SQL E-Commerce Study:

1. 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.

Field name	Type	Mode	Key	Collation	Default Value
customer_id	STRING	NULLABLE			
customer_unique_id	STRING	NULLABLE			
customer_zip_code_prefix	INTEGER	NULLABLE			
customer_city	STRING	NULLABLE			
customer_state	STRING	NULLABLE			

Here we can see two columns customer_id and customer_unique_id. The customer_id column has duplicate values but no duplicate values are there in customer_unique_id

b) Get the time range between which the orders were placed.

```
SELECT min(order_purchase_timestamp) as min_,
       max(order_purchase_timestamp) as max
FROM `Business_case.orders` ;
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

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

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

```
SELECT COUNT(DISTINCT(c.customer_city)) as
City_count,
       COUNT(DISTINCT(c.customer_state)) as
State_count
FROM `Business_case.customers` c
JOIN `Business_case.orders` o
ON c.customer_id = o.customer_id ;
```

Query results

JOB INFORMATION | RESULTS | JSON

Row	City_count	State_count
1	4119	27

2. In-Depth Exploration:
   a) Is there a growing trend in the no. of orders placed over the past years?

```
SELECT EXTRACT(year from order_purchase_timestamp) as year_,
       EXTRACT(month from order_purchase_timestamp) as month_,
       COUNT(order_id) as orders
FROM `Business_case.orders`
GROUP BY 1,2
ORDER BY 1,2;
```

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	year_	month_	orders
1	2016	9	4
2	2016	10	324
3	2016	12	1
4	2017	1	800
5	2017	2	1780
6	2017	3	2682

From the results it can be inferred that, the number of orders placed in 2016 were substantial but in 2017 the rate of order placing has started growing drastically.
In November of 2017 number of orders placed were the highest in the entire year due to various events like Black Friday, Thanksgiving etc.
In 2018 except last 2-3 months, other months are showing high number of orders placed but not the high rate of order placing as compared to 2017 because of high Base Effect.

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

```
SELECT EXTRACT(MONTH FROM order_purchase_timestamp) as
month_,
       COUNT(DISTINCT(order_id)) as no_of_orders
FROM `Business_case.orders`
GROUP BY month_
ORDER BY 1;
```

Query results

JOB INFORMATION | RESULTS | JSON

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

From the query results it is evident that count of orders is on higher side for first Eight months than that of rest Four months and it can be also confirmed that there is monthly seasonality in the orders. The number of orders has been increasing from March to August though in between there is some fluctuations. Here the August is peak month to put number of orders.

Increasing the granularity deliberately to get more insights :

```
SELECT *,
       LAG(quantity,1) OVER(PARTITION BY year_ ORDER
BY    month_) as prev_month_quantity
 FROM
 (
 SELECT year_,
      month_,
      COUNT(*) as quantity
 FROM
 (
```

```
SELECT EXTRACT(YEAR FROM order_purchase_timestamp) as
year_,
       EXTRACT(MONTH FROM order_purchase_timestamp) as
month_
 FROM `Business_case.orders`
 ) s
 GROUP BY 1,2
 ) s1
 ORDER BY year_, month_;
```

Query results

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

Row	year_	month_	quantity	prev_month_quantity
1	2016	9	4	*null*
2	2016	10	324	4
3	2016	12	1	324
4	2017	1	800	*null*
5	2017	2	1780	800

From the outcome of the query it can be inferred that there is a significant seasonality at the end of each year 2016 ,2017 and 2018 (especially in the last three months).
One can find moderate seasonality in 2nd and 3rd quarter of 2017 but it started picking up from 4th quarter.
Whereas from 2nd quarter of 2018 the quantity ordered is showing decreasing trend.

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

```
SELECT time_category,
       COUNT(*) as order_placed
FROM
(
```

```
        SELECT *,
         CASE
         WHEN time_ BETWEEN 0 AND 6 THEN 'dawn'
         WHEN time_ BETWEEN 7 AND 12 THEN 'mornings'
         WHEN time_ BETWEEN 13 AND 18 THEN 'afternoon'
         ELSE 'night'
        END as time_category
    FROM
    (
      SELECT *,
       EXTRACT(HOUR FROM order_purchase_timestamp) as
time_
    FROM
    (
     SELECT order_purchase_timestamp ,
       time(order_purchase_timestamp) as time_1
     FROM `Business_case.orders`
     ) t
     ) t1
     ) tc
   GROUP BY time_category
   ORDER BY 2;
```

JOB INFORMATION | RESULTS | JSON | EXE

Row	time_category	order_placed
1	dawn	5242
2	mornings	27733
3	night	28331
4	afternoon	38135

3. Evolution of E-commerce orders in the Brazil region:
   a) Get the month on month no. of orders placed in each state.

```
SELECT customer_state,
       month_,
```

```
        COUNT(*) as order_count,
    FROM
    (
      SELECT c.customer_id,
        c.customer_state,
        o.order_purchase_timestamp,
        EXTRACT(YEAR FROM o.order_purchase_timestamp) as
    year_,
        EXTRACT(MONTH FROM o.order_purchase_timestamp) as
month_
     FROM `Business_case.customers` c
     LEFT OUTER JOIN `Business_case.orders` o
     ON c.customer_id = o.customer_id
     WHERE o.customer_id IS NOT NULL
     ) t
     GROUP BY 1,2
     ORDER BY 1,2
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	customer_state	month_	order_count
1	AC	1	8
2	AC	2	6
3	AC	3	4
4	AC	4	9
5	AC	5	10
6	AC	6	7

It is evident that SP consistently has the highest number of orders in any given month, followed by RJ and MG.

b) How are the customers distributed across all the states?

```
SELECT customer_state,
       COUNT(DISTINCT(customer_id)) as
customer_distribution
FROM `Business_case.customers`
GROUP BY customer_state
ORDER BY 1,2 ;
```

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

Row	customer_state	customer_distributio
1	AC	81
2	AL	413
3	AM	148
4	AP	68
5	BA	3380
6	CE	1336

Results per page: 50 1 – 27 of 27

Lowest no. of customers are from RR(46) state while highest number of customers are from SP(41746)

4. 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).

```
SELECT
  EXTRACT(MONTH FROM o.order_purchase_timestamp) AS
month,
  (
    (
      SUM   (
           CASE
             WHEN EXTRACT(YEAR FROM
o.order_purchase_timestamp) = 2018 AND
             EXTRACT(MONTH FROM
o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
p.payment_value
            END
            )
      -
      SUM   (
            CASE WHEN EXTRACT(YEAR FROM
o.order_purchase_timestamp) = 2017 AND
                 EXTRACT(MONTH FROM
o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
p.payment_value
            END
            )
    )
    /
          SUM   (
            CASE WHEN EXTRACT(YEAR FROM
o.order_purchase_timestamp) = 2017 AND
                 EXTRACT(MONTH FROM
o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
p.payment_value
            END
            )
   )*100 AS percent_increase
FROM
  `Business_case.orders` o
JOIN
  `Business_case.payments` p ON o.order_id =
p.order_id
```

```
WHERE
  EXTRACT(YEAR FROM o.order_purchase_timestamp) IN
(2017, 2018) AND
  EXTRACT(MONTH FROM o.order_purchase_timestamp)
BETWEEN 1 AND 8
GROUP BY 1
ORDER BY 1;
```

Query results

JOB INFORMATION | RESULTS | JSON

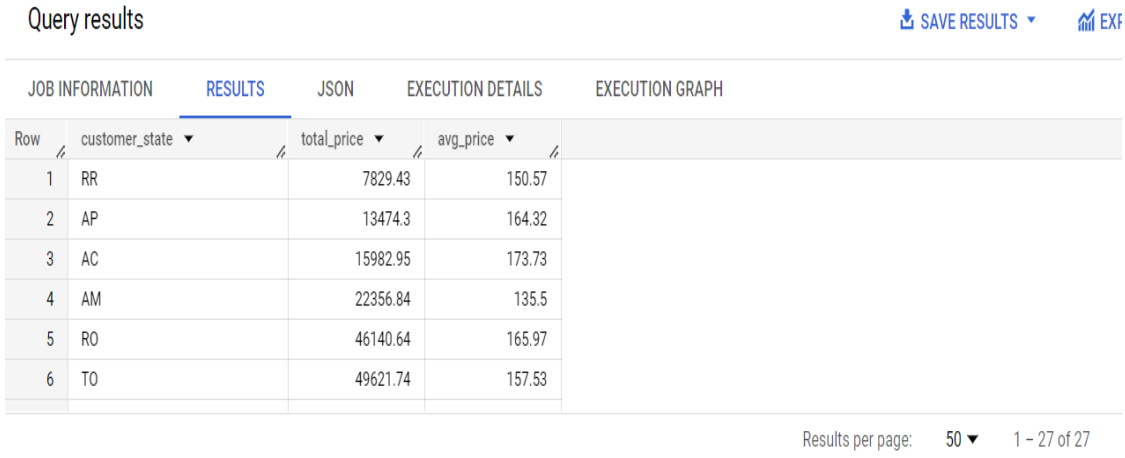
Row	month	percent_increase
1	1	705.1266954171...
2	2	239.9918145445...
3	3	157.7786066709...
4	4	177.8407701149...
5	5	94.62734375677...
6	6	100.2596912456...

For the first month the percentage rise in the cost of orders (2018) is the highest of all but it has fallen drastically in 2nd month. In nutshell from 2nd month they are increasing at decreasing rate. The percentage rise in the cost of orders is on lower side in August,2018.

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

```
SELECT customer_state,
       ROUND(SUM(price),2) as total_price,
       ROUND(AVG(price) ,2) as avg_price
FROM
(
SELECT o.order_id,
       c.customer_state,
       ot.price
FROM `Business_case.customers` c
LEFT OUTER JOIN `Business_case.orders` o
ON c.customer_id = o.customer_id
JOIN `Business_case.order_items` ot
ON o.order_id = ot.order_id
```

```
WHERE o.customer_id IS NOT NULL
) t
GROUP BY customer_state
ORDER BY 2,3 ;
```



Average price for the customers in PB state is the highest which is 191.48 and lowest for customers in SP state which is 109.65.

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

```
SELECT customer_state,
       ROUND(SUM(freight_value),2) as
total_freight_value,
       ROUND(AVG(freight_value) ,2) as
avg_freight_value
FROM
(
SELECT o.order_id,
       c.customer_state,
       ot.freight_value
FROM `Business_case.customers` c
LEFT OUTER JOIN `Business_case.orders` o
ON c.customer_id = o.customer_id
JOIN `Business_case.order_items` ot
ON o.order_id = ot.order_id
WHERE o.customer_id IS NOT NULL
) t
GROUP BY customer_state
ORDER BY 2,3
```

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

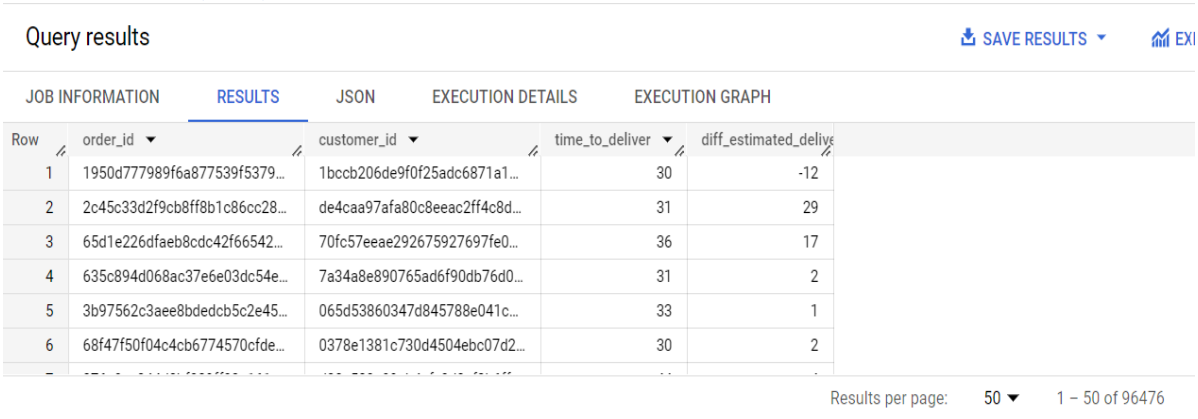
Row	customer_state	total_freight_value	avg_freight_value
1	RR	2235.19	42.98
2	AP	2788.5	34.01
3	AC	3686.75	40.07
4	AM	5478.89	33.21
5	RO	11417.38	41.07
6	TO	11732.68	37.25

Results per page: 50 1 – 27 of 27

The state SP has the lowest average freight cost and RR has the highest average freight charges. This implies that the customers in RR state will have to pay higher cost per order than those of SP state. This might surge the demand of the products.

5. 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.

```
SELECT order_id,
       customer_id,
       DATE_DIFF(date_from_odcd ,date_from_opt ,DAY)
as time_to_deliver,
       DATE_DIFF(date_from_oedd ,date_from_odcd ,DAY)
as diff_estimated_delivery
FROM
(
SELECT order_id,
       customer_id,
       DATE(order_purchase_timestamp) as
date_from_opt,
       DATE(order_delivered_customer_date) as
date_from_odcd,
       DATE(order_estimated_delivery_date) as
date_from_oedd
FROM `Business_case.orders`
WHERE order_delivered_customer_date IS NOT NULL
)t
ORDER BY 3,4 ;
```



Above figure shows the query result before ordering by 3rd and 4th column. It has been deliberately kept in that way to show that, For some customers diff_estimated_delivery is negative i.e. they have received order before the estimated delivery dates.

b) Find out the top 5 states with the highest & lowest average freight value.

```
SELECT *
FROM
(
SELECT customer_state,
       avg_freight_val,
       DENSE_RANK() OVER(ORDER BY avg_freight_val
DESC) as highest_avg_freight_val,
       DENSE_RANK() OVER(ORDER BY avg_freight_val ASC)
as lowest_avg_freight_val
FROM
(
SELECT DISTINCT(customer_state),
       ROUND(AVG(freight_value) OVER(PARTITION BY
customer_state ) , 2) as avg_freight_val
FROM
(
SELECT o.order_id,
       c.customer_state,
       ot.freight_value
FROM `Business_case.customers` c
LEFT OUTER JOIN `Business_case.orders` o
ON c.customer_id = o.customer_id
JOIN `Business_case.order_items` ot
ON o.order_id = ot.order_id
WHERE o.customer_id IS NOT NULL
) t
) t1
) t2
WHERE highest_avg_freight_val <=5
OR lowest_avg_freight_val <= 5
ORDER BY
highest_avg_freight_val,lowest_avg_freight_val DESC
```

RR to PI are the states with highest average freight price and rest all are the top 5 states with the lowest freight price.

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

Row	customer_state	avg_freight_val	highest_avg_freight_	lowest_avg_freight_y
1	RR	42.98	1	27
2	PB	42.72	2	26
3	RO	41.07	3	25
4	AC	40.07	4	24
5	PI	39.15	5	23
6	DF	21.04	23	5
7	RJ	20.96	24	4
8	MG	20.63	25	3

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

```
SELECT customer_state,
       lowest_avg_delivery_time,
       highest_avg_delivery_time
FROM
(
SELECT customer_state,
       avg_delivery_time,
       DENSE_RANK() OVER(ORDER BY avg_delivery_time
ASC) as lowest_avg_delivery_time,
       DENSE_RANK() OVER(ORDER BY avg_delivery_time
DESC) as highest_avg_delivery_time
FROM
(
SELECT c.customer_state ,
       ROUND(AVG(DATE_DIFF(o.order_delivered_customer_
date, o.order_purchase_timestamp, DAY)), 2) as
avg_delivery_time
FROM `Business_case.customers` c
JOIN  `Business_case.orders` o
ON c.customer_id = o.customer_id
WHERE o.order_delivered_customer_date IS NOT NULL
GROUP BY c.customer_state
) t
) t1
WHERE highest_avg_delivery_time <= 5
OR lowest_avg_delivery_time <= 5
ORDER BY 2 , 3 ;
```

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

Row	customer_state	lowest_avg_delivery	highest_avg_delivery
1	SP	1	27
2	PR	2	26
3	MG	3	25
4	DF	4	24
5	SC	5	23
6	PA	23	5
7	AL	24	4
8	AM	25	3

The SP to SC states have the lowest average delivery time and other 5 states have the highest average delivery time.

d) Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery.

```
SELECT customer_state,
       avg_delivery_diff_time,
       DENSE_RANK() OVER(ORDER BY
avg_delivery_diff_time) as ranks_
FROM
(
SELECT c.customer_state ,
       ROUND(AVG(DATE_DIFF(order_estimated_delivery_da
te,o.order_delivered_customer_date, DAY)), 2) as
avg_delivery_diff_time
FROM `Business_case.customers` c
JOIN  `Business_case.orders` o
ON c.customer_id = o.customer_id
WHERE o.order_delivered_customer_date IS NOT NULL
GROUP BY c.customer_state
) t
ORDER BY 3
LIMIT 5 ;
```

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	customer_state	avg_delivery_diff_tim	ranks_
1	AL	7.95	1
2	MA	8.77	2
3	SE	9.17	3
4	ES	9.62	4
5	BA	9.93	5

6. Analysis based on the payments:

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

```
SELECT p.payment_type,
       EXTRACT(MONTH FROM o.order_purchase_timestamp)
AS month_,
       COUNT(DISTINCT o.order_id) AS no_of_orders
FROM `Business_case.orders` o
JOIN `Business_case.payments` p
ON o.order_id = p.order_id
GROUP BY 1, 2
ORDER BY 1, 2;
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	payment_type	month_	no_of_orders
1	UPI	1	1715
2	UPI	2	1723
3	UPI	3	1942
4	UPI	4	1783
5	UPI	5	2035
6	UPI	6	1807

The credit card transactions are the highest , followed by UPI. Whereas Debit Card transactions are the least preferred. This might be due to No Cost EMI, Higher Discount Offers on the credit card.

b) Find the no. of orders placed on the basis of the payment installments that have been paid.

```
SELECT p.payment_installments ,
       COUNT(o.order_id) as no_of_orders
FROM `Business_case.payments` p
JOIN `Business_case.orders` o
ON o.order_id = p.order_id
WHERE lower(o.order_status) != 'canceled'
GROUP BY 1
ORDER BY 1 , 2 ;
```

JOB INFORMATION | RESULTS | JSON

Row	payment_installment	no_of_orders
1	0	2
2	1	52184
3	2	12353
4	3	10392
5	4	7056
6	5	5209

Majority of orders have only one installment. The maximum number of installments are 24 and such orders are 18.

ACTIONABLE INSIGHTS

1. The SP state has the highest number of orders and customer base than the other states. This is actually an indication of improvement and strategy change in the other states.
2. In some states Average Freight Value is on a higher side. This shows a flaw in supply chain and can impact significantly on demand.
3. Some areas require better shipping facilities to bring down the actual delivery timing, since longer delivery time impacts a company's ability to retain customers.
4. In some months the demand of the products is at peak especially in months like August. This might be due to the festivals. In order to cater this demand a well organised marketing and sales strategy is required.
5. The data indicates fall in demand during September and October. To boost up the demand discounts have to be offered to incentivise the customers.

Recommendations

1. Improving the logistics and shipping processes would automatically lower the delivery time. This would encourage the customer to purchase more and improve customer satisfaction.
2. In some states pricing and freight values have to be reconsidered. Lowering them would make products available at competitive rates as well as customer retention. This would also help in maximising the revenue as firms would become efficient and reduce cost of production.

3. Collaborating with better payment platforms in order to bring ease in mode of payment. This would also enable customers to track transaction in real-time.
4. Building the robust digital infrastructure, better e-commerce platform or website so that it can represent the product characteristics more vibrantly or tell whether the product is in stock or not, etc.
5. Increase use of AI and Recommendation mechanism to recommend the products more accurately as per customer preferences.