HARSHAVARDHAN PATRA
Brazilian Retailer - Business case 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.

CODE:-

```
SELECT
  column_name,
  data_type
FROM
  `melodic-rig-463409-t8.Target_sql_scaller.INFORMATION_SCHEMA.COLUMNS`
WHERE
  table_name = 'customers';
```

RESULT:-

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

Insights.

- The table has 5 important columns that would help identify and locate customers.

- All identifiers (customer_id, customer_unique_id) are stored as STRING. That indicates these identifiers can be alphanumeric and are not meant for any numerical calculations.

- customer_zip_code_prefix, which is an INT64 type and can be used for geographical grouping or filtering.

- customer_city and customer_state being STRING enables region-wise analysis and segmentation.

RECOMMENDATION:- NA
ASSUMPTIONS:- NA

B. **Get the time range between which the orders were placed.**

```
SELECT
  MIN(order_purchase_timestamp) AS earliest_order,
  MAX(order_purchase_timestamp) AS latest_order
FROM
  `Target_sql_scaller.orders`;
```

RESULT:-

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

Insight.

- The dataset includes orders collected from September 4, 2016, to October 17, 2018, covering a little over 2 years of time.

Recommendation.

- We can undertake time-based analyze the trend (e.g., monthly or quarterly growth and seasonal patterns).
- Make sure that we have completeness on both the start and end month to avoid a biased interpretation of the trends.

Assumptions.

- All orders of this kind are recorded without missing an intermediate month.

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

CODE:-

```
SELECT
  c.customer_state,
  c.customer_city,
  COUNT(DISTINCT c.customer_id) AS number_of_customers
FROM
  `Target_sql_scaller.customers` c
JOIN
  `Target_sql_scaller.orders` o
ON
  c.customer_id = o.customer_id
WHERE
  o.order_purchase_timestamp BETWEEN TIMESTAMP('2016-09-04') AND
TIMESTAMP('2018-10-17')
GROUP BY
  c.customer_state,
  c.customer_city
ORDER BY
  number_of_customers DESC;
```

RESULT:-

Row	customer_state	customer_city	number_of_custo...
1	SP	sao paulo	15540
2	RJ	rio de janeiro	6882
3	MG	belo horizonte	2773
4	DF	brasilia	2131
5	PR	curitiba	1521
6	SP	campinas	1444
7	RS	porto alegre	1379
8	BA	salvador	1245
9	SP	guarulhos	1189
10	SP	sao bernardo do campo	938

Insights.

- SP cities contribute over 15,500 unique customers with Sao Paulo city alone accounting for most.
- Rio de Janeiro (RJ), Belo Horizonte (MG) and Brasília (DF) are other good cities.

- Most of the top 10 cities are from SP, RJ and MG, showing strong market presence in the Southeast region of Brazil.

Recommendations.

- We should give marketing and logistics priority in SP, RJ and MG due to more customers.
- Opening regional hubs (fulfillment centers) in or around Sao Paulo, Campinas and Rio de Janeiro could reduce delivery time and improve service efficiency.
- Take a good look into cities with increasing amount of customers. E.g. Curitiba, Salvador.

Assumptions.

- This paragraph informs that data includes only orders which are between Sept 4, 2016 to Oct 17, 2018. Each customer_id is considered to be a unique individual..

2.. In-depth Exploration:

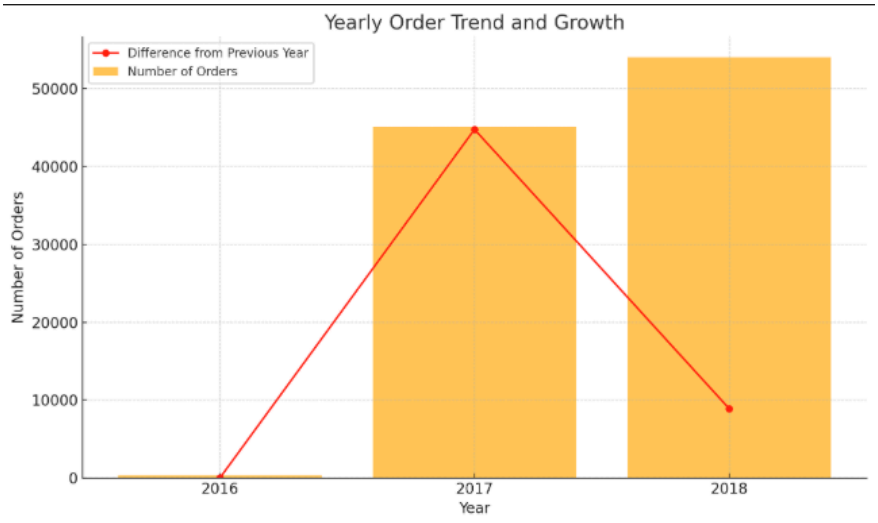
A. Is there a growing trend in the no. of orders placed over the past years?

```sql
WITH yearly_purchases AS (
  SELECT
    EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
    COUNT(order_id) AS number_of_orders
  FROM `Target_sql_scaller.orders`
  GROUP BY year
),
yearly_diff AS (
  SELECT *,
      COALESCE(number_of_orders - LAG(number_of_orders) OVER (ORDER BY year),0) AS difference_from_prev_year
  FROM yearly_purchases
)
SELECT * FROM yearly_diff
order by year;
```

Result :

Row	year	number_of_orders	difference_from_prev_year
1	2016	329	0
2	2017	45101	44772
3	2018	54011	8910

Insights:



Early Traction in 2016 Was Minimal

- With only 329 orders, 2016 was likely when Target launched in Brazil or they were in the testing phase.

Explosive Growth in 2017

- **Orders in 2016:** 329
- **Orders in 2017:** 45,101
- **Increase of 44,772 orders (over 13,600%)**

Sustained Growth in 2018

- **Orders in 2018:** 54,011
- **Growth from 2017:** +8,910 orders (~20% increase)
- While the percentage of growth slowed down compared to 2017, the number of orders has still increased significantly from the previous year , this indicates **stabilization and retention**.

RECOMMENDATIONS:

Target should Investigate the precise causes of this (e.g. new partnerships or aggressive marketing or platform improvements) So Target should reinvest into the same and replicate those strategies across different regions and customer segments.

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

```sql
SELECT  EXTRACT (year from order_purchase_timestamp) as year,
  EXTRACT(MONTH FROM order_purchase_timestamp) AS month_number,
  (FORMAT_DATE('%B', order_purchase_timestamp)) as month,
  count(order_id) as number_of_orders from `Target_sql_scaller.orders`
group by month, month_number,year
order by  year, month_number;
```

Row	year	month_number	month	number_of_orders
1	2016	9	September	4
2	2016	10	October	324
3	2016	12	December	1
4	2017	1	January	800
5	2017	2	February	1780
6	2017	3	March	2682
7	2017	4	April	2404
8	2017	5	May	3700
9	2017	6	June	3245
10	2017	7	July	4026
11	2017	8	August	4331

Insight: Between 2016 and 2018, Target's order volume in Brazil had seen an uptrend.

- **Seasonality Trend:** From 2017 to 2018, order volumes steadily increased from January through August. January, March, and August 2018 showed strong volumes, suggesting growing user engagement during early and mid-year. November 2017 and January 2018 stands out as the highest month in the dataset (7544 orders)&(7269 orders) respectively, potentially linked to promotions or year-end and year-start sales.
- **Notable Observation:**The drop in orders in September 2018 (16 orders) suggests a lack of quality as compared to 2017 (4285 orders). There appears to be a problem with the data quality or functionality.

ASSUMPTIONS: It is assumed that the Monthly data that is captured, is captured completely and consistently.

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 (CASE

  when time_stamp between '00:00:00' and '06:59:59' then 'Dawn'

  when time_stamp between '07:00:00' and '12:59:59' then 'Morning'

  when time_stamp between '13:00:00' and '18:59:59' then 'Afternoon'

  else 'night'

  end) as timezones, count(order_id) as orders_placed

 from (

  SELECT

    EXTRACT(TIME from order_purchase_timestamp ) as time_stamp,order_id

    from `Target_sql_scaller.orders`

  )

group by timezones

order by orders_placed DESC;
```

RESULT:-

Row	timezones	orders_placed
1	Afternoon	38135
2	night	28331
3	Morning	27733
4	Dawn	5242

Insights :-
Based on the data retrieved, the majority of Brazilian customers place their orders in the **afternoon (13:00–18:59)**, followed by night (19:00–23:59).
Mornings and dawn periods show significantly lower engagement, indicating a clear preference for shopping later in the day.

Recommendations :-
It's better to launch marketing campaigns, promotional push notifications, and in-app engagement in afternoon and evening hour. It will be able to get better visibility and conversion. To give quick resolution and enhance the level of satisfaction of the customer, the support must be in operations during the busy hours.

Assumptions:-
It is assumed that the `order_purchase_timestamp` column captures the time of order in **Brazilian local time**. If the data was stored in UTC and not converted, time-of-day classification may not be accurate.

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

A. **Get the month on month no. of orders placed in each state.**

```
SELECT
  EXTRACT(YEAR FROM o.order_purchase_timestamp) AS year,
  EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
  c.customer_state,
  COUNT(o.order_id) AS number_of_orders
FROM `Target_sql_scaller.orders` o
JOIN `Target_sql_scaller.customers` c
  ON o.customer_id = c.customer_id
GROUP BY year, month, c.customer_state
ORDER BY year, month, c.customer_state;
```

RESULT:-

Row	year	month	customer_state	number_of_orders
1	2016	9	RR	1
2	2016	9	RS	1
3	2016	9	SP	2
4	2016	10	AL	2
5	2016	10	BA	4
6	2016	10	CE	8
7	2016	10	DF	6
8	2016	10	ES	4
9	2016	10	GO	9
10	2016	10	MA	4
11	2016	10	MG	40
12	2016	10	MT	3
13	2016	10	PA	4

Insights :-

In the month (MG) and (SP) were constantly engaged signalling strong regional brand presence.

The peaks observed in November and December 2017 and early in 2018 may be due to holiday campaigns, black Friday, or the end of the year bonus which cause a more aggressive spending behavior by Brazilian consumers.

States such as (AM) and (AP) demonstrated action only at later stages, hinting to logistical or digital divide getting bridged.

Recommendations :-
To enhance overall measures, including digital ads, loyalty programs, and faster shipping in MG, SP, BA, and RJ, as the customer base is not only larger, but also on repeat purchase.
Regions like RR, AC, and AP experienced consistently low order volumes. This region may benefit from a partnership with local delivery players along with first-order discounts and targeted influencer initiatives.

Assumptions:-
Unless stated otherwise in the dataset, seasonal assumptions (such as spikes on Black Friday) take into consideration Brazilian general retail behavior.

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

```sql
SELECT
  customer_state,
  COUNT(DISTINCT customer_id) AS number_of_customers
FROM `Target_sql_scaller.customers`
GROUP BY customer_state
ORDER BY number_of_customers DESC;
```

RESULT:-

Row	customer_state	number_of_custo...
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 :-

- **High-Density Customer Zones.**
  The number of SP's customers, 41,746, is greater than three times that of the next state with the most customers.
  RJ and MG come next with their number of customers at 12,852 and 11,635 respectively.
  SP, RJ, MG account for 60% plus of total customers. Thus, these can be considered our focus regions.
- **Mid-Tier Engagement States:-**
  User engagement in RS, PR and SC is quite stable – 5,466; 5,045; 3,637
  DF (2,140) is less extensive, suggesting that it has high digital adoption as a result of urban infrastructure.
- **Low-Penetration Regions.**
  A number of states such as RR (46), AP (68), and AC (81) have very few users which may suggest lack of coverage, digital access, or platform awareness.
  States like AM (148), TO (280) and RO (253) also come in the low category.

Recommendations:-

.

- **Maximize Returns from High-Volume States.**
  Retention campaign, delivery optimisation & upselling in SP, RJ and MG should be the focus to keep the existing base intact.
- **Expand Presence in Mid-Tier Regions.**
  Make use of RS, PR, SC, and DF by offering location-based deals, quicker shipping options or local partnerships.
- **Targeted Outreach for Low-Engagement States.**

You can conduct surveys in RR, AP, AC, AM, and TO to find out the reasons for low turnout.
Work with delivery service and community orgs to improve visibility and increase accessibility.

Assumptions :-.
The dataset shows total historical or current users with no duplicates or truncation.

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).**

CODE:-

```
WITH order_ts AS (
  SELECT
    order_id,
    EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
    EXTRACT(MONTH FROM order_purchase_timestamp) AS month
  FROM `Target_sql_scaller.orders`
  GROUP BY order_id, year, month
  HAVING (year BETWEEN 2017 AND 2018) AND (month BETWEEN 1 AND 8)
),
payment_info AS (
  SELECT
    order_id,
    payment_value
  FROM `Target_sql_scaller.payments`
),
yearly_costs AS (
```

```
  SELECT
    o.year,
    ROUND(SUM(pi.payment_value), 2) AS cost_of_order_per_year
  FROM order_ts o
  LEFT JOIN payment_info pi ON o.order_id = pi.order_id
  GROUP BY o.year
)
SELECT
  year,
  cost_of_order_per_year,
  ROUND(((cost_of_order_per_year - LAG(cost_of_order_per_year) OVER (ORDER BY year))
         / LAG(cost_of_order_per_year) OVER (ORDER BY year)) * 100, 2) AS
percent_increase
FROM yearly_costs
ORDER BY year;
```

RESULT:-

Row	year	cost_of_order_pe...	percent_increase
1	2017	3669022.12	*null*
2	2018	8694733.84	136.98

Insights:

- Orders placed between January and August 2017 resulted in total costs of about 3.60 million BRL.
- For the same period of 2018, the total invoice for orders reached almost 8.69 million BRL.
- The orders have risen in cost by 136.98% from what they used to be.

This state that the level of customer engagement and the volume of transactions that the platform managed to achieve in the first eight months of 2018 were phenomenal given this steep increase in merely one year.

Recommendations:

- Keep the momentum going by executing the successful marketing strategies that were adopted in early 2018. This will ensure sustained growth.
- Further, explore order segmentation to find out whether the growth is from customers or the regional market.
    - Higher number of orders, or
    - Higher average order values per transaction.
- Find out the states like SP, RJ contributing a lot to this growth and double down on them.

ASSUMPTIONS :-

The `payment_value` is assumed to reflect the actual monetary value of the orders paid, without returns or partial payments.

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

```
SELECT
  c.customer_state,
  ROUND(SUM(oi.price),0) as Total_value_of_price_per_state,
  ROUND(AVG(oi.price),0) as Average_value_of_price_per_state
  from `Target_sql_scaller.order_items` oi
     JOIN `Target_sql_scaller.orders` o  on oi.order_id = o.order_id
     JOIN  `Target_sql_scaller.customers` c on o.customer_id = c.customer_id
GROUP BY customer_state
ORDER BY customer_state;
```

Result:-

Row	customer_state	Total_value_of_pr...	Average_value_of...
1	AC	15983.0	174.0
2	AL	80315.0	181.0
3	AM	22357.0	135.0
4	AP	13474.0	164.0
5	BA	511350.0	135.0
6	CE	227255.0	154.0
7	DF	302604.0	126.0
8	ES	275037.0	122.0
9	GO	294592.0	126.0
10	MA	119648.0	145.0

Insights.

- In total order value, (SP) currently has a sizeable advantage relative to II and MD, presently standing at 5,202,955 BRL for (SP). This may indicate significant buying power and/or large customer base. This suggest that on the other hand order volume is high and purchases per order are on the small-size so average order value is low (110 BRL).

- RJ and MG have high total order values of 1,824,093 BRL and 1,585,308, confirming their importance in terms of contribution. Despite this, their average order values (RJ – 125 BRL, MG – 121 BRL) are on the lower side, indicating volume.

- In Paraíba (PB), the average order value amounts to 191 BRL, while the average order value in Alagoas (AL) is 181 BRL. Although Paraíba (PB) and Alagoas (AL) have lower total values, the average order value shows the purchase of high-value or luxury items in each order.

- Passive voice of the sentence: Small total value and small volume of states like (RR) and (AP) show these states are not used for any product.

Recommendations.

- States like (PB), (AL) and (PA) having very high average order value can be targeted for some premium or bundle offering as the customer may be willing to spend more on a single order.

- In states such as SP, RJ, and MG, the average order value can be focused at increasing via upsell, combo deals and minimum cart value offer. This will boost revenue without solely depending on volume.

- States like (RR), (AC) and (AP) having low total order value as well as low average order value will get plans. Low access states will work on proposals like an incentive for first purchase, tie-up with regional players and further improvement in logistics.

Assumptions:-

It is presumed that the order values must be in BRL (Brazilian Real).

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

```
SELECT
  c.customer_state,
  ROUND(SUM(o_i.freight_value),0) as Total_freight_value,
  ROUND(AVG(o_i.freight_value),0) as Average_freight_value
  from `Target_sql_scaller.order_items` o_i
     JOIN `Target_sql_scaller.orders` o  on o_i.order_id = o.order_id
     JOIN  `Target_sql_scaller.customers` c on o.customer_id = c.customer_id
GROUP BY customer_state
ORDER BY customer_state;
```

Result:-

Row	customer_state	Total_freight_value	Average_freight_...
1	AC	3687.0	40.0
2	AL	15915.0	36.0
3	AM	5479.0	33.0
4	AP	2788.0	34.0
5	BA	100157.0	26.0
6	CE	48352.0	33.0
7	DF	50625.0	21.0
8	ES	49765.0	22.0
9	GO	53115.0	23.0
10	MA	31524.0	38.0

Insights.

- (SP)'s total freight value is the greatest (718K BRL), but it has the lowest average freight (15 BRL). This could indicate that there is logistics efficiency or local warehousing at the shipment origin.
- In states like RJ and PR, the total freight is high whilst average costs are relatively low (~21 BRL).
- PB and RR are on the top of the freight averages who got 43 BRL with very low orders. The reason behind that can be remote location or inefficiency in delivery, non-elimination of other possible solutions.
- The shipping costs for (RO) and (AC) are also above the mean which might be caused with limited infrastructure.

Recommendations.

- Make it possible to provide freight discounts or incentives in high-cost states such as (PB), (RR), and (RO)
- Explore local delivery partnerships or micro-hubs in the North/Northeast.
- Maintain current logistics strategies in (SP), (RJ), and (MG) for continued cost efficiency.

Assumptions:

- Freight values are in BRL and fully inclusive.
- Timeframe is uniform across states.

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.

CODE:-

```
SELECT
 order_id,
  DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, day) as
delivery_time,
  DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date, day)
as delivery_difference,
  (CASE
        WHEN
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date, day) > 1
then 'Fast delivery'
        WHEN
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date, day) = 0
then 'On time delivery'
      else 'Late delivery'
    END
  ) as delivery_type
from `Target_sql_scaller.orders`
WHERE order_delivered_customer_date is not null;
```

RESULT :-

Row	order_id	delivery_time	delivery_difference	delivery_type
1	770d331c84e5b214bd9dc70a1...	7	45	Fast delivery
2	dabf2b0e35b423f94618bf965fc...	7	44	Fast delivery
3	8beb59392e21af5eb9547ae1a9...	10	41	Fast delivery
4	1950d777989f6a877539f53795...	30	-12	Late delivery
5	bfbd0f9bdef84302105ad712db...	54	-36	Late delivery
6	cd3b8574c82b42fc8129f6d502...	10	39	Fast delivery
7	31b0dd6152d2e471443debf03...	8	40	Fast delivery
8	a8c3f65a43b2e956cbfca10db4...	15	34	Fast delivery
9	a041155864e51411164582913...	9	41	Fast delivery
10	50013835d7b14aefb45282586...	17	33	Fast delivery

Insights:

- Most orders were delivered significantly faster than the estimated time—indicated by a positive delivery difference, classified as "Fast delivery."

- A smaller portion of deliveries had negative delivery differences, indicating they arrived later than promised.

Recommendations:

- Highlight the fast delivery advantage in marketing efforts to build customer trust and satisfaction.
- For late deliveries, perform root cause analysis (e.g., vendor issues, regional logistics delays) and improve service levels.

Assumptions:

- Estimated delivery dates are considered accurate as per the business logic.

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

```sql
WITH freight_by_state AS (
  SELECT
    c.customer_state,
    ROUND(AVG(o_i.freight_value), 2) AS avg_freight_value
  FROM `Target_sql_scaller.order_items` o_i
  JOIN `Target_sql_scaller.orders` o ON o_i.order_id = o.order_id
  JOIN `Target_sql_scaller.customers` c ON o.customer_id = c.customer_id
  GROUP BY c.customer_state
),
top5 AS (
  SELECT
    customer_state AS top_5_state,
    avg_freight_value AS top_5_avg_freight,
    ROW_NUMBER() OVER (ORDER BY avg_freight_value DESC) AS rn
  FROM freight_by_state
  LIMIT 5
),
bottom5 AS (
  SELECT
    customer_state AS bottom_5_state,
    avg_freight_value AS bottom_5_avg_freight,
    ROW_NUMBER() OVER (ORDER BY avg_freight_value ASC) AS rn
  FROM freight_by_state
  LIMIT 5
)
SELECT
  t.top_5_state, t.top_5_avg_freight,
  b.bottom_5_state, b.bottom_5_avg_freight
FROM top5 t
FULL OUTER JOIN bottom5 b ON t.rn = b.rn
ORDER BY t.rn;
```

Result:-

Row	top_5_state	top_5_avg_freight	bottom_5_state	bottom_5_avg_fr...
1	RR	42.98	SP	15.15
2	PB	42.72	PR	20.53
3	RO	41.07	MG	20.63
4	AC	40.07	RJ	20.96
5	PI	39.15	DF	21.04

Insights.

- States like RR, PB , RO have the highest average freight values which hint there could be more logistical demerits or remote delivery areas.
- The average freight costs of SP, PR, and MG are low in contrast suggesting efficient delivery infrastructure and dense customer clusters.

Recommendations.

- Bring together the regional carriers or set up warehouse hubs to help with high-costing states (RR, PB) logistics.
- Replicating of best practices (e.g. routing, packaging, vendor selection) from low cost states is to be done using insights.

Assumptions.

- The freight is covered by the actual cost of delivery.
- The locations of customers are evenly distributed unless otherwise noted.

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

```
WITH delivery_time as(
  SELECT
    c.customer_state as state,

ROUND(AVG(DATE_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,
day))) as
       avg_delivery_time
  from `Target_sql_scaller.customers` c RIGHT JOIN `Target_sql_scaller.orders`
o on c.customer_id = o.customer_id
  group by customer_state
  ),
top5_avg_del_time as (
  SELECT
    state as top5_states,
    avg_delivery_time as top5_avg_delivery_time,
    ROW_NUMBER() over(order by avg_delivery_time desc) as rn
  from delivery_time
  LIMIT 5
),
bottom_5_avg_del_time as (
```

```
  SELECT
    state as bottom5_states,
    avg_delivery_time as bottom5_avg_delivery_time,
    ROW_NUMBER() over(order by avg_delivery_time Asc) as rn
  from delivery_time
  LIMIT 5
)
SELECT top5_states,top5_avg_delivery_time,
  bottom5_states,bottom5_avg_delivery_time
FROM top5_avg_del_time tp FULL OUTER JOIN
  bottom_5_avg_del_time bt on tp.rn = bt.rn
order by tp.rn;
```

RESULT:-

Row	top5_states	top5_avg_deliver...	bottom5_states	bottom5_avg_deli...
1	RR	29.0	SP	8.0
2	AP	27.0	MG	12.0
3	AM	26.0	PR	12.0
4	AL	24.0	DF	13.0
5	PA	23.0	SC	14.0

Insights.

- The states of RR (29 days), AP (27), and AM (26) take an unusually long time on account probably of geographical remoteness or gaps in infrastructure.
- On the contrary, SP has an operational efficiency of 8 days and MG/PR/DF has a 12 day operational efficiency in the area of fulfilment and logistics.

Recommendations.

- Investing in local fulfillment centers or micro-warehousing in low-performing areas (e.g., RR, AP) should be done.
- Mimic supply chain models drawn from states of SP and MG in states with high delays to help reduce the gap.

Assumptions.

The delivery time is counted from the purchase date to the delivery date, thus it is assumed that the timestamps are logged correctly.

We do not accommodate external factors such as weather or regional holidays (e.g. local lockdowns, state issues).

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

```
WITH del_diff as (
SELECT
  customer_id,
  DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date, day) as
diff_delivery
from `Target_sql_scaller.orders`
  where order_delivered_customer_date is not null
)
SELECT
   c.customer_state, ROUND(AVG(d.diff_delivery),0) as avg_delivery_diff
from del_diff d join `Target_sql_scaller.customers` c
on d.customer_id = c.customer_id
group by customer_state
order by avg_delivery_diff DESC
LIMIT 5;
```

Insights.

- States such as AC, RO, AP, AM, and RR always made the delivery up to 20 days before the estimated date.
- This means there's more buffer time in estimated delivery or last mile logistics is exceptional in these areas.

Recommendation :-

- To increase customer confidence as well as conversion rates, highlight early delivery trends in marketing.

Assumptions:-

The predicted delivery dates are supposed to be system-generated, not user-generated.

6.. Analysis based on the payments:

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

CODE:-

```
select
  extract(year from o.order_purchase_timestamp) as year,
  extract(month from o.order_purchase_timestamp) as month,
    p.payment_type,
  count(distinct o.order_id) as no_of_orders
from `Target_sql_scaller.payments` p
join `Target_sql_scaller.orders` o on p.order_id=o.order_id
group by year,month, p.payment_type
order by year;
```

RESULT:-

Row	year	month	payment_type	no_of_orders
1	2016	10	voucher	11
2	2016	10	credit_card	253
3	2016	12	credit_card	1
4	2016	10	UPI	63
5	2016	10	debit_card	2
6	2016	9	credit_card	3
7	2017	4	voucher	115
8	2017	10	voucher	208
9	2017	6	voucher	142
10	2017	5	voucher	171

Insights.
Credit cards have been the most used payment method since October 2016.
UPI usage was already on-boarded in 2016, indicating that digital-savvy users used it very quickly.

The vouchers and debit card were hardly used throughout the months indicating lower acceptability.

Recommendations.
Through cashback of discounts and EMI offers, encourage UPI and card usage. The same will assist in retaining and growing the user's preference.

We may need to communicate the reasons for low usage of vouchers and debit cards. This could be due to contractual restrictions on use or lack of awareness. Further, the UX may need to be improved.

Assumptions.
All payment types are always recorded with no breaks and misclassifications.

The dataset includes only successful payments; refund, failed orders are excluded from the data.

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

CODE:-

```
select
  payment_installments,
  count(distinct p.order_id) as no_of_orders
from `Target_sql_scaller.payments` p
join `Target_sql_scaller.orders` o ON p.order_id = o.order_id
where p.payment_installments >=1
group by payment_installments
order by payment_installments;
```

RESULT:-

Row	payment_installm...	no_of_orders
1	1	49060
2	2	12389
3	3	10443
4	4	7088
5	5	5234
6	6	3916
7	7	1623
8	8	4253
9	9	644
10	10	5315

Insights.

- Most customers prefer full payments as orders for single installments payment stand at 49060.

- After 2 or 3 installments, drop-off is sharp, with very few users opting for extended plans beyond 6 months.
- Together the installment plans for above 5 months account for a very small percentage.

Recommendations.

- The marketing and discounts should just focus on 1–3 instalments as the customer prefers this.
- On 6+ installments, encourage the purchase of bigger-ticket items or try to offer low-interest EMIs.

Assumptions.

- Every order_id is unique for counting, and installment data refers to completed payments only.
- Any installment value greater than or equal to 1 is deemed valid as it indicates the customer's choice. It is not deemed invalid because it is just a default value or a test entry.