1. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset
    1. Get number of rows in the data :

        - `select count(1) from `targetcasestudy1995.targetdb.customers' : 99441`
        - `select count(1) from `targetcasestudy1995.targetdb.geolocation` :1000163`
        - `select count(1) from `targetcasestudy1995.targetdb.order_items` :112650`
        - `select count(1) from `targetcasestudy1995.targetdb.order_reviews` :99224`
        - `select count(1) from `targetcasestudy1995.targetdb.orders``
        - `select count(1) from `targetcasestudy1995.targetdb.payments``
        - `select count(1) from `targetcasestudy1995.targetdb.products``
        - `select count(1) from `targetcasestudy1995.targetdb.sellers``

    2. Number of null or missing values in a column
    3. Data type of columns in a table
        1. `DESC `targetcasestudy1995.targetdb.customers``
        2. `DESC `targetcasestudy1995.targetdb.geolocation``
        3. `DESC `targetcasestudy1995.targetdb.order_items``
        4. `DESC `targetcasestudy1995.targetdb.order_reviews``
        5. `DESC `targetcasestudy1995.targetdb.orders``
        6. `DESC `targetcasestudy1995.targetdb.payments``
        7. `DESC `targetcasestudy1995.targetdb.products``
        8. `DESC `targetcasestudy1995.targetdb.sellers``

    4. Get the time period for which the data is given
        1. `select max(order_purchase_timestamp),min(order_purchase_timestamp) from `targetcasestudy1995.targetdb.orders` :`

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

    5. Number of cities in our dataset
        1. `Select (distinct geolocation_cities) from `targetcasestudy1995.targetdb.geolocation`: 8011`

    6. Number of states in our dataset
        1. `select count(distinct geolocation_state) from `targetcasestudy1995.targetdb.geolocation` : 27`

In-depth Exploration:

1. How many orders do we have for each order status?

    - `select order_status ,count(1) as Count from `targetcasestudy1995.targetdb.orders` group by order_status`

Row	order_status	Count
1	created	5
2	shipped	1107
3	approved	2
4	canceled	625
5	invoiced	314
6	delivered	96478
7	processing	301
8	unavailable	609

-

2. Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario?
    - `select EXTRACT(YEAR FROM order_purchase_timestamp) as Year, count(1) as COunt from `targetcasestudy1995.targetdb.orders` group by 1 order by Year`

Row	Year	COunt
1	2016	329
2	2017	45101
3	2018	54011

3. On what day of week brazilians customers tend to do online purchasing?
    - `select EXTRACT(DAYOFWEEK FROM order_purchase_timestamp) as DAY, count(1) as Count from `targetcasestudy1995.targetdb.orders` group by 1 order by Count`

Row	DAY	Count
1	7	10887
2	1	11960
3	6	14122
4	5	14761
5	4	15552
6	3	15963
7	2	16196

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

Row	Hour	Count
1	5	188
2	4	206
3	3	272
4	6	502
5	2	510
6	1	1170
7	7	1231
8	0	2394
9	8	2967
10	23	4123
11	9	4785
12	18	5769
13	22	5816
14	19	5982
15	12	5995
16	17	6150
17	10	6177
18	20	6193
19	21	6217
20	15	6454
21	13	6518
22	14	6569
23	11	6578
24	16	6675

```
select EXTRACT(hour FROM order_purchase_timestamp) as Hour, count(1) as Count fr
om `targetcasestudy1995.targetdb.orders` group by 1 order by Count
```

Brazilian like to shop in the Afternoon

5. Feature Extraction: Through order_purchase_timestamp in “orders” dataset extract

```
select  EXTRACT(Year FROM order_purchase_timestamp) as Year,
        EXTRACT(Month FROM order_purchase_timestamp) as Month,
        EXTRACT(Day FROM order_purchase_timestamp) as Day,
        EXTRACT(date FROM order_purchase_timestamp) as date,
        EXTRACT(Dayofweek FROM order_purchase_timestamp) as dayofweek,
        CASE
            when EXTRACT(Dayofweek FROM order_purchase_timestamp) = 1
            then "Sunday"
            when EXTRACT(Dayofweek FROM order_purchase_timestamp) = 2
            then "Monday"
            when EXTRACT(Dayofweek FROM order_purchase_timestamp) = 3
            then "Tuesday"
            when EXTRACT(Dayofweek FROM order_purchase_timestamp) = 4
            then "Wednesday"
            when EXTRACT(Dayofweek FROM order_purchase_timestamp) = 5
            then "Thursday"
            when EXTRACT(Dayofweek FROM order_purchase_timestamp) = 6
            then "Friday"
            when EXTRACT(Dayofweek FROM order_purchase_timestamp) = 7
            then "Saturday"
          end as day_of_week_name,
        EXTRACT(time FROM order_purchase_timestamp) as time,
         count(1) as Count from `targetcasestudy1995.targetdb.orders` group by 1,2,3,4,5,6,7 o
rder by Count
```

2. Evolution of E-commerce orders in the Brazil region:
    1. Get month on month orders by region

```
select Extract(Month From o.order_purchase_timestamp) as Month,
       c.customer_state,
       Count(1) as Count_Of_Orders
       from `targetcasestudy1995.targetdb.orders` as o
       join `targetcasestudy1995.targetdb.customers` as c
       on o.customer_id = c.customer_id
       group by 1,2
```

2. Total of customer orders by state

```
select
        c.customer_state,
        Count(1) as Count_Of_Orders
        from `targetcasestudy1995.targetdb.orders` as o
        join `targetcasestudy1995.targetdb.customers` as c
        on o.customer_id = c.customer_id
        group by 1
```

Highest Orders are from SP state

3. Top 10 brazilian cities most no. of orders

```
select
        c.customer_city,
        Count(1) as Count_Of_Orders
        from `targetcasestudy1995.targetdb.orders` as o
        join `targetcasestudy1995.targetdb.customers` as c
        on o.customer_id = c.customer_id
        group by 1 order by 2 DESC limit 10
```

Row	customer_city	Count_Of_Orders
1	sao paulo	15540
2	rio de janeiro	6882
3	belo horizonte	2773
4	brasilia	2131
5	curitiba	1521
6	campinas	1444
7	porto alegre	1379
8	salvador	1245
9	guarulhos	1189
10	sao bernardo do campo	938

4. How are customers distributed in Brazil

```
select
        c.customer_state,
        Count(1) as Count_Of_Customers
         from `targetcasestudy1995.targetdb.customers` as c
        group by 1 order by 2 DESC
```

Row	customer_state	Count_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
16	MS	715
17	PB	536
18	PI	495
19	RN	485
20	AL	413
21	SE	350
22	TO	280
23	RO	253
24	AM	148
25	AC	81
26	AP	68
27	RR	46

5. City wise number of unique customers

```
select
        c.customer_city,
        Count(distinct customer_id) as Count_Of_Customers
        from `targetcasestudy1995.targetdb.customers` as c
        group by 1 order by 2 DESC
```

Row	customer_city	Count_Of_Customers
1	sao paulo	15540
2	rio de janeiro	6882
3	belo horizonte	2773
4	brasilia	2131
5	curitiba	1521

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

Answer the following questions:

1. Total amount sold in 2017 between Jan to August

```
WITH order_with_items AS (
SELECT EXTRACT(YEAR FROM o.order_purchase_timestamp) as Year,
  EXTRACT(MONTH FROM o.order_purchase_timestamp) as Month,
  COUNT(o.order_id) Number_of_orders,
  SUM(oi.price) as Total_price,
  SUM(oi.freight_value) as total_freight,
  SUM(oi.price)/COUNT(o.order_id) as  price_per_order,
  SUM(oi.freight_value)/COUNT(o.order_id) as freight_per_order
  from `targetcasestudy1995.targetdb.orders` o JOIN
  `targetcasestudy1995.targetdb.order_items` oi ON
  oi.order_id = o.order_id
  GROUP BY 1,2 ORDER BY 1,2
)

SELECT SUM(Total_price) as Tot
from order_with_items
where Year = 2017 and
Month between 1 and 8
```

Row	Tot
1	3113000.3200000809

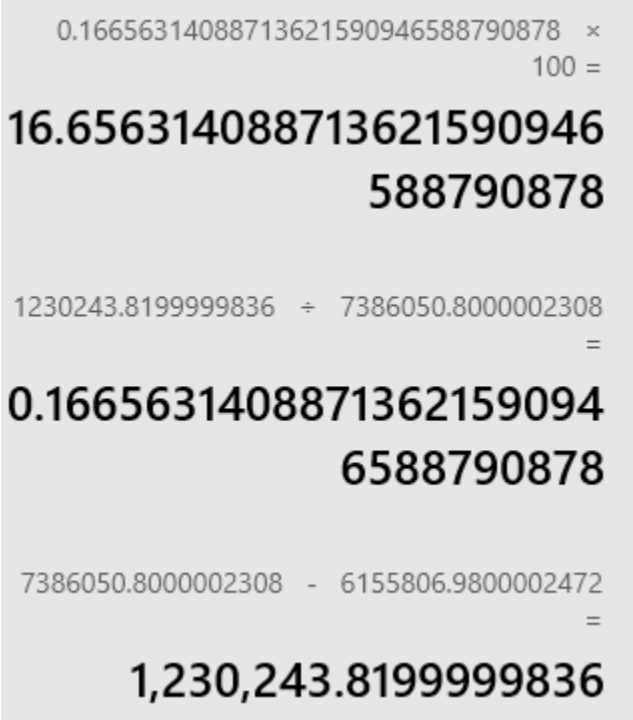
2. Total amount sold in 2018 between Jan to august

```
WITH order_with_items AS (
SELECT EXTRACT(YEAR FROM o.order_purchase_timestamp) as Year,
  EXTRACT(MONTH FROM o.order_purchase_timestamp) as Month,
  COUNT(o.order_id) Number_of_orders,
  SUM(oi.price) as Total_price,
  SUM(oi.freight_value) as total_freight,
  SUM(oi.price)/COUNT(o.order_id) as  price_per_order,
  SUM(oi.freight_value)/COUNT(o.order_id) as freight_per_order
  from `targetcasestudy1995.targetdb.orders` o JOIN
  `targetcasestudy1995.targetdb.order_items` oi ON
  oi.order_id = o.order_id
  GROUP BY 1,2 ORDER BY 1,2
)

SELECT SUM(Total_price) as Tot
from order_with_items
where Year = 2018 and
Month between 1 and 8
```

Row	Tot
1	7385905.8000002308

3. % increase from 2017 to 2018: **16.65%**



Step 3: Join (orders+order_items) table from previous step with “customers” table on Customer_id and find:

1. Mean & Sum of price by customer state

   Mean:

```
WITH order_with_items AS (
SELECT
  c.customer_state as state,
  SUM(oi.price) as Total_price,
  SUM(oi.freight_value) as total_freight
  from `targetcasestudy1995.targetdb.orders` o JOIN
  `targetcasestudy1995.targetdb.order_items` oi ON
  oi.order_id = o.order_id
  JOIN `targetcasestudy1995.targetdb.customers` as c ON
  o.customer_id = c.customer_id
  GROUP BY 1 ORDER BY 1
)

SELECT AVG(Total_price) as Mean
from order_with_items
```

Row	Mean
1	503394.21111119242

SUM:

```
1  WITH order_with_items AS (
2  SELECT
3    c.customer_state as state,
4    SUM(oi.price) as Total_price,
5    SUM(oi.freight_value) as total_freight
6    from `targetcasestudy1995.targetdb.orders` o JOIN
7    `targetcasestudy1995.targetdb.order_items` oi ON
8    oi.order_id = o.order_id
9    JOIN `targetcasestudy1995.targetdb.customers` as c ON
0    o.customer_id = c.customer_id
1    GROUP BY 1 ORDER BY 1
2  )
3
4  SELECT SUM(Total_price) as Tot
5  from order_with_items
6
7
```

Row	Tot
1	13591643.700001186

2. Mean & Sum of freight value by customer state

MEAN:

```
WITH order_with_items AS (
SELECT
  c.customer_state as state,
  SUM(oi.price) as Total_price,
  SUM(oi.freight_value) as total_freight
  from `targetcasestudy1995.targetdb.orders` o JOIN
  `targetcasestudy1995.targetdb.order_items` oi ON
  oi.order_id = o.order_id
  JOIN `targetcasestudy1995.targetdb.customers` as c ON
  o.customer_id = c.customer_id
  GROUP BY 1 ORDER BY 1
)

SELECT AVG(order_with_items.total_freight) as Mean
from order_with_items
```

Row	Mean
1	83404.057037036662

SUM:

```
WITH order_with_items AS (
SELECT
  c.customer_state as state,
  SUM(oi.price) as Total_price,
  SUM(oi.freight_value) as total_freight
  from `targetcasestudy1995.targetdb.orders` o JOIN
  `targetcasestudy1995.targetdb.order_items` oi ON
  oi.order_id = o.order_id
  JOIN `targetcasestudy1995.targetdb.customers` as c ON
  o.customer_id = c.customer_id
  GROUP BY 1 ORDER BY 1
)

SELECT SUM(order_with_items.total_freight) as SUM
from order_with_items
```

Row	SUM
1	2251909.5399999893

5. Analysis on sales, freight and delivery time

1. Calculating days between purchasing, delivering and estimated delivery :

```
SELECT  DATETIME_DIFF( order_delivered_customer_date, order_purchase_timestamp, DAY) AS delivery_daydelta,
        DATEtime_diff( order_estimated_delivery_date, order_purchase_timestamp, DAY) AS estimate_daydelta,
        order_id from `targetcasestudy1995.targetdb.orders`
        where order_delivered_customer_date is not null and
        order_estimated_delivery_date is not null and
        order_purchase_timestamp is not null
```

Row	delivery_daydelta	estimate_daydelta	order_id
1	30	17	1950d777989f6a877539f53795b4c3c3
2	30	59	2c45c33d2f9cb8ff8b1c86cc28c11c30
3	35	52	65d1e226dfaeb8cdc42f665422522d14
4	30	32	635c894d068ac37e6e03dc54eccb6189
5	32	33	3b97562c3aee8bdedcb5c2e45a50d5e1

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

```
SELECT  c.customer_state,
        avg(oi.freight_value) Mean_freight,
        avg(datetime_diff( order_delivered_customer_date, order_purchase_timestamp,day)) AS mean_time_to_delivery,
        avg(datetime_diff( order_estimated_delivery_date, order_purchase_timestamp,day)) AS mean_diff_estimated_delivery
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        Group by 1
```

4. Sort the data to get the following:

a. Top 5 states with highest/lowest average freight value

```
SELECT  c.customer_state,
        avg(oi.freight_value) Mean_freight,
        avg(datetime_diff( order_delivered_customer_date, order_purchase_timestamp,day)) AS mean_time_to_delivery,
        avg(datetime_diff( order_estimated_delivery_date, order_purchase_timestamp,day)) AS mean_diff_estimated_delivery
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        Group by 1 order by 1 desc limit 5
```

Row	customer_state	Mean_freight	mean_time_to_delivery	mean_diff_estimated_delivery
1	TO	37.246603174603187	17.003225806451624	28.803174603174618
2	SP	15.147275390419248	8.25960855241909	18.898290796434175
3	SE	36.653168831168855	20.978666666666651	30.353246753246747
4	SC	21.470368773946436	14.520985846754517	25.50598659003828
5	RS	21.735804330392945	14.708299364095817	28.309061748195688

```
SELECT  c.customer_state,
        avg(oi.freight_value) Mean_freight,
        avg(datetime_diff( order_delivered_customer_date, order_purchase_timestamp,day)) AS mean_time_to_delivery,
        avg(datetime_diff( order_estimated_delivery_date, order_purchase_timestamp,day)) AS mean_diff_estimated_delivery
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        Group by 1 order by 1 limit 5
```

Row	customer_state	Mean_freight	mean_time_to_delivery	mean_diff_estimated_delivery
1	AC	40.073369565217405	20.329670329670336	40.695652173913032
2	AL	35.843671171171152	23.992974238875881	32.175675675675642
3	AM	33.205393939393936	25.963190184049076	45.2060606060606
4	AP	34.006097560975618	27.753086419753075	45.4878048780488
5	BA	26.363958936562248	18.774640238935675	29.135035535667274

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

```
SELECT  c.customer_state,
        avg(oi.freight_value) Mean_freight,
        avg(datetime_diff( order_delivered_customer_date, order_purchase_timestamp,day)) AS mean_time_to_delivery,
        avg(datetime_diff( order_estimated_delivery_date, order_purchase_timestamp,day)) AS mean_diff_estimated_delivery
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        Group by 1 order by 2 limit 5
```

Row	customer_state	Mean_freight	mean_time_to_delivery	mean_diff_estimated_delivery
1	SP	15.147275390419248	8.25960855241909	18.898290796434175
2	PR	20.531651567944248	11.480793060718735	24.37578397212544
3	MG	20.630166806306541	11.515522180072811	24.308401249143145
4	RJ	20.96092393168248	14.689382157500321	26.095068248851021
5	DF	21.041354945968383	12.501486199575384	24.192851205320071

```
SELECT  c.customer_state,
        avg(oi.freight_value) Mean_freight,
        avg(datetime_diff( order_delivered_customer_date, order_purchase_timestamp,day)) AS mean_time_to_delivery,
        avg(datetime_diff( order_estimated_delivery_date, order_purchase_timestamp,day)) AS mean_diff_estimated_delivery
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        Group by 1 order by 2 desc limit 5
```

Row	customer_state	Mean_freight	mean_time_to_delivery	mean_diff_estimated_delivery
1	RR	42.984423076923093	27.826086956521738	45.980769230769219
2	PB	42.723803986710941	20.119453924914676	32.548172757475093
3	RO	41.069712230215842	19.282051282051292	38.651079136690655
4	AC	40.073369565217405	20.329670329670336	40.695652173913032
5	PI	39.147970479704767	18.931166347992352	29.922509225092242

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

```
SELECT  c.customer_state,
        avg(oi.freight_value) Mean_freight,
        avg(datetime_diff( order_delivered_customer_date, order_purchase_timestamp,day)) AS mean_time_to_delivery,
        avg(datetime_diff( order_estimated_delivery_date, order_purchase_timestamp,day)) AS mean_diff_estimated_delivery
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        Group by 1 order by 3 limit 5
```

Row	customer_state	Mean_freight	mean_time_to_delivery	mean_diff_estimated_delivery
1	SP	15.147275390419248	8.25960855241909	18.898290796434175
2	PR	20.531651567944248	11.480793060718735	24.37578397212544
3	MG	20.630166806306541	11.515522180072811	24.308401249143145
4	DF	21.041354945968383	12.501486199575384	24.192851205320071
5	SC	21.470368773946436	14.520985846754517	25.50598659003828

```
SELECT  c.customer_state,
        avg(oi.freight_value) Mean_freight,
        avg(datetime_diff( order_delivered_customer_date, order_purchase_timestamp,day)) AS mean_time_to_delivery,
        avg(datetime_diff( order_estimated_delivery_date, order_purchase_timestamp,day)) AS mean_diff_estimated_delivery
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        Group by 1 order by 3 desc limit 5
```

Row	customer_state	Mean_freight	mean_time_to_delivery	mean_diff_estimated_delivery
1	SP	15.147275390419248	8.25960855241909	18.898290796434175
2	PR	20.531651567944248	11.480793060718735	24.37578397212544
3	MG	20.630166806306541	11.515522180072811	24.308401249143145
4	DF	21.041354945968383	12.501486199575384	24.192851205320071
5	SC	21.470368773946436	14.520985846754517	25.50598659003828

6. Payment type analysis: Join “payments” dataset with the existing data on order_id

a. Count of orders for different payment types

```
SELECT  p.payment_type,
        count(o.order_id) count_of_payments
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        join `targetcasestudy1995.targetdb.payments` p on
        p.order_id = o.order_id
        Group by 1 order by 2
```

Row	payment_type	count_of_payments
1	debit_card	1691
2	voucher	6274
3	UPI	22867
4	credit_card	86769

b. Distribution of payment installments and count of orders

```
SELECT  p.payment_installments,
        count(o.order_id) count_of_payments
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        join `targetcasestudy1995.targetdb.payments` p on
        p.order_id = o.order_id
        Group by 1 order by 2 desc
```

Row	payment_installments	count_of_payments
1	1	58617
2	2	13722
3	3	11756
4	4	7979
5	10	6845
6	5	6017

c. Count of orders for different payment types Month over Month

```
SELECT  p.payment_installments,
        extract(month from o.order_purchase_timestamp) month,
        count(distinct o.order_id) count_of_payments
        from `targetcasestudy1995.targetdb.orders` o join
        `targetcasestudy1995.targetdb.customers` c on
        o.customer_id = c.customer_id
        join `targetcasestudy1995.targetdb.order_items` oi on
        o.order_id = oi.order_id
        join `targetcasestudy1995.targetdb.payments` p on
        p.order_id = o.order_id
        Group by 1,2 order by 3 desc
```

Row	payment_installments	month	count_of_payments
1	1	8	5235
2	1	5	4977
3	1	7	4970
4	1	3	4891
5	1	4	4675
6	1	6	4495
7	1	2	4441
8	1	1	4230
9	1	11	3571
10	1	12	2817

Row	payment_installments	month	count_of_payments
11	1	10	2311
12	1	9	1996
13	2	5	1318
14	2	8	129

6. Actionable Insights:
    - Delivery time and estimated time are high in many case.
    - Delivery and estimates are not near to each other for orders
    - Small cities do not have much costumers and orders as well.
    - Night time has few purchases
7. Recommendations
    - Delivery time can be improved and 1 day and 2 day delivery should be introduced
    - Estimates and actual delivery should be near, company should work on their estimation software
    - Company should focus on Smaller Cities as well.
    - Offers should be given at the night time so that people visit the website and purchase at night as well