFF(order_estimated_delivery_date,
            order_delivered_customer_date, DAY))) as avgdiff_estimated_delivery
 FROM target.orders as o
 JOIN target.customers as c
 ON o.customer_id = c.customer_id
 group by c.customer_state
 order by avgdiff_estimated_delivery desc limit 5;
```

Row	customer_state	avgtime_to _delivery	avgdiff_estimated _delivery
1	AC	21,0	20,0
2	RO	19,0	19,0
3	AM	26,0	19,0
4	AP	27,0	19,0
5	RR	29,0	16,0

6. Payment type analysis

6.1 Month over Month count of orders for different payment types

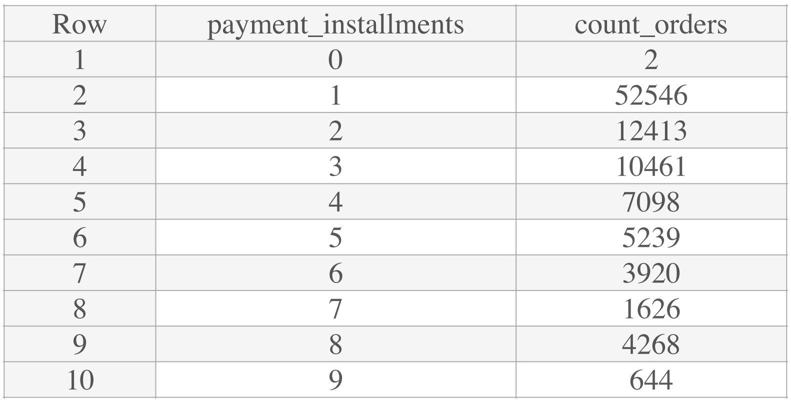
```
SELECT
      p.payment_type,
      count(o.order_id) as count_orders
FROM target.orders as o
JOIN target.payments as p
ON o.order_id = p.order_id
GROUP BY p.payment_type;
```

Row	payment_type	count_orders
1	UPI	19784
2	credit_card	76795
3	voucher	5775
4	debit_card	1529
5	not_defined	3

- It is evident that credit cards are the most preferred payment method with a whopping 73.92% of payments being made through credit cards.

6.2 Count of orders based on the no. of payment installments

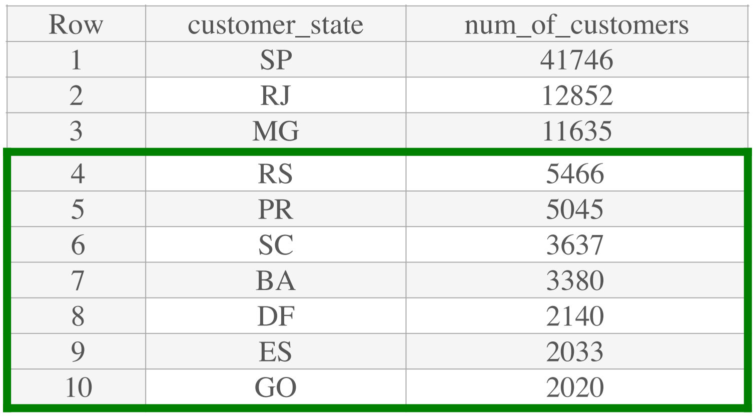
```
SELECT
      p.payment_installments,
      count(o.order_id) as count_orders
FROM target.orders as o
JOIN target.payments as p
ON o.order_id = p.order_id
GROUP BY p.payment_installments
ORDER BY p.payment_installments;
```



Row	payment_installments	count_orders
1	0	2
2	1	52546
3	2	12413
4	3	10461
5	4	7098
6	5	5239
7	6	3920
8	7	1626
9	8	4268
10	9	644

Actionable insights

1. Looking at the month on month sales over a period of 2 years, it is evident that there is a growing trend in e-commerce in Brazil, the YoY for 2017-18 has seen a 19.75% growth. Specifically there is a rise in sales in the first eight months of the year, the pace of business slows for a couple of months and there is a spike in November. It is suggested to roll out some offers or launch a marketing campaign for the months of September and October to increase the sales further.



Row	customer_state	num_of_customers
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

2. Further, 40.41% of Brazilian customers make purchases at night and another 32.4% customers make purchases in the afternoon.
3. The states São Paulo(SP), Rio de Janeiro(RJ) and Minas Gerais(MG) has 66.6% of Target's customer base. We could run a marketing campaign or study further on the issues faced in these states to generate more customers and orders from the states in rows through 4-10.

4. 7% percent of total delivered orders were delayed. Furthermore, taking an average delivery time of 12 days for the states with most customers is not efficient. The delivery period also accounts to a customer's overall experience with Target. I would recommend that we do further research on delivery time of our retail competitors and at the same time ramp up our delivery efficiency.

5. It is apparent from the data that credit cards are the most used method of payment with 73.92% of total payments being made through credit cards. While credit cards have security/fraud risks it may not be advisable to have majority of revenue dependant in credit cards. We could offer some incentives to make payments through UPI or debit cards which is a risk free transaction for the business and a direct sale that cannot be disputed.