

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

```
select * from `Ecommerce.orders`
```

2. Time period for which the data is given

By sorting data in ascending and then descending and subtracted the time value.

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

```
select cust.customer_city , cust.customer_state from  
`business-case-study-387111.customer_db.customers_table` as cust  
--10june 2018 to 17 oct 2018  
inner join `business-case-study-387111.orders.orders_table` as ord on  
cust.customer_id = ord.customer_id
```

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?

```
SELECT  
EXTRACT(YEAR FROM order_purchase_timestamp) AS year,  
EXTRACT(MONTH FROM order_purchase_timestamp) AS month,  
COUNT(*) AS count  
FROM  
`Ecommerce.orders`  
GROUP BY  
year,  
month  
ORDER BY  
year ASC,  
month ASC
```

2. What time do Brazilian customers tend to buy (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 'Morning'  
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 hour_of_day,
        count(*) as count,
    from `Ecommerce.orders`
    group by hour_of_day

```

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

1. Get month on month orders by states

```

select cust.customer_state as state , extract(month from
ord.order_purchase_timestamp) as month ,count(*) as count
from `Ecommerce.customers_table` as cust
inner join `Ecommerce.orders` as ord
on cust.customer_id = ord.customer_id
group by state , month
order by state , month

```

2. Distribution of customers across the states in Brazil

```

select customer_state, count(customer_state) as customer_count
from `Ecommerce.customers_table`

group by customer_state

```

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

5. Analysis on sales, freight and delivery time

1. Calculate days between purchasing, delivering and estimated delivery

```

SELECT
    order_purchase_timestamp,
    order_delivered_carrier_date,
    order_delivered_customer_date,
    order_estimated_delivery_date,
    DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) AS
days_to_deliver,
    DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, DAY) AS
days_to_estimated_delivery
FROM

```

```
`Ecommerce.orders`
```

2. Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:

- time_to_delivery =
order_delivered_customer_date-order_purchase_timestamp

```
SELECT
    order_purchase_timestamp,
    order_delivered_customer_date,
    order_estimated_delivery_date,
    TIMESTAMP_DIFF(order_delivered_customer_date,
    order_purchase_timestamp, DAY) AS time_to_delivery,
    TIMESTAMP_DIFF(order_estimated_delivery_date,
    order_delivered_customer_date, DAY) AS diff_estimated_delivery
FROM
    `Ecommerce.orders`
```
- diff_estimated_delivery =
order_estimated_delivery_date-order_delivered_customer_date

```
SELECT
    order_purchase_timestamp,
    order_delivered_customer_date,
    order_estimated_delivery_date,
    TIMESTAMP_DIFF(order_delivered_customer_date,
    order_purchase_timestamp, DAY) AS time_to_delivery,
    TIMESTAMP_DIFF(order_estimated_delivery_date,
    order_delivered_customer_date, DAY) AS diff_estimated_delivery
FROM
    `Ecommerce.orders`
```

3. Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

```
SELECT
    c.customer_state,
    AVG(oi.freight_value) AS avg_freight_value,
    AVG(TIMESTAMP_DIFF(o.order_delivered_customer_date,
    o.order_purchase_timestamp, DAY)) AS avg_time_to_delivery,
    AVG(TIMESTAMP_DIFF(o.order_estimated_delivery_date,
    o.order_delivered_customer_date, DAY)) AS avg_diff_estimated_delivery
FROM
    `Ecommerce.order_items` as oi
INNER JOIN
    `Ecommerce.orders` as o ON oi.order_id = o.order_id
```

```

INNER JOIN
  `Ecommerce.customers_table` as c ON o.customer_id = c.customer_id
GROUP BY
  c.customer_state;

```

4. Sort the data to get the following:
5. Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

```

select avg(orditem.freight_value) as avg_freight , cust.customer_state as
state from `Ecommerce.order_items` as orditem
inner join `Ecommerce.orders` as ord on ord.order_id = orditem.order_id
inner join `Ecommerce.customers_table` as cust on ord.customer_id =
cust.customer_id
group by cust.customer_state
order by avg_freight desc
limit 5

```

Part 2

```

select avg(orditem.freight_value) as avg_freight , cust.customer_state as
state from `Ecommerce.order_items` as orditem
inner join `Ecommerce.orders` as ord on ord.order_id = orditem.order_id
inner join `Ecommerce.customers_table` as cust on ord.customer_id =
cust.customer_id
group by cust.customer_state
order by avg_freight asc
limit 5

```

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

```

select avg(timestamp_diff(ord.order_purchase_timestamp
,ord.order_delivered_customer_date, day)) as avg_time_to_delivery ,
cust.customer_state as state
from `Ecommerce.order_items` as orditem
inner join `Ecommerce.orders` as ord on ord.order_id = orditem.order_id
inner join `Ecommerce.customers_table` as cust on ord.customer_id =
cust.customer_id
group by cust.customer_state
order by avg_time_to_delivery desc
limit 5

```

Part 2

```

select avg(timestamp_diff(ord.order_purchase_timestamp
,ord.order_delivered_customer_date, day)) as avg_time_to_delivery ,
cust.customer_state as state
from `Ecommerce.order_items` as orditem
inner join `Ecommerce.orders` as ord on ord.order_id = orditem.order_id

```

```
inner join `Ecommerce.customers_table` as cust on ord.customer_id =  
cust.customer_id  
group by cust.customer_state  
order by avg_time_to_delivery desc  
limit 5
```

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

6. Payment type analysis:

1. Month over Month count of orders for different payment types
2. Count of orders based on the no. of payment installments