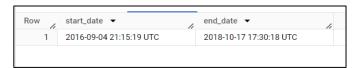
- I. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset.
  - **A.** Data type of all columns in the "customers" table.?



**B.** Get the time range between which the orders were placed?



**C.** Count the Cities & States of customers who ordered during the given period?



# II. In-depth Exploration:

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

Row	year ▼	le	no_of_orders ▼
1		2016	329
2		2017	45101
3		2018	54011

Insights: 1-Here we growing trend between 2016 to 2018.

2-In the year 2016 no. of orders is less because it is the initial stage of the company. After 2016 no. of orders has increased significantly.

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

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

<u>Insights</u>: 1-Seasonality is noticeable in the no. of orders being placed.

- 2-On. Of orders increase from February to August.
- 3-Highest orders placed in the month of August and least in September.
- 4-In September, there was a significant decrease in the number of orders compared to August. However, in October and November, there was a gradual increase in orders. This was followed by another decrease in orders in December.

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

• 0-6 hrs : Dawn

• 7-12 hrs : Mornings

• 13-18 hrs : Afternoon

• 19-23 hrs : Night

```
Code: - 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 "Morning"

when extract (hour from order_purchase_timestamp) between 13 and 18 then "Afternoon"

when extract(hour from order_purchase_timestamp) between 19 and 24 then "Night"end ) as Time, count(order_id) as no_of_orders from `target.orders` group by 1 order by 2 desc;
```

Row	Time ▼	no_of_orders ▼
1	Afternoon	38135
2	Night	28331
3	Morning	27733
4	Dawn	5242

- <u>Insights</u>: 1- Afternoon time (13-18 hrs) is the time where most orders are placed followed by night time (19-23 hrs) closely followed by morning time (7-12 hrs).
  - 2- Dawn (0-6 hrs) is the least preferred time for the customers to place orders

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

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

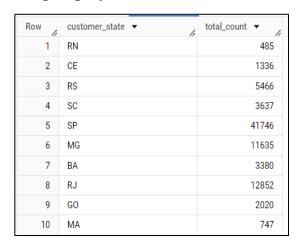
```
Code: - select customer_state,
    row_number() over (partition by customer_state order by month) as month,
    no_of_orders
    from (select customer_state,
    extract(month from order_purchase_timestamp) as month,
    count(order_id) as no_of_orders
    from `target.customers` C
    join `target.orders` O
    on C.customer_id = O.customer_id
    group by customer_state,month )
    order by 1;
```

Row	customer_state ▼	month ▼	no_of_orders ▼
1	AC	1	8
2	AC	2	6
3	AC	3	4
4	AC	4	9
5	AC	5	10
6	AC	6	7
7	AC	7	9
8	AC	8	7
9	AC	9	5
10	AC	10	6

<u>Insights</u>: 1- Can clearly observe month on month performance of each state which can help management to set targets and plan operations accordingly.

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

group by 1;



<u>Insights</u>: 1-SP has got the maximum number of customers.

- 2- RR has got the least number of customers.
- 3- we have to think new idea for RR to increase the no. of orders.

- IV. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.
- **A**. Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only)?

```
Code: - with year 17 as
       (select round(sum(p.payment value)) as cost of orders 2017
       from 'target.orders' O
      join 'target.payments' P
       on P.order id = O.order id
       where extract (year from O.order purchase timestamp)= 2017 and
      extract (month from O.order purchase timestamp) between 1 and 8),
      year18 as
      (select round(sum(P.payment value)) as cost of orders 2018
       from 'target.orders' O
      join 'target.payments' P
      on P.order id = O.order id
      where extract (year from O.order purchase timestamp)= 2018 and
      extract (month from O.order purchase timestamp) between 1 and 8)
      select cost of orders 2017,
      year18.cost of orders 2018,
```

concat(round(((year18.cost\_of\_orders\_2018-year17.cost\_of\_orders\_2017)/ year17.cost\_of\_orders\_2017)\*100),"%") as Percent\_increase from year17,year18;



<u>Insights</u>: 1- The cost of orders has increased from 2017 to 2018 by 137%.

2- This data is telling us to increase the inputs of the business (Goods, Manpower, etc) because this year we can expect more Orders.

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

Code: - select c.customer state,

round(sum(o2.price),2) as `total\_price`,
round(avg(o2.price),2) as `avg\_price`
from `target.customers` c join `target.orders` o
on c.customer\_id=o.customer\_id
join `target.order\_items` o2
on o.order\_id=o2.order\_id
group by c.customer\_state;

Row	customer_state ▼	total_price ▼	avg_price ▼
1	RN	83034.98	156.97
2	CE	227254.71	153.76
3	RS	750304.02	120.34
4	SC	520553.34	124.65
5	SP	5202955.05	109.65
6	MG	1585308.03	120.75
7	BA	511349.99	134.6
8	RJ	1824092.67	125.12
9	GO	294591.95	126.27
10	MA	119648.22	145.2

- <u>Insights</u>: 1-Staes with higher total prices and lesser average price shows large volume of smaller transactions.
  - 2-States with lower total prices and higher total price shows fewer volumes of larger transactions.
  - 3- States of SP has the highest total revenue followed by RJ.
  - 4- State of ac has the highest avg price but it has one of the

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

```
Code: - select c.customer state,
```

```
round(sum(o2.freight_value)) as `total_freight_value`,
round(avg(o2.freight_value) )as `avg_freight_value`
from `target.customers` c
join `target.orders` o
on c.customer_id = o.customer_id
join `target.order_items` o2
on o.order_id = o2.order_id
group by 1
order by 1;
```

Row	customer_state ▼	total_freight_value	avg_freight_value
1	MT	29715.43000000	28.16628436018
2	MA	31523.77000000	38.25700242718
3	AL	15914.58999999	35.84367117117
4	SP	718723.0699999	15.14727539041
5	MG	270853.4600000	20.63016680630
6	PE	59449.65999999	32.91786267995
7	RJ	305589.3100000	20.96092393168
8	DF	50625.499999999	21.04135494596
9	RS	135522.7400000	21.73580433039
10	SE	14111.46999999	36.65316883116

- <u>Insights</u>: 1- The state of SP has the highest total freight value and the lowest average freight value, indicating the cheapest and most efficient freight services.
  - 2- The state of AP has highest avg freight cost which shows

non efficient freight services.

3- States like PB, RN even after having lower revenue base has highest freight costs indicating inefficiency of logistics.

## V. Analysis based on sales, freight and delivery time.

A. 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?

Row	order_id ▼	time_to_deliver ▼	total_diff_estimated_
1	00010242fe8c5a6d1ba2dd792	7	8
2	00018f77f2f0320c557190d7a1	16	2
3	000229ec398224ef6ca0657da	7	13
4	00024acbcdf0a6daa1e931b03	6	5
5	00042b26cf59d7ce69dfabb4e	25	15
6	00048cc3ae777c65dbb7d2a06	6	14
7	00054e8431b9d7675808bcb8	8	16
8	000576fe39319847cbb9d288c	5	15
9	0005a1a1728c9d785b8e2b08	9	0
10	0005f50442cb953dcd1d21e1f	2	18

<u>Insights</u>: 1- The data shows the difference between actual delivery time versus estimated delivery time of an order.

2- Orders as seen in chart are mostly delivered

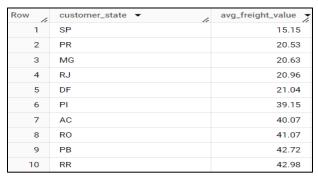
in the time frame of 30-40 days.

- 3- Another took 43 days to reach customer, which was 4 days longer than estimated delivery date.
- **B**. Find out the top 5 states with the highest & lowest average freight value.

```
Code: - with states freight value as
       (select c.customer state,
       round(avg(o2.freight value),2) as avg freight value,
       row number() over(order by round(avg(o2.freight value),2)asc)
       as 'rank of freight',
       row number() over(order by round(avg(o2.freight value),2)desc)
       as 'rank of freight2'
       from target.customers c
       join target.orders o
      on c.customer id=o.customer id
      join target.order items o2
      on o.order id=o2.order id
      group by c.customer state),
      table2 as
     (select customer state, avg freight value
     From states freight value
     where rank of freight <= 5
      union all
     select customer state, avg freight value
```

from states\_freight\_value where rank\_of\_freight2 <=5)

select \* from table2 order by avg\_freight\_value;



Insights: 1-RR,PB,RO,AV and PI are the top5 states with the highest avg freight Value.

2-SP,PR,MG,RJ and DF are the top5 states with the lowest avg freight Value.

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

```
Ans- with order details as
     (select customer state,
     round(avg(date diff(o.order delivered customer_date,o.order_purchase_tim
     estamp,day)),2) as avg delivery time,
     dense rank() over(order by
     avg(date diff(o.order delivered customer date,o.order purchase timestam
     p,day ))desc)
     as'del rank',
     dense rank() over(order by
     avg(date diff(o.order delivered customer date,o.order purchase timestam
     p,day ))asc)
     as 'del rank2'
     from target.customers c
    join target.orders o
     on c.customer id=o.customer id
     group by customer state),
     'table' as
     (select customer state, avg delivery time
     from order details
     where del rank2 between 1 and 5
     union all
```

select customer\_state,avg\_delivery\_time

from order details

where del\_rank between 1 and 5)

select \* from table order by avg\_delivery\_time asc;

Row	customer_state ▼	avg_delivery_time
1	SP	8.3
2	PR	11.53
3	MG	11.54
4	DF	12.51
5	SC	14.48
6	PA	23.32
7	AL	24.04
8	AM	25.99
9	AP	26.73
10	RR	28.98

<u>Insights</u>: 1-RR,AP,AM,AL and PA are the top5 states with highest avg deliver time.

2- SP,PR,MG,DF and SC are the top5 states with lowest avg deliver time.

**D**. Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery?

Row	customer_state ▼	avg_delivery_date 🎤
1	AL	7.95
2	MA	8.77
3	SE	9.17
4	ES	9.62
5	BA	9.93

Insights: 1-the data display the states where the order deliver is faster with reference to estimated time of delivery.

2- the data displays the states with mod=st efficient delivery time.

## VI. Analysis based on the payments:

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

```
Code: - select extract (year from order_purchase_timestamp) as

Order_Year,

extract(month from order_purchase_timestamp) as

Order_Month,

p.payment_type as Payment_method,

count(p.order_id) AS Number_of_Orders,

from `target.orders` o

join `target.payments` p

on o.order_id=p.order_id

group by Order_year,order_month,p.payment_type

order by Order year,order month
```

Row	Order_Year ▼	Order_Month ▼	Payment_method ▼	Number_of_Orders
1	2016	9	credit_card	3
2	2016	10	credit_card	254
3	2016	10	voucher	23
4	2016	10	debit_card	2
5	2016	10	UPI	63
6	2016	12	credit_card	1
7	2017	1	voucher	61
8	2017	1	UPI	197
9	2017	1	credit_card	583
10	2017	1	debit_card	9

Insights: 1-The data displays the various payment methods used by the customers for payment of the orders in each month.2-As observed, the most used methods include credit card, UPI,

Debit card and vouchers.

**B**. Find the no. of orders placed on the basis of the payment installments that have been paid?

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

<u>Insights</u>: 1- The data shows the payments made basis the number of installments paid till that time period.

2- More no. of orders are placed with a 1 instalment.

