

## 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
2. Time period for which the data is given

```
with cte_1 as (  
    select min(order_purchase_timestamp) as first_date,  
           max(order_purchase_timestamp) as last_date  
    from dbo.orders)  
select concat(year(first_date), '/', month(first_date), '/', day(first_date)) as start_of_time_period,  
       concat(year(last_date), '/', month(last_date), '/', day(last_date)) as end_of_time_period  
from cte_1;
```

	start_of_time_period	end_of_time_period
1	2016/9/4	2018/10/17

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

```
select count(*) as total_cities from (select distinct customer_city from dbo.customers) t;
```

	total_cities
1	4119

```
select count(*) as total_states from (select distinct customer_state from dbo.customers) t;
```

	total_states
1	27

	customer_state
1	PE
2	PB
3	PA
4	RS
5	AC
6	BA
7	SP
8	SC
9	SE
10	MA

	customer_city
1	santa izabel do oeste
2	claudia
3	sao jose dos cordeiros
4	santa salete
5	andradina
6	buritis
7	santa quiteria
8	deodapolis
9	mostardas
10	novo jardim

## 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?

```
with cte_1 as (select order_id, year(order_purchase_timestamp) as year1,
month(order_purchase_timestamp) as month1 from dbo.orders where order_status <> 'canceled')
select distinct year1, month1, count(*) over(partition by year1,month1) as order_count from cte_1
order by year1,month1;
```

	year1	month1	order_count
1	2016	9	2
2	2016	10	300
3	2016	12	1
4	2017	1	797
5	2017	2	1763
6	2017	3	2649
7	2017	4	2386
8	2017	5	3671
9	2017	6	3229
10	2017	7	3998

```
with cte_1 as
(select order_id, year(order_purchase_timestamp) as year,
month(order_purchase_timestamp) as month,
case when month(order_purchase_timestamp) in (1,2,3) then 1
when month(order_purchase_timestamp) in (4,5,6) then 2
when month(order_purchase_timestamp) in (7,8,9) then 3
when month(order_purchase_timestamp) in (10,11,12) then 4
end as quarter from dbo.orders where order_status <> 'canceled')
select distinct year, quarter,
count(*) over(partition by year,quarter) as order_count
from cte_1 order by year,quarter;
```

	year	quarter	order_count
1	2016	3	2
2	2016	4	301
3	2017	1	5209
4	2017	2	9286
5	2017	3	12567
6	2017	4	17774
7	2018	1	21075
8	2018	2	19922
9	2018	3	12680

```
with cte_1 as (select order_id, year(order_purchase_timestamp) as year,
month(order_purchase_timestamp) as month from dbo.orders where order_status <> 'canceled')
select distinct year, count(*) over(partition by year) as order_count from cte_1
order by year;
```

	year	order_count
1	2016	303
2	2017	44836
3	2018	53677

Looks like Target just started their operations in Brazil from 2016 September. Considering the initial difficulties to setup the operations, Target did not get much orders in first 3-4 months. From 2017, orders started flowing in and gradually kept on increasing. We can say that e-commerce trend is increasing YOY and QOQ in Brazil. With the given data, I don't see any seasonality.

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

```
with cte_2 as(select order_id, order_status,
convert(varchar, order_purchase_timestamp, 108) as time from dbo.orders),
cte_3 as (select *,case
when time between '04:00:00' and '05:59:59' then 'Dawn'
when time between '06:00:00' and '11:59:59' then 'Morning'
when time between '12:00:00' and '18:00:00' then 'Afternoon'
when time between '18:00:01' and '23:59:59' or time between '00:00:00' and '03:59:59' then 'Night'
end as times_of_the_day from cte_2)
select distinct times_of_the_day, count(*) over(partition by times_of_the_day) as order_count
from cte_3 order by order_count desc;
```

100 %

Results Messages

	times_of_the_day	order_count
1	Night	38442
2	Afternoon	38365
3	Morning	22240
4	Dawn	394

It is clear that Brazilians equally prefers to place most orders at night and afternoon and least in Dawn.

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

#### 1. Get month on month orders by states

```

select * from (select order_id, c.customer_state, year(o.order_purchase_timestamp) as YEAR,
month(o.order_purchase_timestamp) as MONTH from dbo.customers c
join dbo.orders o on c.customer_id = o.customer_id) t
PIVOT (count(order_id) for customer_state in([AC],[AL],[AM],[AP],[BA],[CE],[DF],[ES],[GO],[MA],[MG],[MS],[MT],
[PA],[PB],[PE],[PI],[PR],[RJ],[RN],[RO],[RR],[RS],[SC],[SE],[SP],[TO])) as pvt
order by pvt.YEAR, pvt.MONTH;

```

100 %

Results

Messages

	YEAR	MONTH	AC	AL	AM	AP	BA	CE	DF	ES	GO	MA	MG	MS	MT	PA	PB	PE	PI	PR	RJ	RN	RO	RR	RS	SC	SE	SP	TO
1	2016	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	2	0	
2	2016	10	0	2	0	0	4	8	6	4	9	4	40	0	3	4	1	7	1	19	56	4	0	1	24	11	3	113	0
3	2016	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
4	2017	1	2	2	0	0	25	9	13	12	18	9	108	1	11	12	2	9	7	65	97	5	3	0	54	31	4	299	2
5	2017	2	3	12	8	2	59	13	24	34	27	11	259	11	17	25	12	21	12	118	254	8	11	2	105	59	12	654	7
6	2017	3	2	10	5	3	91	28	57	48	53	24	358	20	16	36	16	45	13	127	395	13	16	2	151	110	25	1010	8
7	2017	4	5	23	13	0	93	43	35	46	41	27	275	15	27	36	20	40	13	114	338	10	9	2	139	105	13	908	14
8	2017	5	8	27	10	5	127	62	64	94	87	33	428	29	37	35	18	68	25	213	488	17	9	2	208	152	11	1425	18
9	2017	6	4	10	1	2	106	47	70	80	79	17	363	27	25	38	23	46	14	170	412	13	10	3	221	116	9	1331	8
10	2017	7	5	17	5	1	155	53	77	83	77	39	453	25	38	39	27	73	20	203	571	27	11	1	249	158	14	1604	1

Overall, SP (São Paulo) has maximum number of orders in all months.

#### 2. Distribution of customers across the states in Brazil

```
select *, round(cast(customer_count as float)*100/total_customers,2) as percent_of_cust from
(select *, sum(customer_count) over() as total_customers from
(select customer_state, count(distinct customer_unique_id) customer_count
from dbo.customers
group by customer_state) t1) t2
order by customer_count desc;
```

	customer_state	customer_count	total_customers	percent_of_cust
1	SP	40302	96136	41.92
2	RJ	12384	96136	12.88
3	MG	11259	96136	11.71
4	RS	5277	96136	5.49
5	PR	4882	96136	5.08
6	SC	3534	96136	3.68
7	BA	3277	96136	3.41
8	DF	2075	96136	2.16
9	ES	1964	96136	2.04
10	GO	1952	96136	2.03

```
select *, round(cast(customer_count as float)*100/total_customers,2) as percent_of_cust from
(select *, sum(customer_count) over() as total_customers from
(select customer_state, count(distinct customer_unique_id) customer_count
from dbo.customers
group by customer_state) t1) t2
order by customer_count;
```

	customer_state	customer_count	total_customers	percent_of_cust
1	RR	45	96136	0.05
2	AP	67	96136	0.07
3	AC	77	96136	0.08
4	AM	143	96136	0.15
5	RO	240	96136	0.25
6	TO	273	96136	0.28
7	SE	342	96136	0.36
8	AL	401	96136	0.42
9	RN	474	96136	0.49
10	PI	482	96136	0.5

From above results, SP (São Paulo) has highest number of customers and RR (Roraima) has least number of customers. Also, the result is pretty much proportional to the population of individual states.

#### 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

```
-- Get average order value in 2017 and 2018
select distinct avg(p.payment_value) over(partition by year(o.order_purchase_timestamp)) as order_value,
year(o.order_purchase_timestamp) as Year from dbo.orders o
join dbo.payments p on o.order_id = p.order_id
where year(o.order_purchase_timestamp) in ('2017','2018')
and month(o.order_purchase_timestamp) in ('1','2','3','4','5','6','7','8');

-- Get percent increase in order value from 2017 to 2018
with cte as (select distinct avg(p.payment_value) over(partition by
year(o.order_purchase_timestamp)) as order_value,
year(o.order_purchase_timestamp) as Year from dbo.orders o
join dbo.payments p on o.order_id = p.order_id
where year(o.order_purchase_timestamp) in ('2017','2018')
and month(o.order_purchase_timestamp) in ('1','2','3','4','5','6','7','8')),
cte1 as (select ((order_value - lag(order_value) over(order by Year))/lag(order_value)
over(order by Year))*100 as percent_increase from cte)
select * from cte1 where percent_increase is not null;
```

	order_value	Year
1	150.42524367513	2017
2	155.276968331494	2018

	percent_increase
1	3.225339403034

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

```
select c.customer_state, avg(oi.price) as mean_price, sum(oi.price) as sum_price,
avg(oi.freight_value) as mean_freight, sum(oi.freight_value) as sum_freight from customers c
join orders o on c.customer_id = o.customer_id
join order_items oi on o.order_id = oi.order_id
group by c.customer_state
order by customer_state;
```

	customer_state	mean_price	sum_price	mean_freight	sum_freight
1	AC	173.727717358133	15982.9499969482	40.0733695548514	3686.74999904633
2	AL	180.889211497865	80314.8099050522	35.8436711693669	15914.5899991989
3	AM	135.49600017432	22356.8400287628	33.2053939530344	5478.89000225067
4	AP	164.320731325847	13474.2999687195	34.006097584236	2788.50000190735
5	BA	134.60120810029	511349.989573002	26.3639589913388	100156.680208096
6	CE	153.758261139564	227254.709964275	32.7142015664923	48351.5899152756
7	DF	125.770548605859	302603.939945698	21.041354959161	50625.5000317413
8	ES	121.913701212998	275037.309936523	22.0587766029928	49764.6000163518
9	GO	126.271731597591	294591.949817181	22.7668153039067	53114.9801040143
10	MA	145.20415049502	119648.220007896	38.2570024183364	31523.7699927092

## 5. Analysis on sales, freight and delivery time

```

with cte_1 as (select o.order_id, customer_state,
    CONVERT(VARCHAR(10), order_purchase_timestamp, 111) as purchase_date,
    CONVERT(VARCHAR(10), order_delivered_customer_date, 111) as delivered_date,
    CONVERT(VARCHAR(10), order_estimated_delivery_date, 111) as estimated_date,
    freight_value from orders o
join customers c on o.customer_id = c.customer_id
join order_items oi on o.order_id = oi.order_id
where order_delivered_customer_date is not null)
select top 5 customer_state, avg(freight_value) as mean_freight,
    avg(datediff(day,purchase_date,delivered_date)) as time_to_delivery,
    avg(datediff(day,estimated_date,delivered_date)) as diff_estimated_delivery
from cte_1 group by customer_state

```

1. Calculate days between purchasing, delivering and estimated delivery
2. Find time\_to\_delivery & diff\_estimated\_delivery. Formula for the same given below:
  - $\text{time\_to\_delivery} = \text{order\_purchase\_timestamp} - \text{order\_delivered\_customer\_date}$
  - $\text{diff\_estimated\_delivery} = \text{order\_estimated\_delivery\_date} - \text{order\_delivered\_customer\_date}$
3. Group data by state, take mean of freight\_value, time\_to\_delivery, diff\_estimated\_delivery
4. Sort the data to get the following:
5. Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

	customer_state	mean_freight	time_to_delivery	diff_estimated_delivery
1	PB	43.0916894632598	20	-13
2	RR	43.088043378747	28	-18
3	RO	41.3305494986174	19	-20
4	AC	40.0479120841393	20	-20
5	PI	39.1150860959668	19	-11

Above states have the highest freight value.

	customer_state	mean_freight	time_to_delivery	diff_estimated_delivery
1	SP	15.1149940990789	8	-11
2	PR	20.4718162916188	11	-13
3	MG	20.6258373119181	11	-13
4	RJ	20.9097844371546	15	-12
5	DF	21.0721613708963	12	-12

Above states have the lowest freight value.

## 6. Top 5 states with highest/lowest average time to delivery

	customer_state	mean_freight	time_to_delivery	diff_estimated_delivery
1	RR	43.088043378747	28	-18
2	AP	34.160493850708	28	-18
3	AM	33.3106135093361	26	-19
4	AL	35.8706557410104	24	-8
5	PA	35.6290132626184	23	-14

Above states have highest average time to delivery.

	customer_state	mean_freight	time_to_delivery	diff_estimated_delivery
1	SP	15.1149940990789	8	-11
2	PR	20.4718162916188	11	-13
3	MG	20.6258373119181	11	-13
4	DF	21.0721613708963	12	-12
5	SC	21.5066276626937	14	-11

Above states have lowest average time to delivery.

## 7. Top 5 states where delivery is really fast/ not so fast compared to estimated date

	customer_state	mean_freight	time_to_delivery	diff_estimated_delivery
1	AL	35.8706557410104	24	-8
2	MA	38.4927124927938	21	-9
3	BA	26.4875563981925	19	-10
4	SE	36.5731733792623	21	-10
5	ES	22.0289797848245	15	-10

Delivery in above states is not so fast compared to estimated date.

	customer_state	mean_freight	time_to_delivery	diff_estimated_delivery
1	AC	40.0479120841393	20	-20
2	RO	41.3305494986174	19	-20
3	AM	33.3106135093361	26	-19
4	RR	43.088043378747	28	-18
5	AP	34.160493850708	28	-18

Delivery in above states is really fast compared to estimated date.

## 6. Payment type analysis:

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

```
select * from (select p.order_id, payment_type, month(order_purchase_timestamp) as Month,
year(order_purchase_timestamp) as Year from payments p
join orders o on p.order_id=o.order_id
where payment_value <> 0 and year(order_purchase_timestamp) = '2017'
or (month(order_purchase_timestamp) not in ('10','9') and year(order_purchase_timestamp) = '2018'))t
PIVOT (count(order_id) for payment_type in ([credit_card],[debit_card],[UPI],[voucher])) pvt
order by Year,Month;
```

100 %

Results Messages

	Month	Year	credit_card	debit_card	UPI	voucher
1	1	2017	583	9	197	61
2	2	2017	1356	13	398	119
3	3	2017	2016	31	590	200
4	4	2017	1846	27	496	200
5	5	2017	2853	30	772	288
6	6	2017	2463	27	707	238
7	7	2017	3086	22	845	364
8	8	2017	3284	34	938	294
9	9	2017	3283	43	903	287
10	10	2017	3524	52	993	290

Credit card is the most preferred payment type and it is increasing faster compared to other payment types.

### 2. Count of orders based on the no. of payment installments

```
select count(order_id) as no_of_orders ,payment_installments from payments
group by payment_installments
order by no_of_orders desc;
```

100 %

Results Messages

	no_of_orders	payment_installments
1	52546	1
2	12413	2
3	10461	3
4	7098	4
5	5328	10
6	5239	5
7	4268	8
8	3920	6
9	1626	7
10	644	9

Almost half of the orders are paid in one instalment.



➤ **Recommendations and Actionable Insights**

- Target can open more warehouses/stores to reduce the freight value and delivery time.
- Target can partner with banks for cashback offers on credit cards.
- Target can give free delivery over a certain order value to increase the average payment value significantly.
- Should focus more on increasing average payment value and decreasing average delivery time.