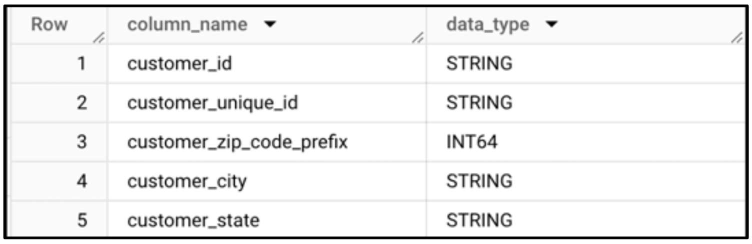
BUSINESS CASE : TARGET SQL

MOHIT CHAND

1.1 Data type of all columns in the “customers” table.

QUERY:

```
SELECT column_name,data_type
FROM `target.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name='customers';
```

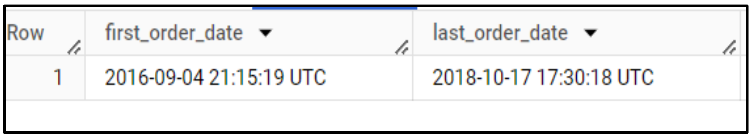


Row	column_name	data_type
1	customer_id	STRING
2	customer_unique_id	STRING
3	customer_zip_code_prefix	INT64
4	customer_city	STRING
5	customer_state	STRING

1.2. Get the time range between which the orders were placed.

QUERY:

```
SELECT
MIN(order_purchase_timestamp) first_order_date,
MAX(order_purchase_timestamp) last_order_date
FROM `target.orders`;
```



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

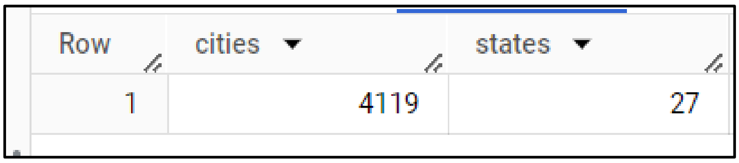
INSIGHTS:

- The first order was placed on 2016-09-04 and the last order was placed on 2018-10-17.

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

QUERY:

```
SELECT
COUNT(DISTINCT c.customer_city) AS cities,
COUNT(DISTINCT c.customer_state) AS states
FROM `target.customers` c
JOIN `target.orders` o
ON c.customer_id = o.customer_id;
```



Row	cities	states
1	4119	27

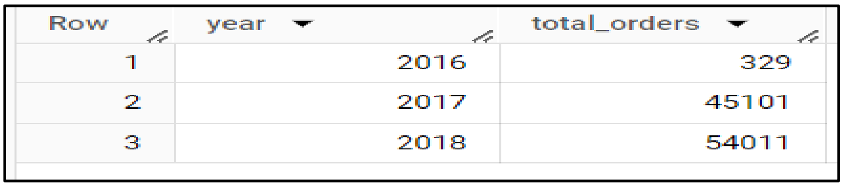
INSIGHTS:

- People from 27 different states ordered.
- Further drilling down we can see people from 4,119 different cities have ordered.

2.1. Is there a growing trend in the no. of orders placed over the past years?

QUERY:

```
SELECT
EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
COUNT(order_id) AS orders
FROM `target.orders`
GROUP BY year
ORDER BY year;
```



Row	year	total_orders
1	2016	329
2	2017	45101
3	2018	54011

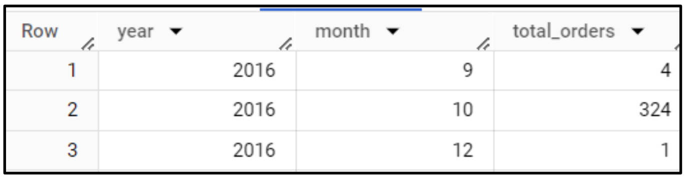
Insights:

- There is a sudden increase in orders from 2016 to 2017 approax
- The increase in orders from 2017 to 2018 is approximately 20%.

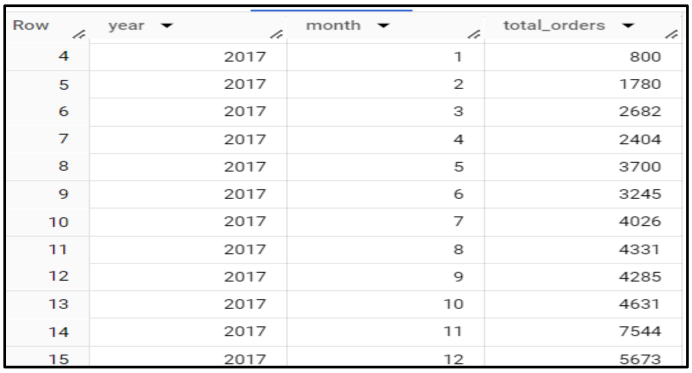
2.2. Can we see some kind of monthly seasonality in terms of the no. of orders being placed?

QUERY:

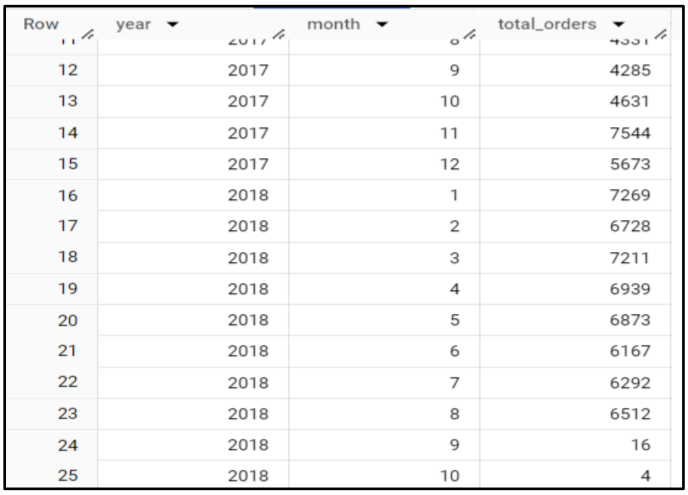
```
SELECT
EXTRACT(year FROM order_purchase_timestamp) AS year,
EXTRACT(Month FROM order_purchase_timestamp) AS month,
COUNT(order_id) AS total_orders
FROM `target.orders`
GROUP BY year,month
ORDER BY year,month;
```



Row	year	month	total_orders
1	2016	9	4
2	2016	10	324
3	2016	12	1



Row	year	month	total_orders
4	2017	1	800
5	2017	2	1780
6	2017	3	2682
7	2017	4	2404
8	2017	5	3700
9	2017	6	3245
10	2017	7	4026
11	2017	8	4331
12	2017	9	4285
13	2017	10	4631
14	2017	11	7544
15	2017	12	5673



Row	year	month	total_orders
11	2017	8	4331
12	2017	9	4285
13	2017	10	4631
14	2017	11	7544
15	2017	12	5673
16	2018	1	7269
17	2018	2	6728
18	2018	3	7211
19	2018	4	6939
20	2018	5	6873
21	2018	6	6167
22	2018	7	6292
23	2018	8	6512
24	2018	9	16
25	2018	10	4

INSIGHTS:

- November 2017 and January 2018 saw the highest number of orders, suggesting possible holiday.
- Orders consistently dropped from February to June 2018, indicating a potential off-season period.
- The drastic fall in orders in September and October 2018 (16 and 4 orders) suggests business disruptions.

Recommendations:

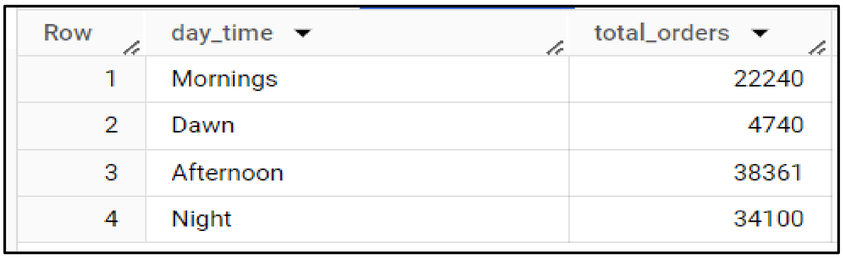
- November 2017 and January 2018 saw the highest number of orders, suggesting possible holiday. Recommend Increasing the number of delivery executives and maintaining higher inventory levels during these months to accommodate the anticipated surge in demand.

2.3. 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

QUERY:

```
 WITH CTE AS (
  SELECT *,
  EXTRACT(HOUR FROM order_purchase_timestamp) AS ordered_hour
  FROM `target.orders`
),
CTE2 AS(
  SELECT *,
    CASE
      WHEN ordered_hour BETWEEN 0  AND 5 THEN 'Dawn'
      WHEN ordered_hour BETWEEN 6  AND 11 THEN 'Mornings'
      WHEN ordered_hour BETWEEN 12  AND 17 THEN 'Afternoon'
      WHEN ordered_hour BETWEEN 17  AND 23 THEN 'Night'
    END AS day_time
  FROM CTE
)
SELECT
day_time,COUNT(order_id) AS total_orders
FROM CTE2
GROUP BY day_time;
```



Row	day_time	total_orders
1	Mornings	22240
2	Dawn	4740
3	Afternoon	38361
4	Night	34100

Insights:

- The data indicates that the 13:00-18:00 (Afternoon) time window is the most preferred period for placing orders.

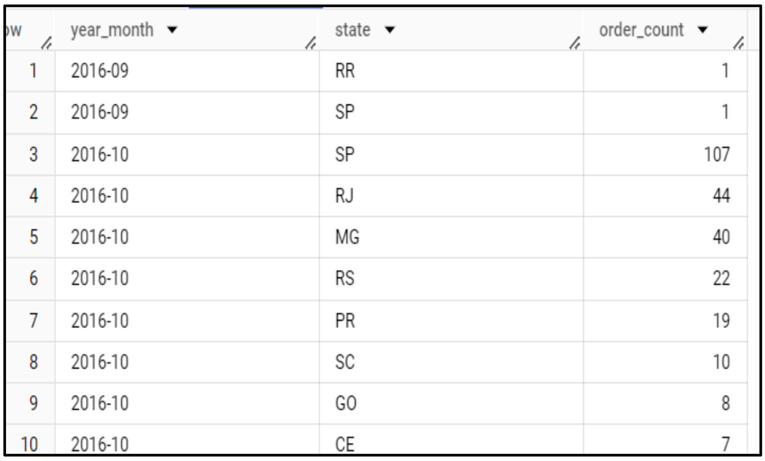
Recommendations:

- It is recommended to allocate additional customer support executives during these hours to effectively manage the increased demand and address any troubleshooting issues that may arise with the e commerce platform.

3.1. Get the month-on-month no. of orders placed in each state.

QUERY:

```
SELECT format_date('%Y-%m',o.order_purchase_timestamp) AS year_month,
c.customer_state as state,count(o.order_id)AS order_count
FROM `target.orders` o
JOIN `target.customers` c
ON c.customer_id=o.customer_id
WHERE LOWER(o.order_status)!='canceled'
GROUP BY year_month,state
ORDER BY year_month,order_count DESC;
```



ow	year_month	state	order_count
1	2016-09	RR	1
2	2016-09	SP	1
3	2016-10	SP	107
4	2016-10	RJ	44
5	2016-10	MG	40
6	2016-10	RS	22
7	2016-10	PR	19
8	2016-10	SC	10
9	2016-10	GO	8
10	2016-10	CE	7

Insights

- São Paulo has experienced a notable surge in order volume during the holiday and festive seasons.

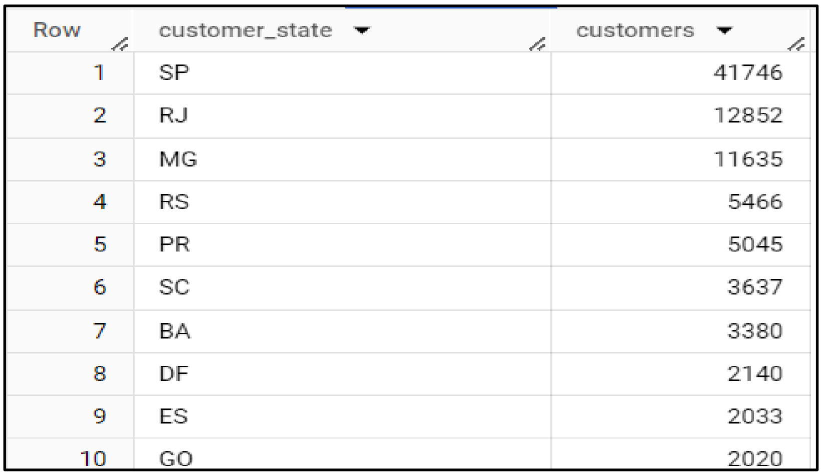
Recommendations

- It is advisable to increase the number of delivery personnel and boost inventory levels in São Paulo to meet the rising demand.

3.2. How are the customers distributed across all the states?

QUERY:

```
SELECT
customer_state,
COUNT(customer_id) customers
FROM `target.customers`
GROUP BY customer_state
ORDER BY customers DESC;
```



Row	customer_state	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

INSIGHTS:

- Customers from SP account for nearly more than the combined total of the next six states.

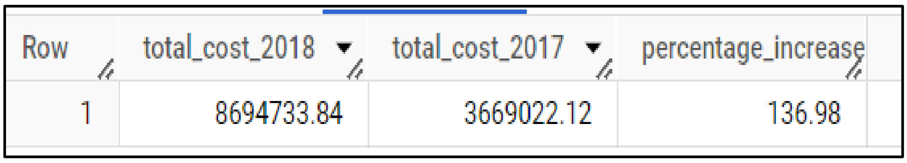
Recommendations:

- It is recommended to scale up the number of delivery executives and increase inventory levels in São Paulo to accommodate 41746 unique customers.

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

QUERY:

```
WITH filtered_orders AS (
    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_cost
    FROM `target.orders` o
    JOIN `target.payments` p
    ON o.order_id = p.order_id
    WHERE
    EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8
    AND EXTRACT(YEAR FROM o.order_purchase_timestamp) IN (2017, 2018)
    GROUP BY year, month
),
yearly_cost AS (
    SELECT year,
    SUM(total_cost) AS yearly_total_cost
    FROM filtered_orders
    GROUP BY year
)
SELECT
ROUND((SELECT yearly_total_cost FROM yearly_cost WHERE year = 2018),2) AS
total_cost_2018,
ROUND((SELECT yearly_total_cost FROM yearly_cost WHERE year = 2017),2) AS
total_cost_2017,
ROUND(((SELECT yearly_total_cost FROM yearly_cost WHERE year = 2018) -
      (SELECT yearly_total_cost FROM yearly_cost WHERE year = 2017)) /
      (SELECT yearly_total_cost FROM yearly_cost WHERE year = 2017) * 100,
2) AS percentage_increase
FROM yearly_cost
LIMIT 1;
```



Row	total_cost_2018	total_cost_2017	percentage_increase
1	8694733.84	3669022.12	136.98

Insights:

- A substantial increase of 136.98% in the total payment value of orders was observed from 2017 to 2018, for the months between Jan to Aug.

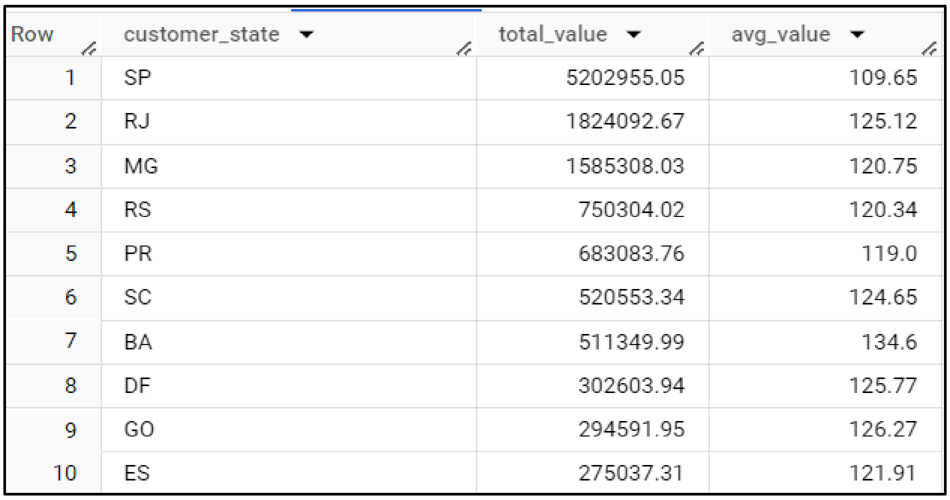
Recommendations:

- Optimize inventory, logistics, and marketing efforts during January to August to capitalize on the 136.98% growth in order value.

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

QUERY:

```
SELECT c.customer_state,
ROUND(SUM(oi.price),2) AS total_value,
ROUND(AVG(oi.price),2) AS avg_value
FROM `target.customers` c
JOIN `target.orders` o
ON c.customer_id = o.customer_id
JOIN `target.order_items` oi
ON o.order_id = oi.order_id
GROUP BY c.customer_state
ORDER BY total_value DESC,avg_value DESC;
```



Row	customer_state	total_value	avg_value
1	SP	5202955.05	109.65
2	RJ	1824092.67	125.12
3	MG	1585308.03	120.75
4	RS	750304.02	120.34
5	PR	683083.76	119.0
6	SC	520553.34	124.65
7	BA	511349.99	134.6
8	DF	302603.94	125.77
9	GO	294591.95	126.27
10	ES	275037.31	121.91

Insights:

- São Paulo leads in total order value to date, followed by Rio de Janeiro and Minas Gerais.

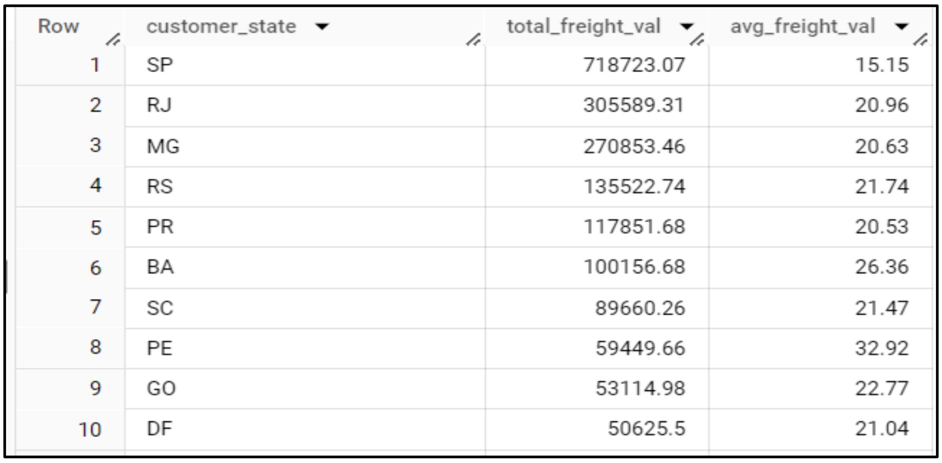
Recommendations:

- Focus on resource allocation and marketing initiatives in São Paulo, while expanding targeted growth strategies in Rio de Janeiro and Minas Gerais to capitalize on their potential and maximize overall revenue.

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

QUERY:

```
SELECT c.customer_state,
ROUND(SUM(i.freight_value),2) AS total_freight_val,
ROUND(AVG(i.freight_value),2) AS avg_freight_val
FROM `target.customers` c
JOIN `target.orders` o
ON c.customer_id = o.customer_id
JOIN `target.order_items` i
ON o.order_id = i.order_id
GROUP BY c.customer_state
ORDER BY total_freight_val DESC,avg_freight_val DESC;
```



Row	customer_state	total_freight_val	avg_freight_val
1	SP	718723.07	15.15
2	RJ	305589.31	20.96
3	MG	270853.46	20.63
4	RS	135522.74	21.74
5	PR	117851.68	20.53
6	BA	100156.68	26.36
7	SC	89660.26	21.47
8	PE	59449.66	32.92
9	GO	53114.98	22.77
10	DF	50625.5	21.04

Insights:

- SP accounts for the highest share of total order freight to date, followed by Rio de Janeiro and Minas Gerais.

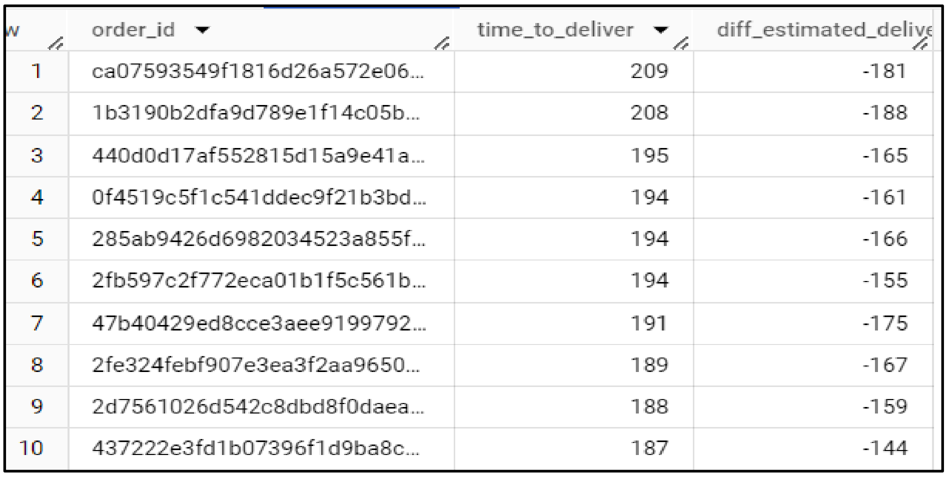
Recommendations:

- Optimize shipping logistics and negotiate better freight rates in SP to maximize cost efficiency, while focusing on streamlining delivery operations in Rio de Janeiro and Minas Gerais to reduce overall freight expenses.

5.1. 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.

QUERY:

```
SELECT order_id,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) AS
time_to_deliver,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY)
AS diff_estimated_delivery
FROM `target.orders`
WHERE order_status='delivered'
ORDER BY time_to_deliver DESC;
```



w	order_id	time_to_deliver	diff_estimated_delive
1	ca07593549f1816d26a572e06...	209	-181
2	1b3190b2dfa9d789e1f14c05b...	208	-188
3	440d0d17af552815d15a9e41a...	195	-165
4	0f4519c5f1c541ddec9f21b3bd...	194	-161
5	285ab9426d6982034523a855f...	194	-166
6	2fb597c2f772eca01b1f5c561b...	194	-155
7	47b40429ed8cce3aee9199792...	191	-175
8	2fe324febf907e3ea3f2aa9650...	189	-167
9	2d7561026d542c8dbd8f0daea...	188	-159
10	437222e3fd1b07396f1d9ba8c...	187	-144

Insights:

- Our data reveals a concerning trend of extended delivery times, with over 60 deliveries surpassing the 100-day mark.
- This suggests a significant potential bottleneck in our logistics process.

Recommendations:

- Perform a comprehensive review of our processes to identify inefficiencies in logistics operations, concentrating on areas such as order processing, inventory management, transportation, and delivery scheduling. Implement lean principles and automation wherever feasible to streamline workflows, minimize manual errors, and boost productivity.

- Create and integrate predictive models to forecast delivery times and identify potential delays. Set up automated alert systems to inform logistics teams and customers about expected delays, facilitating proactive mitigation strategies and improving communication. This approach will promote data-driven decision-making, enhance customer satisfaction, and reduce the chances of extended delivery times.

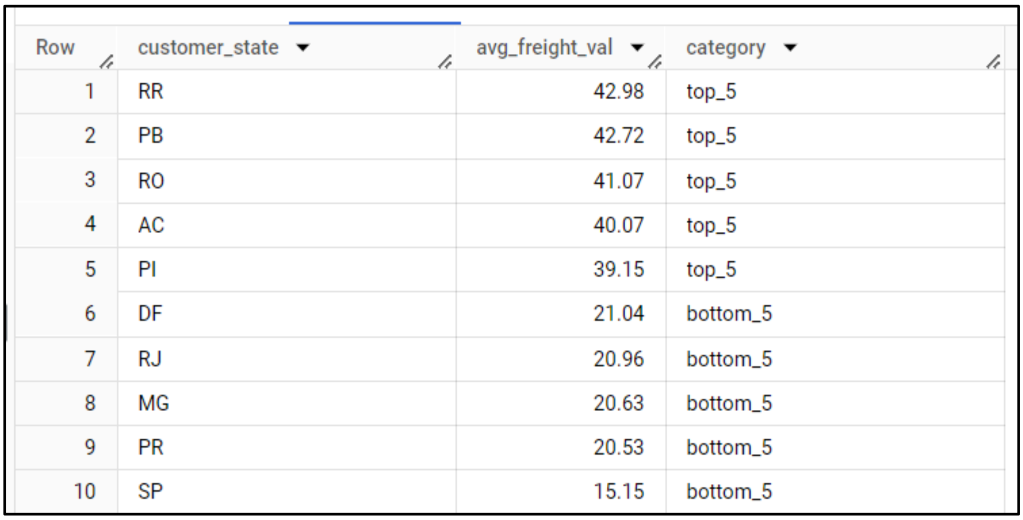
5.2. Find out the top 5 states with the highest & lowest average freight value.

QUERY:

```
WITH avg_freight AS(
  SELECT c.customer_state,
  ROUND(AVG(i.freight_value),2) AS avg_freight_val
  FROM `target.customers` c
  JOIN `target.orders` o
  ON c.customer_id = o.customer_id
  JOIN `target.order_items` i
  ON o.order_id = i.order_id
  GROUP BY c.customer_state
)
(SELECT
customer_state,avg_freight_val,'top_5' AS category
FROM avg_freight
ORDER BY avg_freight_val DESC
LIMIT 5)

UNION ALL

SELECT *
FROM
(SELECT
customer_state,avg_freight_val,'bottom_5' AS category
FROM avg_freight
ORDER BY avg_freight_val
LIMIT 5) t1 order by avg_freight_val desc;
```



Row	customer_state	avg_freight_val	category
1	RR	42.98	top_5
2	PB	42.72	top_5
3	RO	41.07	top_5
4	AC	40.07	top_5
5	PI	39.15	top_5
6	DF	21.04	bottom_5
7	RJ	20.96	bottom_5
8	MG	20.63	bottom_5
9	PR	20.53	bottom_5
10	SP	15.15	bottom_5

Insights:

- The states with the highest average freight value—Roraima, Paraíba, Rondônia, Acre, and Piauí—are mainly found in the North and Northeast regions of Brazil, reflecting a greater demand for freight services in these areas.
- Conversely, the states with the lowest average freight value—Distrito Federal (Brasília), Rio de Janeiro, Minas Gerais, Paraná, and São Paulo—are primarily located in the Southeast and South regions, which are more urbanized and have a stronger presence of service-oriented industries.

Recommendations:

- The e-commerce company should explore opportunities to expand its logistics and transportation services in the North and Northeast regions, possibly by investing in new warehouses, distribution centers, and transportation hubs.
- In the Southeast and South regions, the e-commerce company should prioritize the development of specialized logistics services, such as same-day delivery and supply chain optimization solutions, to enhance customer satisfaction and meet market demands.
- Additionally, conducting market research in these regions can help identify specific customer needs and preferences, allowing for more targeted service offerings.

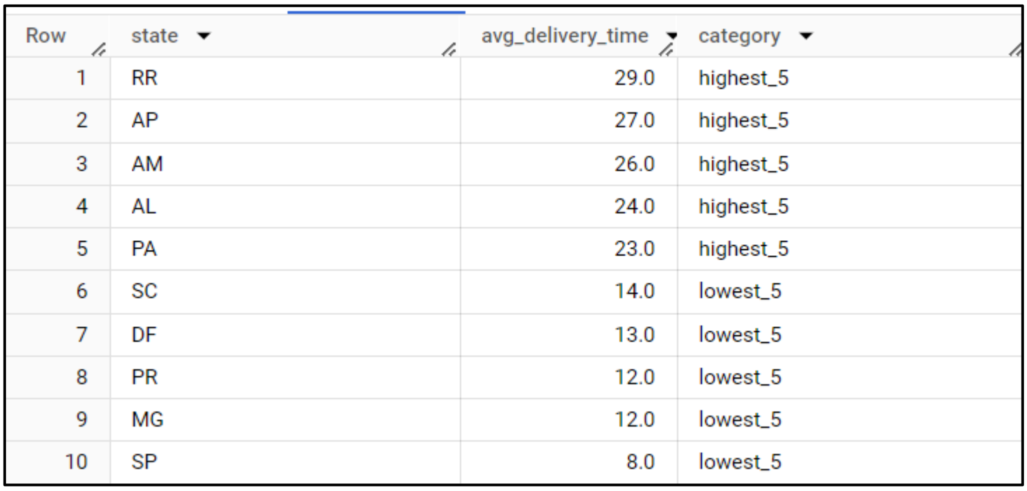
5.3. Find out the top 5 states with the highest & lowest average delivery time.

QUERY:

```
WITH CTE1 AS(
  SELECT o.order_id,c.customer_id,c.customer_state,
  DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,DAY)
AS time_to_deliver
  FROM `target.orders` o
  JOIN `target.customers` c
  ON o.customer_id = c.customer_id
  WHERE order_status = 'delivered'
),
Avg_delivery AS(
  SELECT customer_state ,
  ROUND(AVG(time_to_deliver)) AS avg_delivery_time
  FROM CTE1
  GROUP BY customer_state
)
(SELECT
customer_state AS state,avg_delivery_time,'highest_5' AS category
FROM Avg_delivery
ORDER BY Avg_delivery_time DESC
LIMIT 5)

UNION ALL

SELECT *
FROM
(SELECT
customer_state AS state,avg_delivery_time,'lowest_5' AS category
FROM Avg_delivery
ORDER BY Avg_delivery_time
LIMIT 5) t1 order by avg_delivery_time desc;
```



Row	state	avg_delivery_time	category
1	RR	29.0	highest_5
2	AP	27.0	highest_5
3	AM	26.0	highest_5
4	AL	24.0	highest_5
5	PA	23.0	highest_5
6	SC	14.0	lowest_5
7	DF	13.0	lowest_5
8	PR	12.0	lowest_5
9	MG	12.0	lowest_5
10	SP	8.0	lowest_5

Insights:

- The states with the highest average delivery days—Roraima, Amapá, Amazonas, Alagoas, and Pará—are mainly situated in the North region of Brazil, indicating longer delivery times in these areas, likely due to the extensive territory.
- In contrast, the states with the lowest average delivery days—Santa Catarina, Distrito Federal (Brasília), Paraná, Minas Gerais, and São Paulo—are primarily found in the South and Southeast regions, characterized by higher urbanization and population density.

Recommendations:

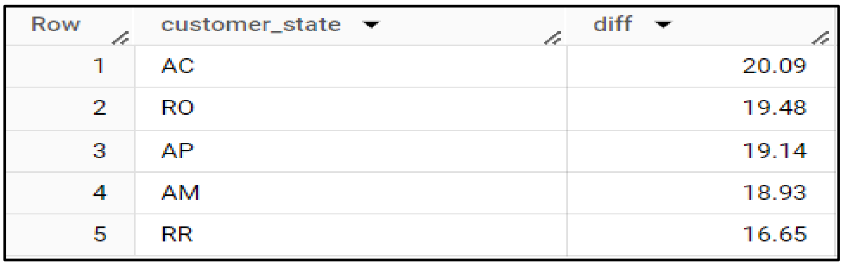
- The e-commerce company should explore the implementation of a regionalized logistics strategy that optimizes the delivery network specifically for the North region.
- This approach will allow the company to effectively tackle the unique delivery challenges faced in different regions, enhancing customer satisfaction.

- Additionally, the company should invest in targeted infrastructure improvements, such as local distribution centers, to further streamline operations in areas with longer delivery times.
- Conducting a thorough analysis of regional demographics and delivery patterns can provide valuable insights for tailoring services to meet customer needs more effectively.

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

Query:

```
WITH dates_avg AS(
  SELECT c.customer_state,
  ROUND(AVG(DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,DAY)),2) AS actual_delv_date,
  ROUND(AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_purchase_timestamp,DAY)),2) AS estimated_delv_date
  FROM `target.customers` c
  JOIN `target.orders` o
  ON c.customer_id = o.customer_id
  WHERE LOWER(o.order_status) = 'delivered'
  GROUP BY c.customer_state
)
SELECT
customer_state, ROUND(estimated_delv_date-actual_delv_date,2) AS diff
FROM dates_avg
ORDER BY diff DESC
LIMIT 5;
```



Row	customer_state	diff
1	AC	20.09
2	RO	19.48
3	AP	19.14
4	AM	18.93
5	RR	16.65

Insights:

- The top 5 states where order delivery is significantly faster than the estimated delivery date—Acre, Amazonas, Rondônia, Amapá, and Roraima—are all located in the North region of Brazil, indicating greater efficiency in logistics and delivery operations in these areas.

Recommendations:

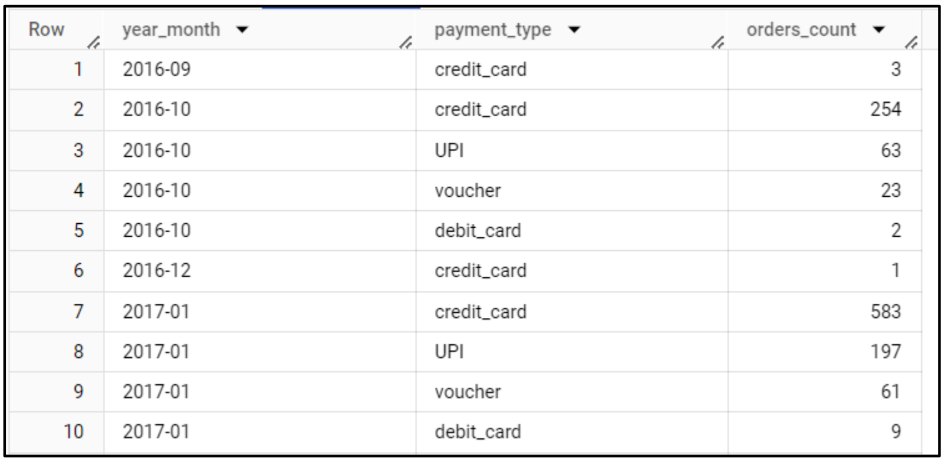
- The e-commerce company should consider introducing a “Fast Delivery” option exclusively for customers in these 5 states, ensuring delivery within a specific timeframe (e.g., 24-48 hours).
- This service could be offered at a premium price, allowing the company to capitalize on the efficient logistics capabilities in these regions while enhancing customer satisfaction.

- Marketing this fast delivery option could also attract new customers who prioritize speed and reliability in their online shopping experiences.

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

QUERY:

```
SELECT
format_datetime('%Y-%m',o.order_purchase_timestamp) AS year_month,
p.payment_type,
COUNT(o.order_id) AS orders_count
FROM target.orders o
JOIN target.payments p
ON o.order_id = p.order_id
GROUP BY year_month, p.payment_type
ORDER BY year_month, orders_count DESC;
```



Row	year_month	payment_type	orders_count
1	2016-09	credit_card	3
2	2016-10	credit_card	254
3	2016-10	UPI	63
4	2016-10	voucher	23
5	2016-10	debit_card	2
6	2016-12	credit_card	1
7	2017-01	credit_card	583
8	2017-01	UPI	197
9	2017-01	voucher	61
10	2017-01	debit_card	9

Insights:

- The data indicates that customers prioritize the convenience and security of credit card transactions, followed closely by the ease and speed of UPI payments.

Recommendations:

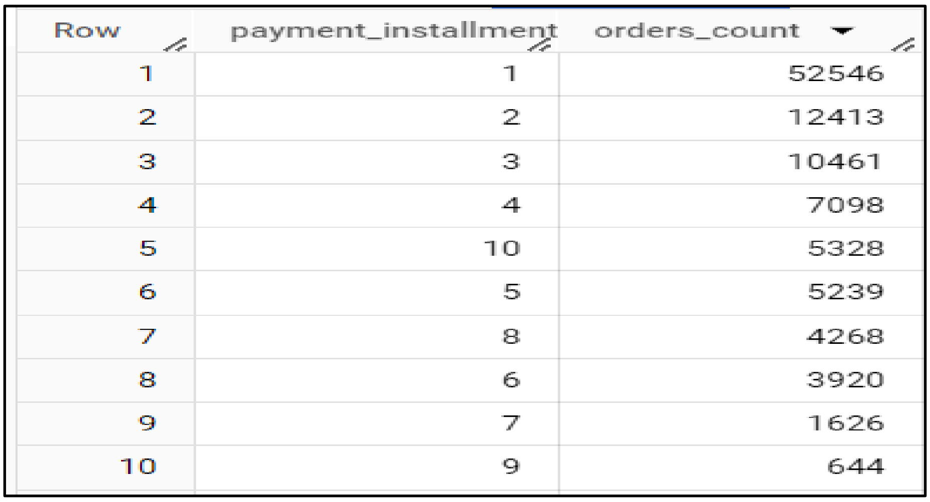
- The e-commerce company should consider providing exclusive benefits and rewards for customers who predominantly use credit cards, such as:
    - Cashback offers or discounts.
    - Early access to special sales and promotions.
    - Additional loyalty program points.
    - Free shipping or expedited delivery options.
- To enhance the UPI payment experience, the company should focus on:
    - Simplifying the UPI payment process to minimize friction.
    - Introducing UPI-specific promotions and discounts to incentivize usage.
    - Developing UPI-based loyalty programs to reward frequent users.
    - Ensuring seamless integration with popular UPI apps for a smoother transaction experience.

- Additionally, gathering customer feedback on payment preferences can help refine these strategies and better meet user expectations.

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

QUERY:

```
SELECT
p.payment_installments,COUNT(p.order_id) AS orders_count
FROM `target.orders` o
JOIN `target.payments` p
ON o.order_id=p.order_id
WHERE p.payment_installments>0
GROUP BY p.payment_installments
ORDER BY orders_count DESC;
```



Row	payment_installment	orders_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:

- The most popular payment option is 6 installments, with 5,239 orders, indicating a preference for moderate payment plans.
- Order counts significantly decrease as the number of installments increases, suggesting that customers are hesitant to commit to longer payment terms.

RECOMMENDATIONS:

- Promote the 6 and 7 installment plans more aggressively to encourage purchases.
- Consider offering incentives for higher installment options to attract more customers willing to spread their payments over longer periods.