BUSINESS CASE STUDY

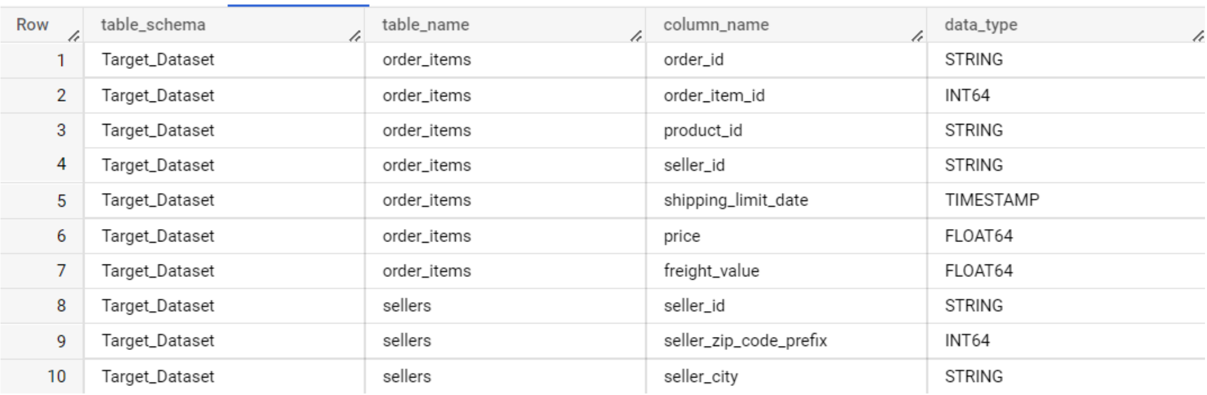
Target SQL

Q.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

Ans:

```
SELECT table_schema,
       table_name,
       column_name,
       data_type
FROM   target_dataset.information_schema.columns;
```

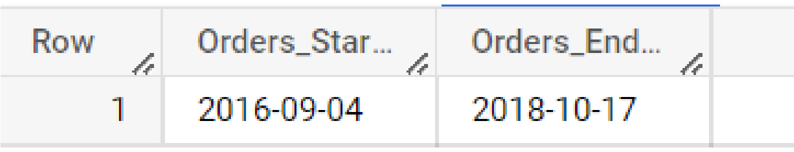


Row	table_schema	table_name	column_name	data_type
1	Target_Dataset	order_items	order_id	STRING
2	Target_Dataset	order_items	order_item_id	INT64
3	Target_Dataset	order_items	product_id	STRING
4	Target_Dataset	order_items	seller_id	STRING
5	Target_Dataset	order_items	shipping_limit_date	TIMESTAMP
6	Target_Dataset	order_items	price	FLOAT64
7	Target_Dataset	order_items	freight_value	FLOAT64
8	Target_Dataset	sellers	seller_id	STRING
9	Target_Dataset	sellers	seller_zip_code_prefix	INT64
10	Target_Dataset	sellers	seller_city	STRING

2. Time period for which the data is given

Ans:

```
SELECT Min(Extract (date FROM order_purchase_timestamp)) AS
       Orders_Start_Time_Period,
       Max(Extract (date FROM order_purchase_timestamp)) AS
       Orders_End_Time_Period
FROM   `target_dataset.orders`;
```



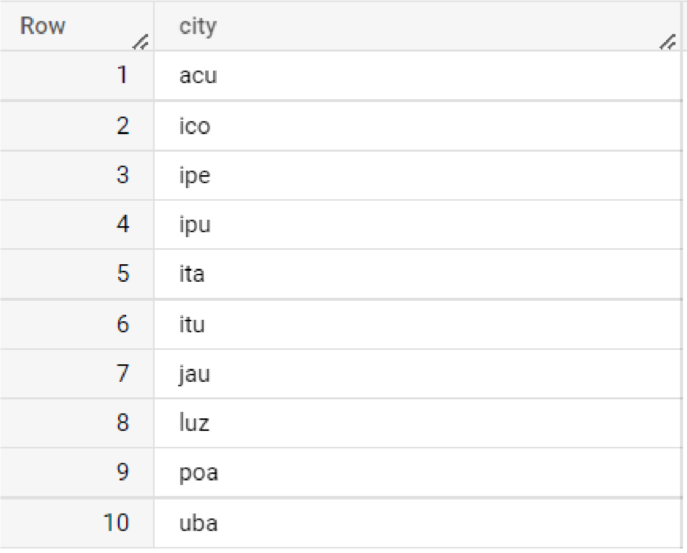
Row	Orders_Star...	Orders_End...
1	2016-09-04	2018-10-17

3. Cities and States covered in the dataset

Ans:

Distinct Cities

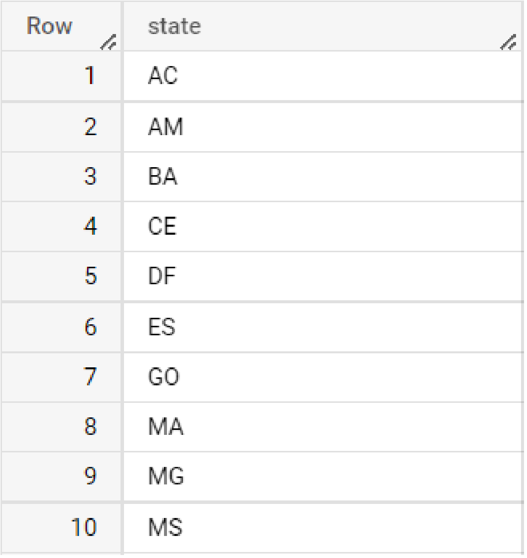
```
SELECT seller_city AS city
FROM   `Target_Dataset.sellers`
UNION DISTINCT
SELECT customer_city AS city
FROM   `Target_Dataset.customers`;
```



Row	city
1	acu
2	ico
3	ipe
4	ipu
5	ita
6	itu
7	jau
8	luz
9	poa
10	uba

Distinct States

```
SELECT seller_state AS state
FROM   `target_dataset.sellers`
UNION DISTINCT
SELECT customer_state AS state
FROM   `target_dataset.customers`;
```



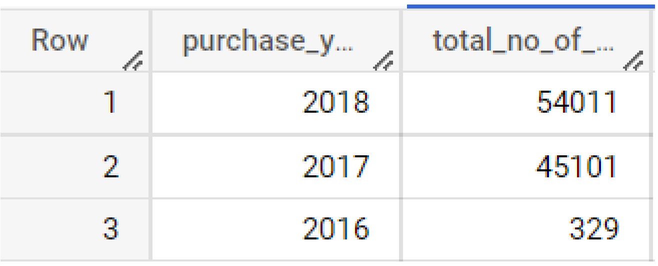
Row	state
1	AC
2	AM
3	BA
4	CE
5	DF
6	ES
7	GO
8	MA
9	MG
10	MS

Q.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?

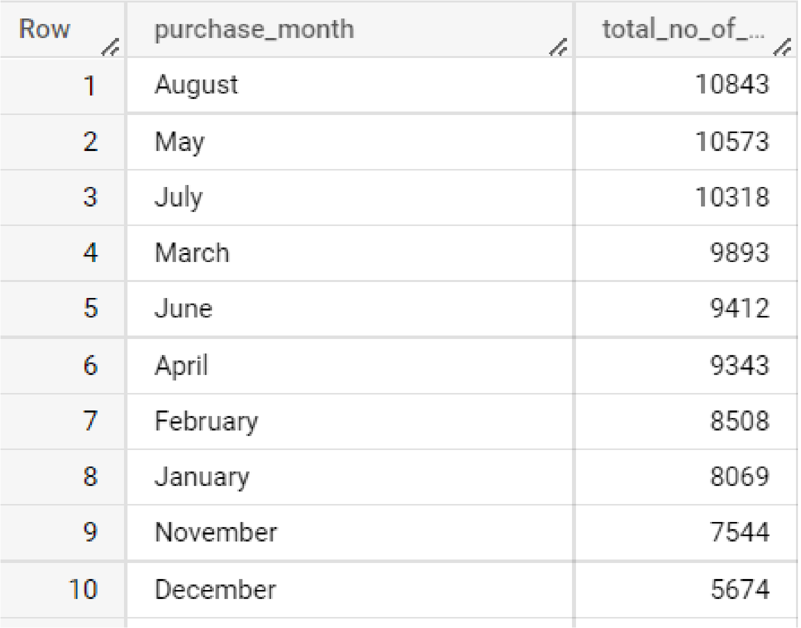
Ans:

```
SELECT Extract(year FROM order_purchase_timestamp) AS purchase_year,
       Count(order_id)                             AS total_no_of_or
ders
FROM   target_dataset.orders
GROUP  BY purchase_year
ORDER  BY total_no_of_orders DESC;
```



Row	purchase_y...	total_no_of_...
1	2018	54011
2	2017	45101
3	2016	329

```
SELECT Format_datetime("%b", order_purchase_timestamp) AS purchase_m
onth,
       Count(order_id)                                  AS total_no_o
f_orders
FROM   target_dataset.orders
GROUP  BY purchase_month
ORDER  BY total_no_of_orders DESC;
```

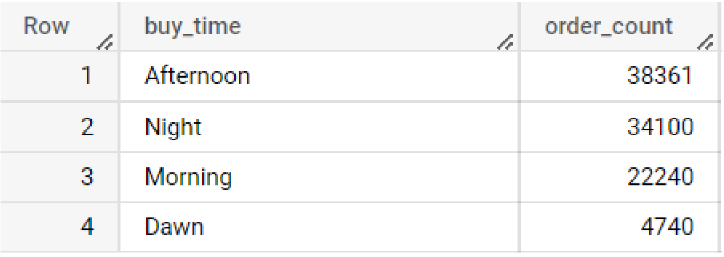


Row	purchase_month	total_no_of_...
1	August	10843
2	May	10573
3	July	10318
4	March	9893
5	June	9412
6	April	9343
7	February	8508
8	January	8069
9	November	7544
10	December	5674

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

Ans:

```
SELECT
CASE
WHEN Extract(hour FROM order_purchase_timestamp) >=00 AND Extract(hour FROM order_purchase_timestamp) <=05 THEN "dawn"
WHEN Extract(hour FROM order_purchase_timestamp) >=06 AND Extract(hour FROM order_purchase_timestamp) <=11 THEN "morning"
WHEN Extract(hour FROM order_purchase_timestamp) >=12 AND Extract(hour FROM order_purchase_timestamp) <=17 THEN "afternoon"
WHEN Extract(hour FROM order_purchase_timestamp) >=18 AND Extract(hour FROM order_purchase_timestamp) <=23 THEN "night"
end AS buy_time,
Count(order_purchase_timestamp) AS order_count
FROM `target_dataset.orders`
GROUP BY buy_time
ORDER BY order_count DESC;
```



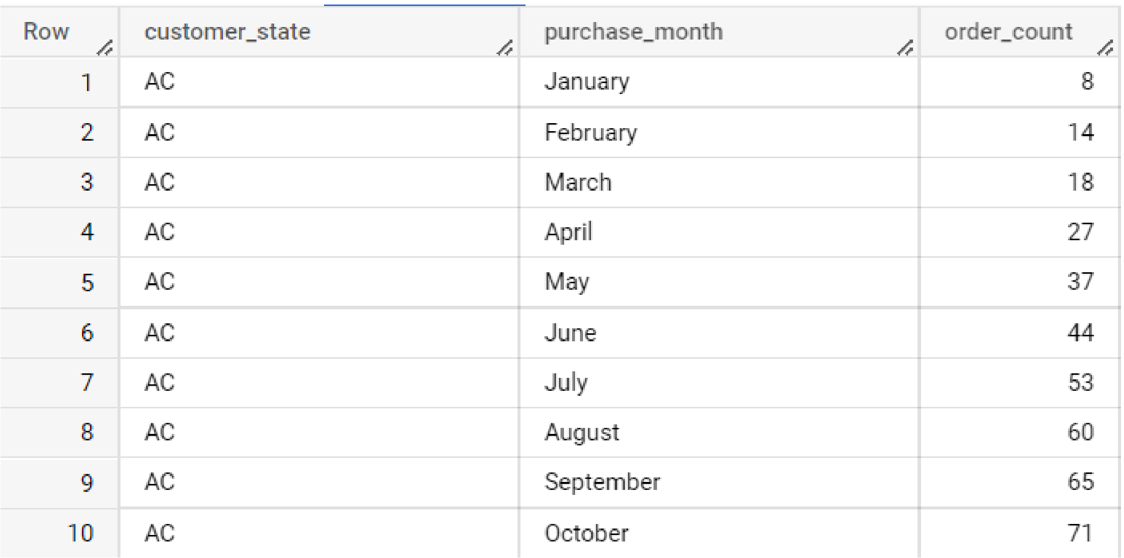
Row	buy_time	order_count
1	Afternoon	38361
2	Night	34100
3	Morning	22240
4	Dawn	4740

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

1. Get month on month orders by region, states

Ans:

```
select distinct
customer_state,
format_datetime("%B", order_purchase_timestamp) as purchase_month,
count(order_id) over(partition by customer_state order by extract(mo
nth from order_purchase_timestamp)) as order_count
from Target_Dataset.orders as o
join Target_Dataset.customers as c on o.customer_id=c.customer_id;
```

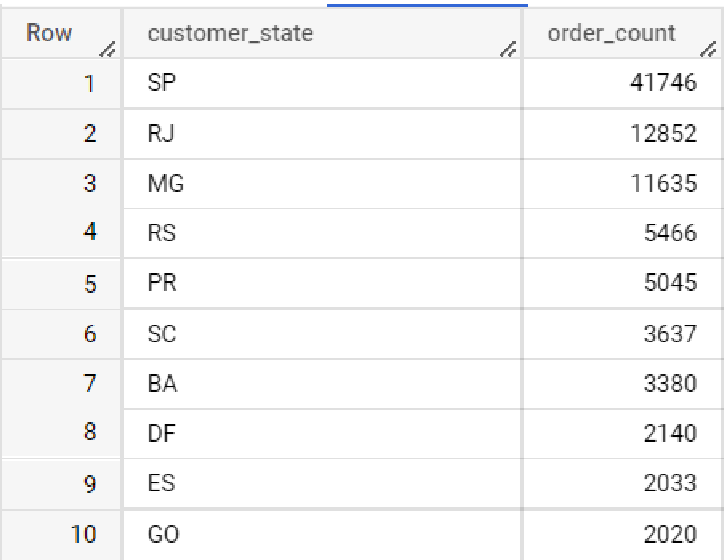


Row	customer_state	purchase_month	order_count
1	AC	January	8
2	AC	February	14
3	AC	March	18
4	AC	April	27
5	AC	May	37
6	AC	June	44
7	AC	July	53
8	AC	August	60
9	AC	September	65
10	AC	October	71

2. How are customers distributed in Brazil

Ans:

```
SELECT c1.customer_state, Count(o1.order_id) AS order_count
FROM `Target_dataset.orders` AS o1
INNER JOIN `Target_dataset.customers` AS c1
ON o1.customer_id = c1.customer_id
GROUP BY c1.customer_state
ORDER BY order_count DESC;
```



Row	customer_state	order_count
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

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

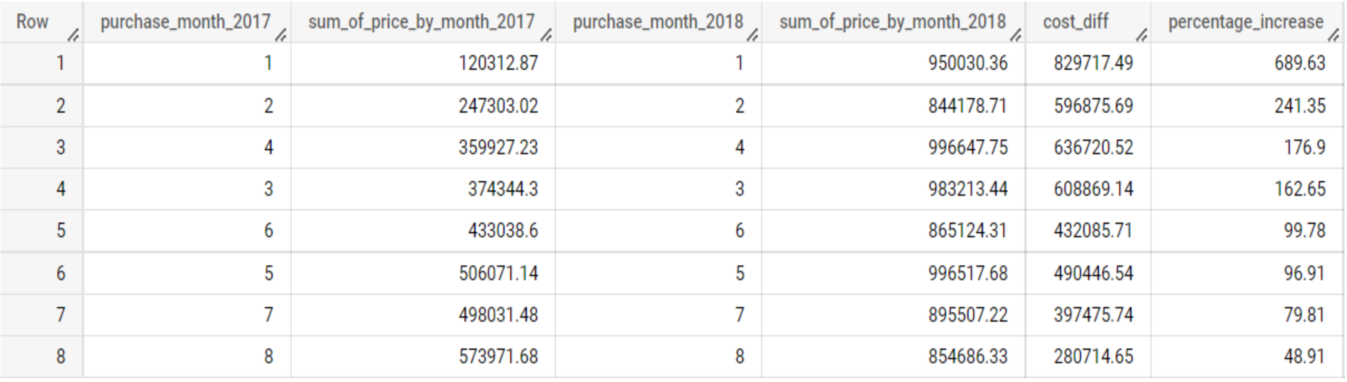
1. Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only)

Ans:

```
WITH
cost_2017 AS
(SELECT
EXTRACT(MONTH FROM o1.order_purchase_timestamp) AS purchase_month_2017,
ROUND(SUM(oi1.price), 2) AS sum_of_price_by_month_2017,
FROM `Target_Dataset.orders` AS o1
INNER JOIN `Target_Dataset.order_items` AS oi1
ON o1.order_id = oi1.order_id
WHERE EXTRACT(YEAR FROM o1.order_purchase_timestamp) = 2017 AND EXTRACT(MON
TH FROM o1.order_purchase_timestamp) <= 8
GROUP BY purchase_month_2017),


cost_2018 AS
(SELECT
EXTRACT(MONTH FROM o1.order_purchase_timestamp) AS purchase_month_2018,
ROUND(SUM(oi1.price), 2) AS sum_of_price_by_month_2018,
FROM `Target_Dataset.orders` AS o1
INNER JOIN `Target_Dataset.order_items` AS oi1
ON o1.order_id = oi1.order_id
WHERE EXTRACT(YEAR FROM o1.order_purchase_timestamp) = 2018 AND EXTRACT(MON
TH FROM o1.order_purchase_timestamp) <= 8
GROUP BY purchase_month_2018)

SELECT *,
ROUND((cost_2018.sum_of_price_by_month_2018 -
 cost_2017.sum_of_price_by_month_2017), 2) AS cost_diff,
ROUND((((cost_2018.sum_of_price_by_month_2018 -
 cost_2017.sum_of_price_by_month_2017) / cost_2017.sum_of_price_by_month_20
17) * 100), 2) AS percentage_increase
FROM cost_2017
INNER JOIN cost_2018
ON cost_2017.purchase_month_2017 = cost_2018.purchase_month_2018
ORDER BY percentage_increase DESC;
```

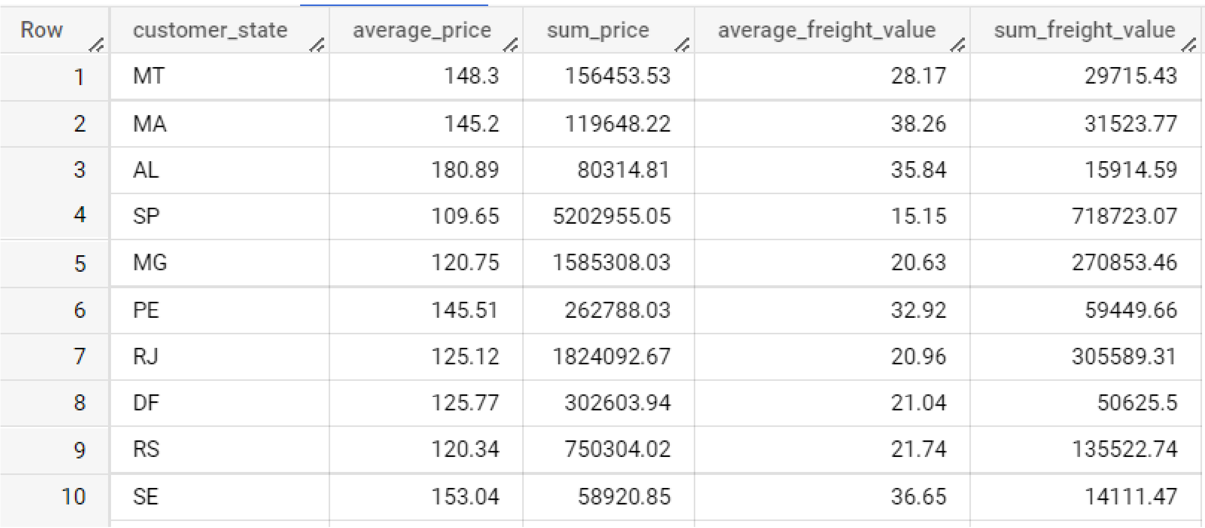


Row	purchase_month_2017	sum_of_price_by_month_2017	purchase_month_2018	sum_of_price_by_month_2018	cost_diff	percentage_increase
1	1	120312.87	1	950030.36	829717.49	689.63
2	2	247303.02	2	844178.71	596875.69	241.35
3	4	359927.23	4	996647.75	636720.52	176.9
4	3	374344.3	3	983213.44	608869.14	162.65
5	6	433038.6	6	865124.31	432085.71	99.78
6	5	506071.14	5	996517.68	490446.54	96.91
7	7	498031.48	7	895507.22	397475.74	79.81
8	8	573971.68	8	854686.33	280714.65	48.91

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

Ans:

```
SELECT c1.customer_state,
       Round(Avg(oi1.price), 2)          AS average_price,
       Round(Sum(oi1.price), 2)          AS sum_price,
       Round(Avg(oi1.freight_value), 2) AS average_freight_value,
       Round(Sum(oi1.freight_value), 2) AS sum_freight_value
FROM   `target_dataset.orders` AS o1
       INNER JOIN `target_dataset.customers` AS c1
               ON o1.customer_id = c1.customer_id
       INNER JOIN `target_dataset.order_items` AS oi1
               ON o1.order_id = oi1.order_id
GROUP  BY c1.customer_state;
```



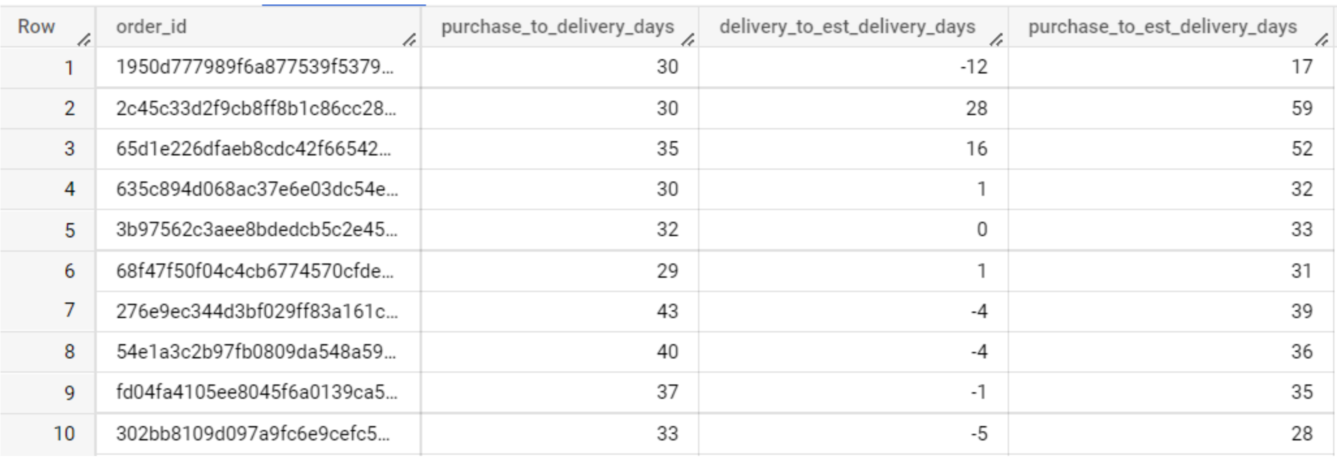
Row	customer_state	average_price	sum_price	average_freight_value	sum_freight_value
1	MT	148.3	156453.53	28.17	29715.43
2	MA	145.2	119648.22	38.26	31523.77
3	AL	180.89	80314.81	35.84	15914.59
4	SP	109.65	5202955.05	15.15	718723.07
5	MG	120.75	1585308.03	20.63	270853.46
6	PE	145.51	262788.03	32.92	59449.66
7	RJ	125.12	1824092.67	20.96	305589.31
8	DF	125.77	302603.94	21.04	50625.5
9	RS	120.34	750304.02	21.74	135522.74
10	SE	153.04	58920.85	36.65	14111.47

Q.5) Analysis on sales, freight and delivery time

1. Calculate days between purchasing, delivering and estimated delivery

Ans:

```
select
order_id,
datetime_diff(order_delivered_customer_date,order_purchase_timestamp
, day) as purchase_to_delivery_days,
datetime_diff(order_estimated_delivery_date,order_delivered_customer
_date, day) as delivery_to_est_delivery_days,
datetime_diff(order_estimated_delivery_date,order_purchase_timestamp
, day) as purchase_to_est_delivery_days
from Target_Dataset.orders
where order_delivered_customer_date is not null;
```

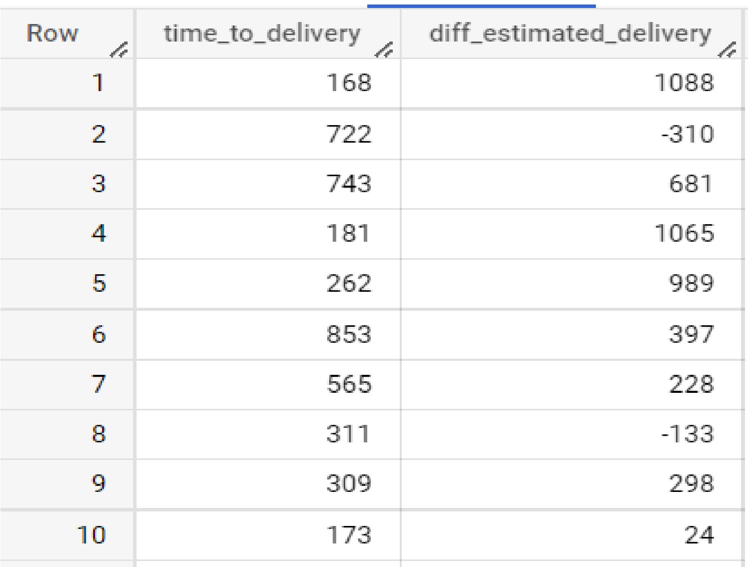


Row	order_id	purchase_to_delivery_days	delivery_to_est_delivery_days	purchase_to_est_delivery_days
1	1950d777989f6a877539f5379...	30	-12	17
2	2c45c33d2f9cb8ff8b1c86cc28...	30	28	59
3	65d1e226dfaeb8cdc42f66542...	35	16	52
4	635c894d068ac37e6e03dc54e...	30	1	32
5	3b97562c3aee8bdedcb5c2e45...	32	0	33
6	68f47f50f04c4cb6774570cfde...	29	1	31
7	276e9ec344d3bf029ff83a161c...	43	-4	39
8	54e1a3c2b97fb0809da548a59...	40	-4	36
9	fd04fa4105ee8045f6a0139ca5...	37	-1	35
10	302bb8109d097a9fc6e9cefc5...	33	-5	28

2. Create columns:

- time_to_delivery = order_purchase_timestamp-order_delivered_customer_date
- diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date

```
SELECT Datetime_diff(order_delivered_customer_date, order_purchase_t
imestamp,
       hour) AS
       time_to_delivery,
       Datetime_diff(order_estimated_delivery_date,
       order_delivered_customer_date, hour
       )
       AS diff_estimated_delivery
FROM   target_dataset.orders
WHERE  order_delivered_customer_date IS NOT NULL;
```

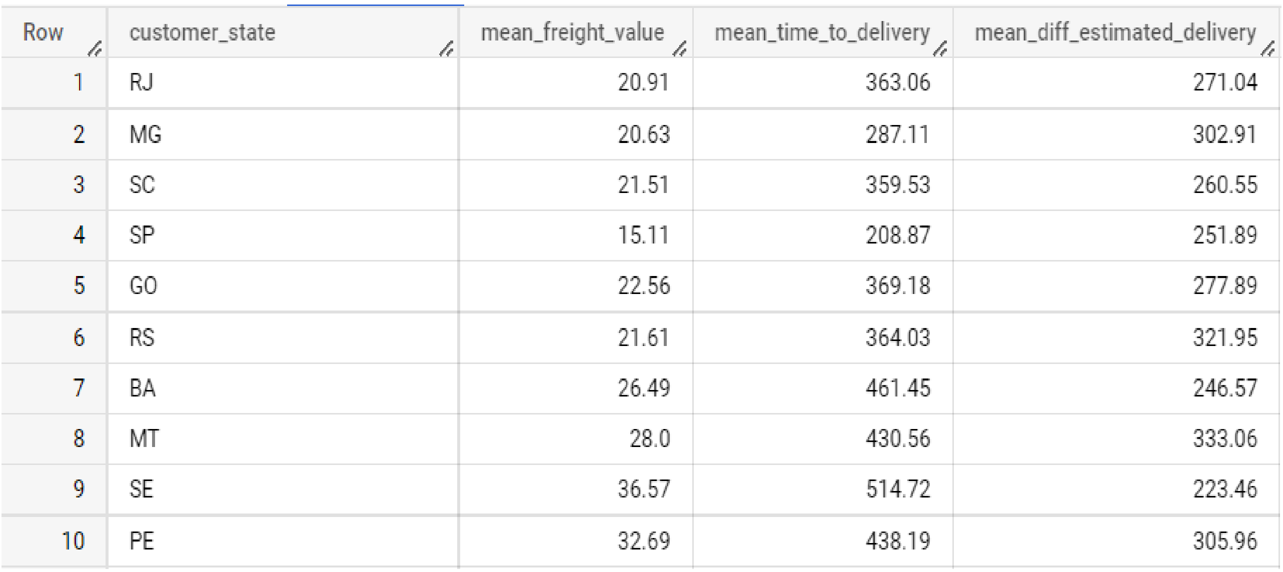


Row	time_to_delivery	diff_estimated_delivery
1	168	1088
2	722	-310
3	743	681
4	181	1065
5	262	989
6	853	397
7	565	228
8	311	-133
9	309	298
10	173	24

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

Ans:

```
WITH cte_o_oi_c
     AS (SELECT c.customer_state,
                oi.freight_value,
                Datetime_diff(o.order_delivered_customer_date,
                o.order_purchase_timestamp, hour)
                      AS time_to_delivery,
                Datetime_diff(o.order_estimated_delivery_date,
                o.order_delivered_customer_date,
                hour) AS diff_estimated_delivery
         FROM   target_dataset.orders AS o
                JOIN target_dataset.order_items AS oi
                  ON o.order_id = oi.order_id
                JOIN target_dataset.customers AS c
                  ON o.customer_id = c.customer_id
         WHERE  order_delivered_customer_date IS NOT NULL)
SELECT customer_state,
       Round(Avg(freight_value), 2)           AS mean_freight_value,
       Round(Avg(time_to_delivery), 2)        AS mean_time_to_delivery,
       Round(Avg(diff_estimated_delivery), 2) AS mean_diff_estimated_delive
ry
FROM   cte_o_oi_c
GROUP  BY customer_state;
```



Row	customer_state	mean_freight_value	mean_time_to_delivery	mean_diff_estimated_delivery
1	RJ	20.91	363.06	271.04
2	MG	20.63	287.11	302.91
3	SC	21.51	359.53	260.55
4	SP	15.11	208.87	251.89
5	GO	22.56	369.18	277.89
6	RS	21.61	364.03	321.95
7	BA	26.49	461.45	246.57
8	MT	28.0	430.56	333.06
9	SE	36.57	514.72	223.46
10	PE	32.69	438.19	305.96

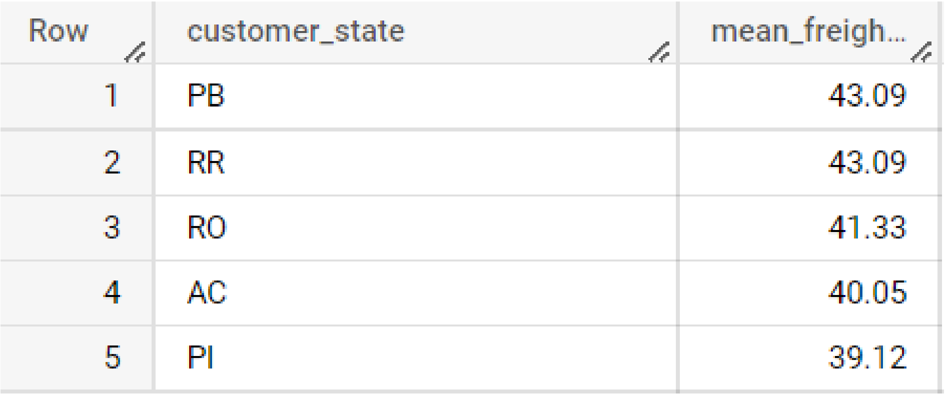
4. Sort the data to get the following:

4.1 Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

Ans:

Top 5 States with hightest average freight value

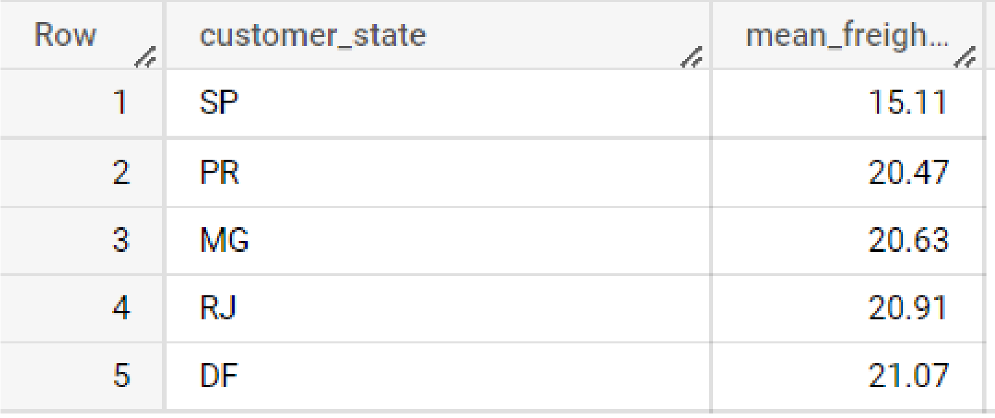
```
WITH mean_cte
AS
  (WITH cte_o_oi_c
AS
  (
         SELECT c.customer_state,
                oi.freight_value,
                datetime_diff(o.order_delivered_customer_date, o.order_p
urchase_timestamp, hour)      AS time_to_delivery,
                datetime_diff(o.order_estimated_delivery_date, o.order_d
elivered_customer_date, hour) AS diff_estimated_delivery
         FROM   target_dataset.orders
                              AS o
         JOIN   target_dataset.order_items
                              AS oi
         ON     o.order_id=oi.order_id
         JOIN   target_dataset.customers AS c
         ON     o.customer_id=c.customer_id
         WHERE  order_delivered_customer_date IS NOT NULL)
  SELECT   customer_state,
           round(avg(freight_value), 2)           AS mean_freight_value,
           round(avg(time_to_delivery), 2)        AS mean_time_to_delive
ry,
           round(avg(diff_estimated_delivery), 2) AS mean_diff_estimated
_delivery
  FROM     cte_o_oi_c
  GROUP BY customer_state)
SELECT   customer_state,
         mean_freight_value
FROM     mean_cte
ORDER BY mean_freight_value DESC
LIMIT    5;
```



Row	customer_state	mean_freigh...
1	PB	43.09
2	RR	43.09
3	RO	41.33
4	AC	40.05
5	PI	39.12

Top 5 States with lowest average freight value

```
WITH mean_cte
AS
  (WITH cte_o_oi_c
AS
  (
         SELECT c.customer_state,
                oi.freight_value,
                datetime_diff(o.order_delivered_customer_date, o.order_purc
hase_timestamp, hour)      AS time_to_delivery,
                datetime_diff(o.order_estimated_delivery_date, o.order_deli
vered_customer_date, hour) AS diff_estimated_delivery
         FROM   target_dataset.orders
                           AS o
         JOIN   target_dataset.order_items
                           AS oi
         ON     o.order_id=oi.order_id
         JOIN   target_dataset.customers AS c
         ON     o.customer_id=c.customer_id
         WHERE  order_delivered_customer_date IS NOT NULL)
  SELECT   customer_state,
           round(avg(freight_value), 2)          AS mean_freight_value,
           round(avg(time_to_delivery), 2)       AS mean_time_to_delivery,
           round(avg(diff_estimated_delivery), 2) AS mean_diff_estimated_de
livery
  FROM     cte_o_oi_c
  GROUP BY customer_state)
SELECT   customer_state,
         mean_freight_value
FROM     mean_cte
ORDER BY mean_freight_value
LIMIT    5;
```



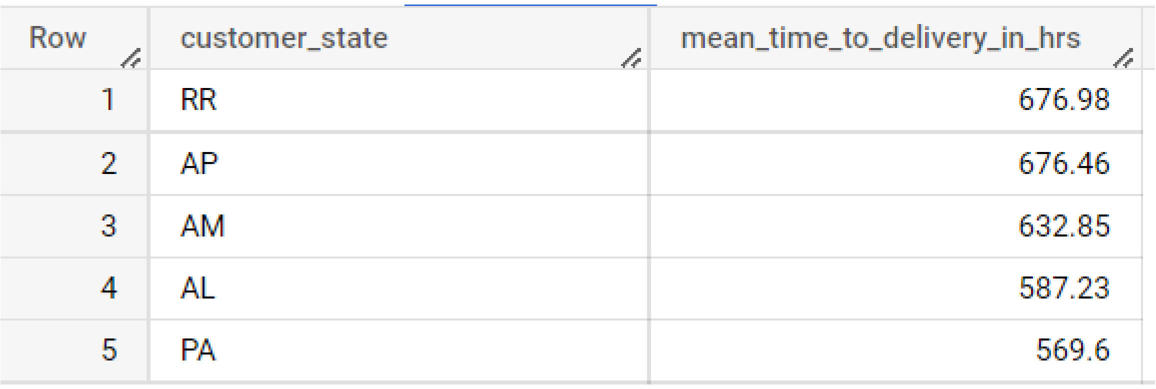
Row	customer_state	mean_freigh...
1	SP	15.11
2	PR	20.47
3	MG	20.63
4	RJ	20.91
5	DF	21.07

4.2 Top 5 states with highest/lowest average time to delivery

Ans:

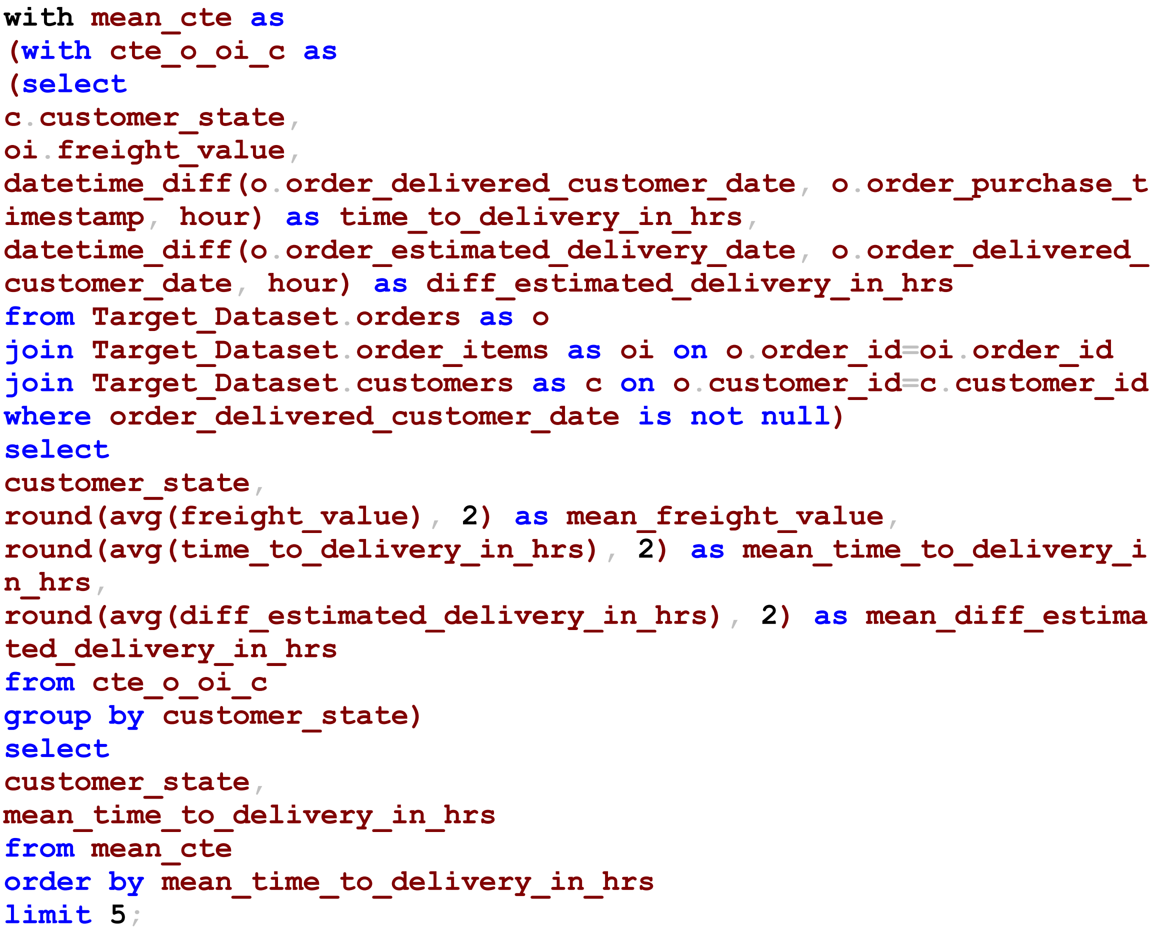
Top 5 States with highest average time to delivery in Hrs

```sql
with mean_cte as
(with cte_o_oi_c as
(select
c.customer_state,
oi.freight_value,
datetime_diff(o.order_delivered_customer_date, o.order_purchase_timestamp, hour) as time_to_delivery_in_hrs,
datetime_diff(o.order_estimated_delivery_date, o.order_delivered_customer_date, hour) as diff_estimated_delivery_in_hrs
from Target_Dataset.orders as o
join Target_Dataset.order_items as oi on o.order_id=oi.order_id
join Target_Dataset.customers as c on o.customer_id=c.customer_id
where order_delivered_customer_date is not null)
select
customer_state,
round(avg(freight_value), 2) as mean_freight_value,
round(avg(time_to_delivery_in_hrs), 2) as mean_time_to_delivery_in_hrs,
round(avg(diff_estimated_delivery_in_hrs), 2) as mean_diff_estimated_delivery_in_hrs
from cte_o_oi_c
group by customer_state)
select
customer_state,
mean_time_to_delivery_in_hrs
from mean_cte
order by mean_time_to_delivery_in_hrs desc
limit 5;
```

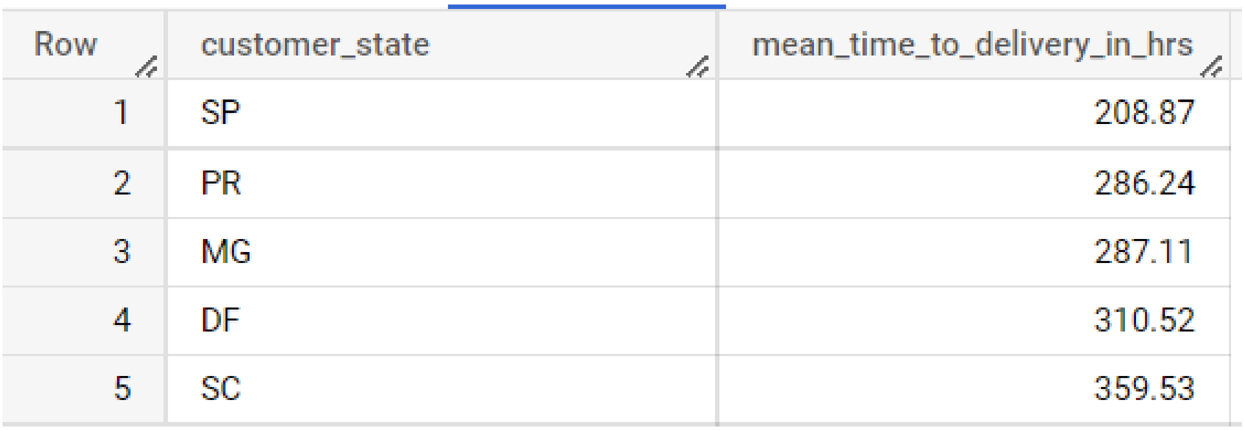


Row	customer_state	mean_time_to_delivery_in_hrs
1	RR	676.98
2	AP	676.46
3	AM	632.85
4	AL	587.23
5	PA	569.6

Top 5 States with lowest average time to delivery in Hrs



```
with mean_cte as
(with cte_o_oi_c as
(select
c.customer_state,
oi.freight_value,
datetime_diff(o.order_delivered_customer_date, o.order_purchase_timestamp, hour) as time_to_delivery_in_hrs,
datetime_diff(o.order_estimated_delivery_date, o.order_delivered_customer_date, hour) as diff_estimated_delivery_in_hrs
from Target_Dataset.orders as o
join Target_Dataset.order_items as oi on o.order_id=oi.order_id
join Target_Dataset.customers as c on o.customer_id=c.customer_id
where order_delivered_customer_date is not null)
select
customer_state,
round(avg(freight_value), 2) as mean_freight_value,
round(avg(time_to_delivery_in_hrs), 2) as mean_time_to_delivery_in_hrs,
round(avg(diff_estimated_delivery_in_hrs), 2) as mean_diff_estimated_delivery_in_hrs
from cte_o_oi_c
group by customer_state)
select
customer_state,
mean_time_to_delivery_in_hrs
from mean_cte
order by mean_time_to_delivery_in_hrs
limit 5;
```



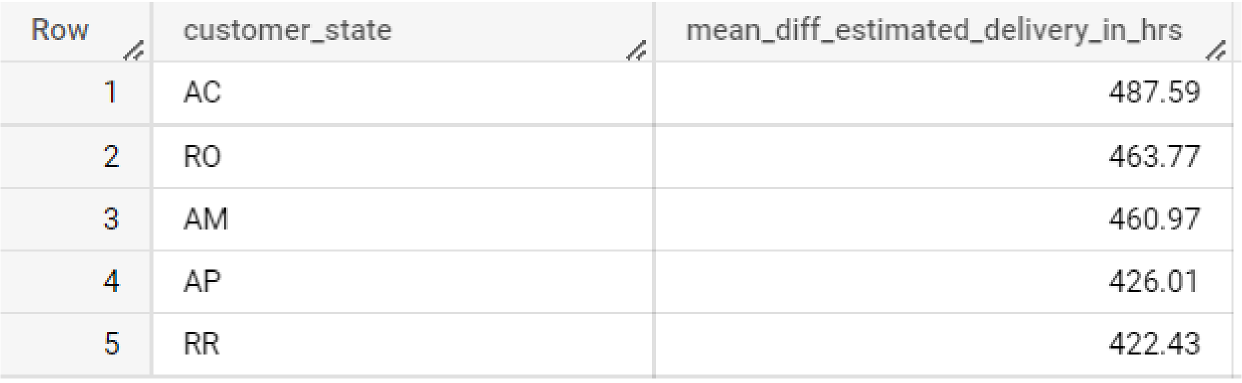
Row	customer_state	mean_time_to_delivery_in_hrs
1	SP	208.87
2	PR	286.24
3	MG	287.11
4	DF	310.52
5	SC	359.53

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

Ans:

Top 5 States where delivery is really fast compared to estimated date

```
WITH mean_cte
AS
  (WITH cte_o_oi_c
AS
  (
         SELECT c.customer_state,
                oi.freight_value,
                datetime_diff(o.order_delivered_customer_date, o.order_purc
hase_timestamp, hour)      AS time_to_delivery_in_hrs,
                datetime_diff(o.order_estimated_delivery_date, o.order_deli
vered_customer_date, hour) AS diff_estimated_delivery_in_hrs
         FROM   target_dataset.orders
                           AS o
         JOIN   target_dataset.order_items
                           AS oi
         ON     o.order_id=oi.order_id
         JOIN   target_dataset.customers AS c
         ON     o.customer_id=c.customer_id
         WHERE  order_delivered_customer_date IS NOT NULL)
  SELECT   customer_state,
           round(avg(freight_value), 2)                     AS mean_freight_va
lue,
           round(avg(time_to_delivery_in_hrs), 2)           AS mean_time_to_de
livery_in_hrs,
           round(avg(diff_estimated_delivery_in_hrs), 2) AS mean_diff_estim
ated_delivery_in_hrs
  FROM     cte_o_oi_c
  GROUP BY customer_state)
SELECT   customer_state,
         mean_diff_estimated_delivery_in_hrs
FROM     mean_cte
ORDER BY mean_diff_estimated_delivery_in_hrs DESC
LIMIT    5;
```



Row	customer_state	mean_diff_estimated_delivery_in_hrs
1	AC	487.59
2	RO	463.77
3	AM	460.97
4	AP	426.01
5	RR	422.43

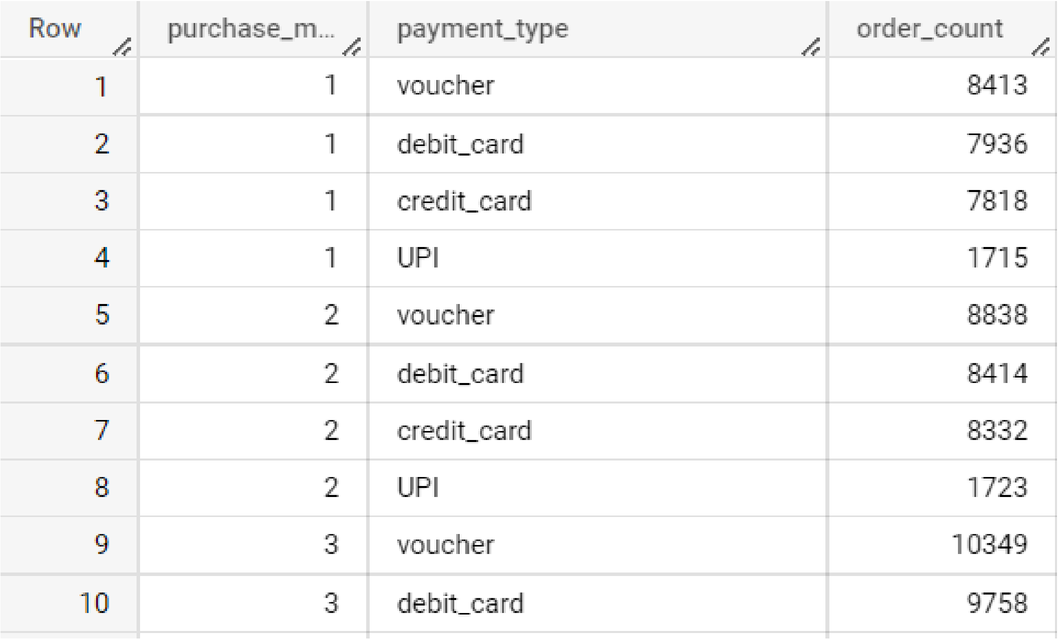
Q.6) Payment type analysis:

1. Month over Month count of orders for different payment types

Ans:

```
SELECT
DISTINCT EXTRACT(MONTH FROM o1.order_purchase_timestamp) AS purchase
_month,
p1.payment_type,
COUNT(o1.order_id) OVER (PARTITION BY EXTRACT(MONTH FROM o1.order_pu
rchase_timestamp) ORDER BY p1.payment_type) AS order_count

FROM `Target_Ecommerce.orders` AS o1
INNER JOIN `Target_Ecommerce.payments` AS p1
ON o1.order_id = p1.order_id
ORDER BY purchase_month, order_count DESC;
```



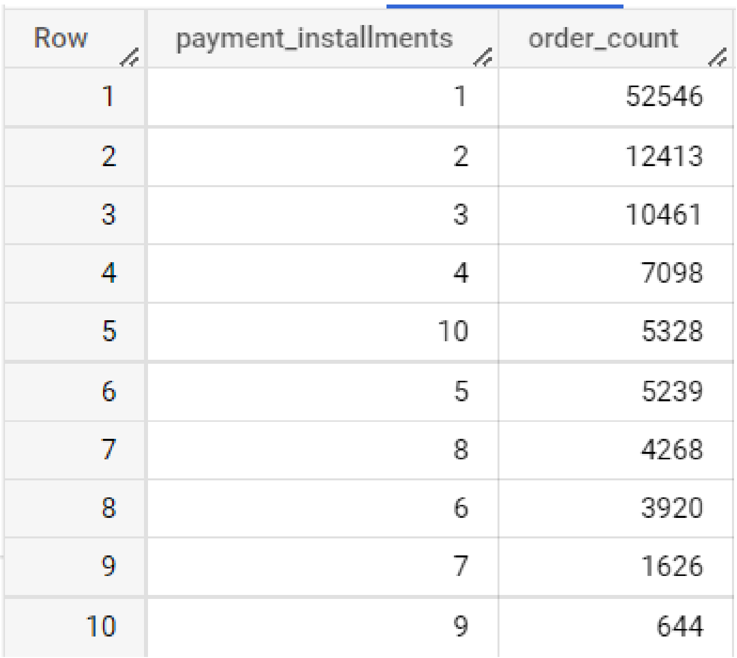
Row	purchase_m...	payment_type	order_count
1	1	voucher	8413
2	1	debit_card	7936
3	1	credit_card	7818
4	1	UPI	1715
5	2	voucher	8838
6	2	debit_card	8414
7	2	credit_card	8332
8	2	UPI	1723
9	3	voucher	10349
10	3	debit_card	9758

2. Distribution of payment installments and count of orders

Ans:

```
SELECT
DISTINCT p1.payment_installments,
COUNT(p1.order_id) AS order_count

FROM `Target_Ecommerce.payments` AS p1
GROUP BY p1.payment_installments
ORDER BY order_count DESC;
```



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

INSIGHTS

1. As per this dataset Target has operations in almost 4196 cities situated in 26 States.
2. There is a growing trend on e-commerce in Brazil, as per the analysis on yearly basis the number of orders have grown enormously since 2016 to 2018.
3. Highest number of orders are placed in August, followed by May and July, these months fall under winter season.
4. Brazilian customers tend to buy in the Afternoon and Night, some customers also buy in Morning but the ratio is not that good compared to Afternoon and Night, customers avoid to buy in the dawn.
5. After analysing the data I found out that the count of orders has grown each and every passing month throughout the years 2016, 2017 and 2018 gradually. This depicts that the customers in Brazil are interested in buying on e-commerce platforms.
6. 40% of the orders in Brazil are from Sao Paulo, 12% are from Rio de Janeiro and 11% from Minas Gerais. These 3 States combined has 63% orders and rest of the orders are from remaining states.
7. Highest increase in cost of orders is in the month January 2018 which is 600% compared to January 2017, followed by February, March and April with more than 150% increase in cost of orders.
8. Top 5 States with highest average freight value are State of Paraiba, State of Roraima, State of Rondonia, Acre and Piaui. Top among these is State of Paraiba and Roraima with highest mean freight value of 43.09.
9. Top 5 States with lowest average freight value are Sau Paulo, State of Parana, Minas Gerais, Rio de Janeiro and Federal District. Top among these is Sau Paulo with lowest mean freight value of 15.11.
10. Roraima and Amapa are the states where time taken for delivery of the goods is maximum whereas, Sau Paulo is the state where minimum time is taken for delivery of the goods.
11. Maximum number of orders are places using Vouchers followed by Debit cards and Credit cards, seems like UPI payment option is not appreciated by Brazilian customers.

RECOMMENDATIONS

1. As the highest number of orders are placed in August, May and July which fall in winter, Target should focus more on these months.
2. Brazilian customers buy more in the afternoon and night as compared to morning and dawn, as it is working well for Target, they should start triggering the customers through offers on the products in the Night and Afternoon to increase their sales even to the next level.
3. Sau Paulo, Rio de Janeiro and Minas Gerais are the states with maximum number of orders as these are highly populated states and people here are interested in e-commerce but Target should also focus on some moderately populated states as Bahia, Rio Grande do Sul, Parana as this states can be their next big market to capture.
4. Looking at the increase in cost of orders from the years 2017 and 2018, seems like 1st quarter of year looks good in terms of increasing the cost of orders, Target should continue as it is for the 1st quarter months as it is working pretty well for them.
5. Customers mostly use vouchers, debit and credit cards as payment options, target should also focus on UPI payment methods as these are more secure and easy to use. Smartphones are pretty common these days even in small cities or towns, Target focusing on UPI payments and giving vouchers or discount offers with UPI payment can attract a lot of customers from small cities as well as small towns.
6. Target should also focus on high mean average freight value as it is too much in states like State of Paraiba, State of Roraima, State of Rondonia, Acre and Piaui. Focusing on reducing this cost can save them a lot of money and will also benefit the organisation in increasing the profits even more.
7. Target can also focus on reducing the time taken for delivery of goods in the states like Roraima and Amapa. States like Sau Paulo, Rio de Janeiro and Mina Gerais which are pretty big and populated states should be provided with one day delivery option for that logistics network should be strong.