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

1.1Data type of all columns in the "customers" table.

```
SELECT
  column_name,
  data_type
FROM
  snappy-axle-414907.target.INFORMATION_SCHEMA.COLUMNS
WHERE
  table_name = 'customers';
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXEC

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: Most of the data types in customers table columns are STRING

1.2.Get the time range between which the orders were placed.

```
select
min(order_purchase_timestamp) as start_date,
max(order_purchase_timestamp) as end_date from target.orders
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXE

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

Insights: orders were placed in time range between 4th september 2016 and 17th october 2018.

1.3.Count the Cities & States of customers who ordered during the given period.

```
select count(distinct customer_city) as num_of_cities,count(distinct
customer_state) as num_of_states from target.customers
where customer_id in (select customer_id from target.orders)
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	num_of_cities	num_of_states
1	4119	27

Insights : There are 4119 number of cities from 27 states which are placed orders in given period of time.

2.In_depth Exploration

2.1.Is there a growing trend in the no. of orders placed over the past years?

```
select extract(year from order_purchase_timestamp) as order_year,count(order_id)as
orders_count from target.orders
group by extract(year from order_purchase_timestamp)
order by order_year;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	order_year	orders_count
1	2016	329
2	2017	45101
3	2018	54011

Insights: Trend is increasing every year.There is massive jump in orders count in between 2016 and 2017 and there is decent increase in 2018.

2.2Can we see some kind of monthly seasonality in terms of the no. of orders being placed?

```
select extract(year from order_purchase_timestamp) as order_year,
extract(month from order_purchase_timestamp) as order_month,
count(order_id)as order_count from target.orders
group by extract(year from order_purchase_timestamp),extract(month from
order_purchase_timestamp)
order by order_count
```

Query results

SAVE RESULTS EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	order_year	order_month	order_count
1	2016	12	1
2	2016	9	4
3	2018	10	4
4	2018	9	16
5	2016	10	324
6	2017	1	800
7	2017	2	1780
8	2017	4	2404
9	2017	3	2682
10	2017	6	3245

Results per page: 50 1 – 25 of 25

Insight: Count of orders is decreased in month of september ,october in every year.

2.3During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

```
select case
when extract(hour from order_purchase_timestamp) between 0 and 6 then 'Dawn'
when extract(hour from order_purchase_timestamp) between 7 and 12 then 'Mornings'
when extract(hour from order_purchase_timestamp) between 13 and 18 then 'Afternoon'
when extract(hour from order_purchase_timestamp) between 19 and 23 then 'Night' end
as time_of_the_day,
count(*)as order_count
from target.orders
group by time_of_the_day
order by order_count
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	time_of_the_day	order_count
1	Dawn	5242
2	Mornings	27733
3	Night	28331
4	Afternoon	38135

Insights: Most of the Orders placed in afternoon time with order count 38135.Order count of mornings and night are almost same with difference is less than 1000 orders.

Recommendations: In the month of september, october orders are decreased in very huge amounts. They should focus more on what customers require and make necessary changes.

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

3.1.Get the month on month no. of orders placed in each state.

```
select c.customer_state,extract(month from order_purchase_timestamp) as month
 from target.customers c
join target.orders o
on c.customer_id=o.customer_id
group by c.customer_state,extract(month from order_purchase_timestamp)
order by order_count desc,month asc
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_state	month	order_count
1	SP	8	4982
2	SP	5	4632
3	SP	7	4381
4	SP	6	4104
5	SP	3	4047
6	SP	4	3967
7	SP	2	3357
8	SP	1	3351
9	SP	11	3012
10	SP	12	2357
11	SP	10	1908
12	SP	9	1648

Results per page: 50 | 1 – 50 of 322

Insights: In every individual Month Maximum numbers of orders are coming from SP state.Follwed by RJ in some months and MG in some months.

3.2.How are the customers distributed across all the states?

```
select customer_state,count(distinct customer_id) as no_of_customers from
target.customers
group by customer_state
order by no_of_customers desc
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_state	no_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

Results per page: 50 | 1 – 27 of 27

Insights : Maximum numbers of customers are from SP state which has 41746 customers which is around 42% of the total customers.Followed by RJ and MJ which have close number of customers in there states.
There are around 15 states which has below 1000 customers in there states in that RR,AP,AC has number of customers has below 100.

Recommandations : Maximum customers are from SP state and so orders more placed are from SP state.There are states like RR,AP,AC very less customers which are below 100 and around 15 states customers are below 1000 so they should focus more on these states what went wrong in so less customers.

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

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

```
select
round((
(
sum(case when extract(year from order_purchase_timestamp)=2018 and extract(month
from order_purchase_timestamp) between 1 and 8 then p.payment_value end )-
sum(case when extract(year from order_purchase_timestamp)=2017 and extract(month
from order_purchase_timestamp) between 1 and 8 then p.payment_value end))
/sum(case when extract(year from order_purchase_timestamp)=2017 and extract(month
from order_purchase_timestamp) between 1 and 8 then p.payment_value end ))*100,2)
as increase_in_percentage
from target.orders o
join target.payments p
on o.order_id=p.order_id
where extract(year from order_purchase_timestamp) in (2017,2018) and extract(month
from order_purchase_timestamp) between 1 and 8
```

Query results

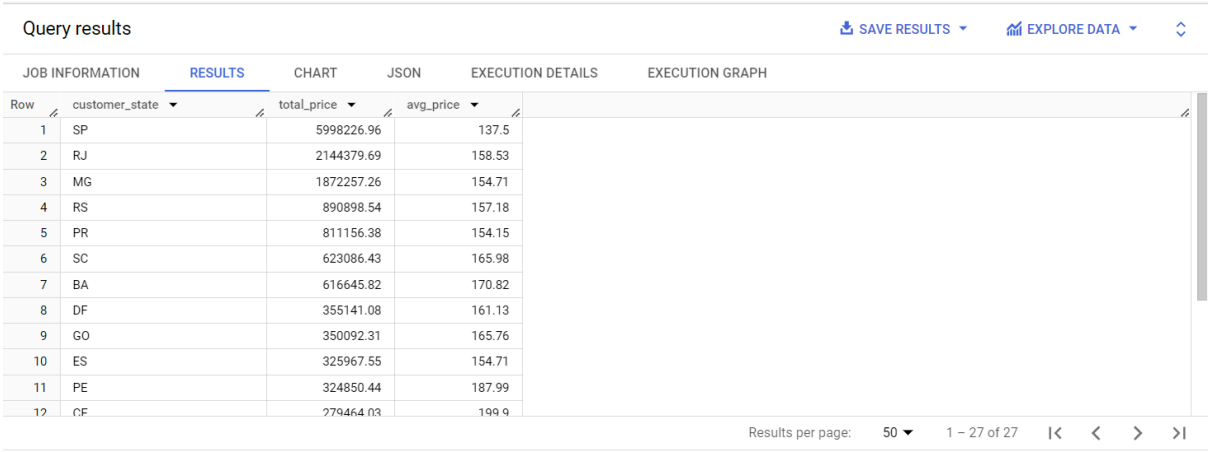
JOB INFORMATION | RESULTS | CHART

Row	increase_in_percentage
1	136.98

Insights : There is total of 136.98% increase from 2017 to 2018 in January to August months.In every individual months january has highest increase percentage and August has lowest increase percentage amongst the months from 2017 and 2018.

4.2.Calculate the Total & Average value of order price for each state.

```
select c.customer_state,round(sum(p.payment_value),2)
total_price,round(avg(p.payment_value),2) as avg_price from target.customers c
join target.orders o
on c.customer_id =o.customer_id
join target.payments p
on o.order_id=p.order_id
group by c.customer_state
order by total_price desc
```


Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_state	total_price	avg_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
11	PE	324850.44	187.99
12	CE	279464.03	199.9

Results per page: 50 | 1 – 27 of 27

Insights : Maximum revenue is coming from SP state and next state total price is not even has half of the SP total price.

4.3.Calculate the Total & Average value of order freight for each state.

```
select c.customer_state,round(sum(oi.freight_value),2)
total_value,round(avg(oi.freight_value),2) as avg_value from target.customers c
join target.orders o
on c.customer_id =o.customer_id
join target.order_items oi
on o.order_id=oi.order_id
group by c.customer_state
order by total_value desc,avg_value desc
```

Query results

SAVE RESULTS EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_state	total_value	avg_value
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
11	ES	49764.6	22.06

Results per page: 50 1 – 27 of 27

Insights : SP state has highest total of freight value of total orders.

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

5.1.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.

```
select order_id,
date_diff(order_delivered_customer_date,order_purchase_timestamp,day ) as
time_to_deliver,
date_diff(order_delivered_customer_date,order_estimated_delivery_date,day) as
diff_estimated_delivery
 from target.orders
 where date_diff(order_delivered_customer_date,order_purchase_timestamp,day ) is
not null
```

Query results

SAVE RESULTS EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	order_id	time_to_deliver	diff_estimated_delive
1	1950d777989f6a877539f5379...	30	12
2	2c45c33d2f9cb8ff8b1c86cc28...	30	-28
3	65d1e226dfaeb8cdc42f66542...	35	-16
4	635c894d068ac37e6e03dc54e...	30	-1
5	3b97562c3aee8bdedcb5c2e45...	32	0
6	68f47f50f04c4cb6774570cfde...	29	-1
7	276e9ec344d3bf029ff83a161c...	43	4
8	54e1a3c2b97fb0809da548a59...	40	4
9	fd04fa4105ee8045f6a0139ca5...	37	1
10	302bb8109d097a9fc6e9cefc5...	33	5
11	66057d37308e787052a32828...	38	6

Results per page: 50 1 – 50 of 96476

Insights : Here we can see some of the orders are delivered before the estimated delivery date and some of the orders are delivered after the estimated delivery date.

Recommendation : Some of the orders are delivering in near to 180 days. They should focus on these states and deliver in estimated delivery time.

5.2.Find out the top 5 states with the highest & lowest average freight value:

```
with highest as (
select c.customer_state as highest_top5_states,
round(avg(oi.freight_value),2)as highest_avg_freight_value,
row_number()over(order by round(avg(oi.freight_value),2)desc) as row_num
  from target.order_items oi
join target.orders o
on oi.order_id=o.order_id
join target.customers c
on o.customer_id=c.customer_id
group by c.customer_state
order by highest_avg_freight_value desc
limit 5),
lowest as (
  select c.customer_state as lowest_top5_states,
round(avg(oi.freight_value),2)as lowest_avg_freight_value,
row_number()over(order by round(avg(oi.freight_value),2)) as row_num
  from target.order_items oi
join target.orders o
on oi.order_id=o.order_id
join target.customers c
on o.customer_id=c.customer_id
group by c.customer_state
order by lowest_avg_freight_value asc
limit 5)
select
h.highest_top5_states,h.highest_avg_freight_value,l.lowest_top5_states,l.lowest_avg
_freight_value from highest h
join lowest l
on h.row_num=l.row_num
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	highest_top5_states	highest_avg_freight_value	lowest_top5_states	lowest_avg_freight_value
1	RR	42.98	SP	15.15
2	PB	42.72	PR	20.53
3	RO	41.07	MG	20.63
4	AC	40.07	RJ	20.96
5	PI	39.15	DF	21.04

Insights : Top 5 states which has highest average freight value are RR,PB,RO,AC,PI and 5 states that has lowest average freight value are SP,PR,MG,RJ,DF.

5.3.Find out the top 5 states with the highest & lowest average delivery time.

```
with highest as (
  select c.customer_state as highest_top5_states,
round(avg(date_diff(order_delivered_customer_date,order_purchase_timestamp,day)),2)
as highest_avg,
row_number() over(order by
avg(date_diff(order_delivered_customer_date,order_purchase_timestamp,day))desc) as
row_num
 from target.orders o
join target.customers c
on o.customer_id=c.customer_id
group by c.customer_state
order by highest_avg desc
limit 5
),
lowest as (
  select c.customer_state as lowest_top5_states,
round(avg(date_diff(order_delivered_customer_date,order_purchase_timestamp,day)),2)
as lowest_avg,
row_number() over(order by
avg(date_diff(order_delivered_customer_date,order_purchase_timestamp,day))) as
row_num
 from target.orders o
join target.customers c
on o.customer_id=c.customer_id
group by c.customer_state
order by lowest_avg asc
limit 5)
select h.highest_top5_states,h.highest_avg,l.lowest_top5_states,l.lowest_avg from
highest h
join lowest l
on h.row_num=l.row_num
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	highest_top5_states	highest_avg	lowest_top5_states	lowest_avg
1	RR	28.98	SP	8.3
2	AP	26.73	PR	11.53
3	AM	25.99	MG	11.54
4	AL	24.04	DF	12.51
5	PA	23.32	SC	14.48

Insights : RR,AP,AM,AL,PA states has highest average in delivering to customers from order purchase date And SP,PR,MG,DF,SC states has lowest average in delivering to customers from order purchase date.

5.4.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.

```
with cte as (
select c.customer_state,
  round(avg(date_diff(
o.order_delivered_customer_date,o.order_estimated_delivery_date, DAY)), 2) as
diff_estimated_delivery
from target.customers c
join target.orders o
on c.customer_id=o.customer_id
group by c.customer_state
order by diff_estimated_delivery asc
limit 5
)
select customer_state from cte
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_state
1	AC
2	RO
3	AP
4	AM
5	RR

Insights. In AC,RO,AP,AM,RR deliveries are getting faster than estimated delivery date.

6.Analysis based on the payments:

6.1.Find the month on month no. of orders placed using different payment types.

```
select
extract(month from order_purchase_timestamp) as month,
p.payment_type,count(distinct o.order_id) as order_count from target.orders o
join target.payments p
on o.order_id=p.order_id
group by p.payment_type,extract(month from order_purchase_timestamp)
order by month
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

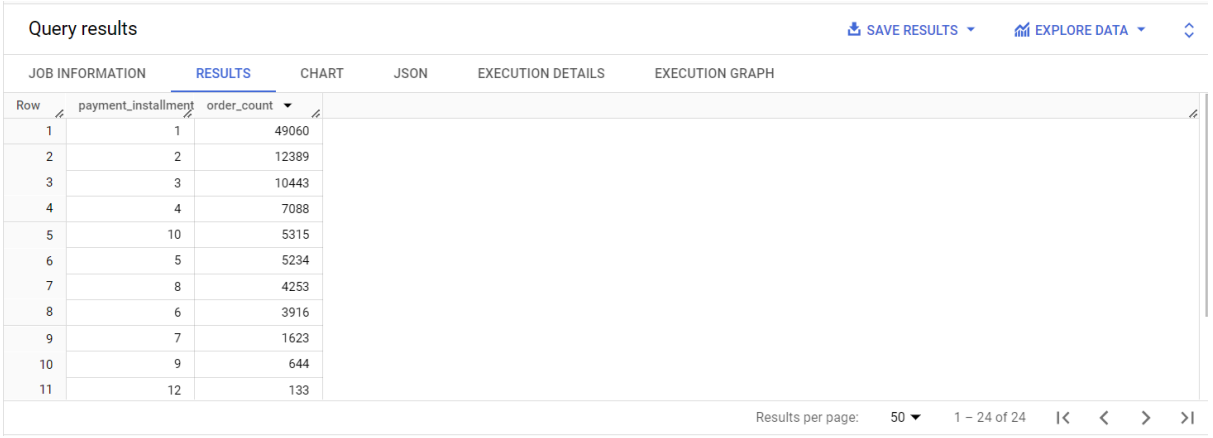
Row	month	payment_type	order_count
1	1	credit_card	6093
2	1	UPI	1715
3	1	voucher	337
4	1	debit_card	118
5	2	credit_card	6582
6	2	UPI	1723
7	2	voucher	288
8	2	debit_card	82
9	3	credit_card	7682
10	3	UPI	1942
11	3	voucher	395
12	3	debit_card	109

Results per page: 50 | 1 – 50 of 50

Insights : By seeing the output in every month credit cards have been used most for payments. upi payment type has been used after credit card.

6.2.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 order_count from
target.payments
group by payment_installments
order by order_count desc
```



Insights :Most of the orders around 50% payment done in 1 installments and remaining are done in 2 to 24 installments.