Project:
Business CASE: Target SQL

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

# 1.1 Data type of columns in a table?

# For customer Table :
SELECT * EXCEPT(is_generated, generation_expression, is_stored, is_updatable)
FROM `driven-elixir-385510.project_data_set.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'customers';

# For geolacation Table:
SELECT * EXCEPT(is_generated, generation_expression, is_stored, is_updatable)
FROM `driven-elixir-385510.project_data_set.INFORMATION_SCHEMA.COLUMNS`

# For order_items Table:
SELECT * EXCEPT(is_generated, generation_expression, is_stored, is_updatable)
FROM `driven-elixir-385510.project_data_set.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'order_items';

# For orders reviews Table:
SELECT * EXCEPT(is_generated, generation_expression, is_stored, is_updatable)
FROM `driven-elixir-385510.project_data_set.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'order_reviews';

# For orders Table:
SELECT * EXCEPT(is_generated, generation_expression, is_stored, is_updatable)
FROM `driven-elixir-385510.project_data_set.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'orders';

#For payments Table:
SELECT * EXCEPT(is_generated, generation_expression, is_stored, is_updatable)
FROM `driven-elixir-385510.project_data_set.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'payments';

#For products Table:
SELECT * EXCEPT(is_generated, generation_expression, is_stored, is_updatable)
FROM `driven-elixir-385510.project_data_set.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'products';

#For sellers Table:
SELECT * EXCEPT(is_generated, generation_expression, is_stored, is_updatable)
FROM `driven-elixir-385510.project_data_set.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'sellers';

# 1.2  Time period for which the data is given?
SELECT Min(order_purchase_timestamp) AS start_date ,Max(order_purchase_timestamp)AS
END_date
FROM `project_data_set.orders`

# 1.3 Cities AND States of customers ordered during the given period?
SELECT c.customer_city,c.customer_state
FROM `project_data_set.customers` c JOIN `project_data_set.orders` o  ON c.customer_id =
o.customer_id
WHERE o.order_purchase_timestamp BETWEEN "2016-09-04 21:15:19" AND "2018-10-17 17:30:18"
```

Project:
Business CASE: Target SQL

```
# 2. In-depth Exploration:

# 2.1 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(order_id) AS order_id_COUNT
FROM `project_data_set.orders`
GROUP BY 1,2
ORDER BY 3 DESC


#2.2 What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?
SELECT CASE
WHEN EXTRACT(hour FROM order_purchase_timestamp) BETWEEN 1 AND 7 THEN "Dawn"
WHEN EXTRACT(hour FROM order_purchase_timestamp) BETWEEN 8 AND 12 THEN "Morning"
WHEN EXTRACT(hour FROM order_purchase_timestamp) BETWEEN 13  AND 20 THEN "AfternoON"
ELSE "Night" END
AS day_slots,COUNT(distinct order_id)
FROM `project_data_set.orders`
GROUP BY 1

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

#3.1 Get month on month orders by states?
SELECT EXTRACT(month FROM o.order_purchase_timestamp) AS month,c.customer_state
,COUNT(o.order_id) AS COUNT_orders
FROM `project_data_set.orders`  o JOIN `project_data_set.customers` c ON c.customer_id =
o.customer_id
GROUP BY 1,2
ORDER BY 1

#3.2 Distribution of customers across the states in Brazil?
SELECT customer_state,COUNT(customer_id) AS customer_Count
FROM `project_data_set.customers`
GROUP BY 1

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

#4.1 Get % increase in cost of orders form 2017 to 2018 (include months between Jan to Aug
only) - You can use “payment_value” column in payments table?
 SELECT ROUND((SUM(CASE WHEN year = 2018 THEN Total END)-SUM(CASE WHEN year = 2017 THEN
Total END))/SUM(CASE WHEN year = 2017 THEN Total END)*100) AS per_incr
 FROM
       (SELECT  EXTRACT(year FROM o.order_purchase_timestamp) AS year,EXTRACT(month FROM
       o.order_purchase_timestamp) AS month,
       SUM(p.payment_value) AS Total
       FROM `project_data_set.orders`  o JOIN `project_data_set.payments` p ON o.order_id
       = p.order_id
       WHERE EXTRACT(year FROM o.order_purchase_timestamp)  BETWEEN 2017 AND  2018 AND
       EXTRACT(month FROM o.order_purchase_timestamp) BETWEEN 1 AND 7
       GROUP BY 1,2) A
```

```
#4.2 Mean & SUM of price and freight value by customer state?
SELECT c.customer_state,AVG(oi.price) AS mean_price, SUM(oi.price) AS
total_price,SUM(oi.freight_value) AS total_freight_val
FROM `project_data_set.customers` c JOIN `project_data_set.orders`o ON c.customer_id =
o.customer_id
JOIN `project_data_set.order_items` oi  ON o.order_id = oi.order_id
GROUP BY 1

# 5. Analysis on sales, freight and delivery time

#5.1 Calculate days BETWEEN purchase, delivering AND estimated delivery?
SELECT DATE_DIFF(order_purchase_timestamp,order_delivered_customer_date,day) AS delivering
,DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date,day) AS
estimated_delivery
FROM `project_data_set.orders`

#5.2 Find time_to_delivery & diff_estimated_delivery?
SELECT order_id,DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,day) AS
time_to_delivery,order_id,DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_
date,day) AS diff_estimated_delivery FROM `project_data_set.orders`


#5.3 Group data by state, take mean of freight_value, time_to_delivery,
diff_estimated_delivery?

SELECT c.customer_state,AVG(oi.freight_value) AS
mean_freight_val,AVG(DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,d
ay)) AS
mean_time_to_delivery,AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_delivered_custo
mer_date,day)) AS mean_diff_estimated_delivery
FROM `project_data_set.customers` c JOIN `project_data_set.orders` o ON c.customer_id =
o.customer_id
JOIN `project_data_set.order_items` oi  ON o.order_id = oi.order_id
GROUP BY 1

# 5.4 Sort the data to get the following?
SELECT c.customer_state,AVG(oi.freight_value) AS
mean_freight_val,AVG(DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,d
ay)) AS
mean_time_to_delivery,AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_delivered_custo
mer_date,day)) AS mean_diff_estimated_delivery
FROM `project_data_set.customers` c JOIN `project_data_set.orders` o ON c.customer_id =
o.customer_id
JOIN `project_data_set.order_items` oi  ON o.order_id = oi.order_id
GROUP BY 1
ORDER BY 1,2,3,4

#5.5.1 Top 5 states with highest/lowest average freight value - sort in DESC/ASC LIMIT 5?
SELECT c.customer_state,AVG(oi.freight_value) AS
AVG_freight_val,AVG(DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,da
y)) AS
mean_time_to_delivery,AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_delivered_custo
mer_date,day)) AS mean_diff_estimated_delivery
FROM `project_data_set.customers` c JOIN `project_data_set.orders` o ON c.customer_id =
o.customer_id
JOIN `project_data_set.order_items` oi  ON o.order_id = oi.order_id
GROUP BY 1 ORDER BY AVG_freight_val DESC LIMIT 5
```

```
#5.5.2  Top 5 lowest average freight value
SELECT c.customer_state,AVG(oi.freight_value) AS
AVG_freight_val,AVG(DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,da
y)) AS
mean_time_to_delivery,AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_delivered_custo
mer_date,day)) AS mean_diff_estimated_delivery
FROM `project_data_set.customers` c JOIN `project_data_set.orders` o ON c.customer_id =
o.customer_id
JOIN `project_data_set.order_items` oi  ON o.order_id = oi.order_id
GROUP BY 1
ORDER BY AVG_freight_val ASC
LIMIT 5

#5.6.1Top 5 states with highest/lowest average time to delivery?
#lowest average time to delivery
SELECT c.customer_state,AVG(oi.freight_value) AS
AVG_freight_val,AVG(DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,da
y)) AS
mean_time_to_delivery,AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_delivered_custo
mer_date,day)) AS diff_estimated_delivery
FROM `project_data_set.customers` c JOIN `project_data_set.orders` o ON c.customer_id =
o.customer_id
JOIN `project_data_set.order_items` oi  ON o.order_id = oi.order_id
GROUP BY 1
ORDER BY time_to_delivery ASC
LIMIT 5

#5.6.2 Highest average time to delivery
SELECT c.customer_state,AVG(oi.freight_value) AS
AVG_freight_val,AVG(DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,da
y)) AS
mean_time_to_delivery,AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_delivered_custo
mer_date,day)) AS diff_estimated_delivery
FROM `project_data_set.customers` c JOIN `project_data_set.orders` o ON c.customer_id =
o.customer_id
JOIN `project_data_set.order_items` oi  ON o.order_id = oi.order_id
GROUP BY 1
ORDER BY time_to_delivery DESC
LIMIT 5

#5.7 Top 5 states where delivery is really fast/ not so fast compared to estimated date ?
SELECT c.customer_state,(oi.freight_value) AS
AVG_freight_val,DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,day)
AS time_to_delivery,
DATE_DIFF(o.order_estimated_delivery_date,o.order_delivered_customer_date,day) AS
estimated_delivery
FROM `project_data_set.customers` c JOIN `project_data_set.orders` o ON c.customer_id =
o.customer_id
JOIN `project_data_set.order_items` oi  ON o.order_id = oi.order_id
ORDER BY time_to_delivery DESC
LIMIT 5
```

Project:
Business CASE: Target SQL

```
# 6. Payment type analysis:

# 6.1 Month over Month COUNT of orders for different payment types?
 SELECT p.payment_type, EXTRACT(year  FROM o.order_purchase_timestamp) AS year
,EXTRACT(month FROM o.order_purchase_timestamp) AS month,COUNT(o.order_id)  AS COUNT_orders
FROM `project_data_set.orders` o JOIN `project_data_set.payments` p ON o.order_id =
p.order_id
GROUP BY 1,2,3
ORDER BY 3

#6.2 COUNT of orders based on the no. of payment instalments?
 SELECT p.payment_installments, COUNT(o.order_id)  AS COUNT_orders
FROM `project_data_set.orders` o JOIN `project_data_set.payments` p ON o.order_id =
p.order_id
GROUP BY 1
ORDER BY 1
```