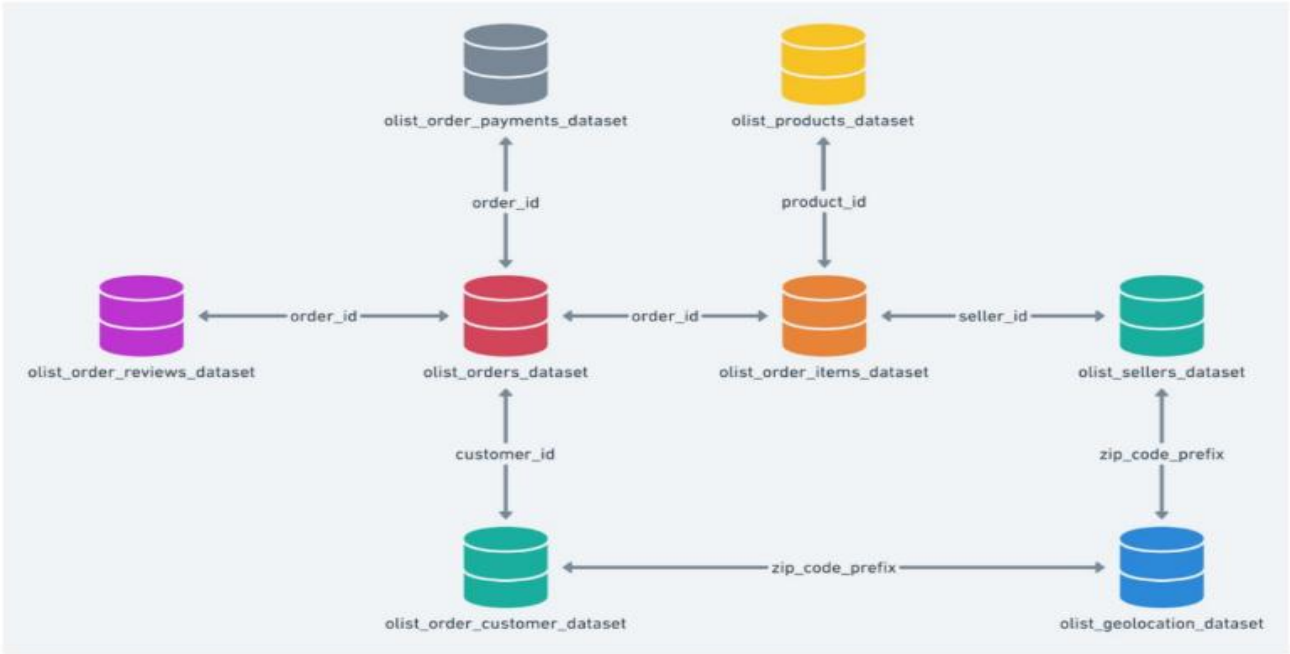
TARGET- BUSINESS CASE STUDY

Company Introduction

- ✓ Target is a globally renowned brand and a prominent retailer in the United States. Target makes itself a preferred shopping destination by offering outstanding value, inspiration, innovation and an exceptional guest experience that no other retailer can deliver.
- ✓ This particular business study focuses on the operations of Target in Brazil and provides insightful information about 100,000 orders placed between 2016 and 2018. The dataset offers a comprehensive view of various dimensions including the order status, price, payment and freight performance, customer location, product attributes, and customer reviews.
- ✓ In this study, the given dataset has been analysed to extract valuable insights from various aspects of the business and to suggest actionable recommendations. BiqQuery was used for writing SQL queries and fetching data from dataset. Tableau and Excel were used for creating visualisations.

1. Exploratory Data Analysis

- ➢ **Dataset Schema**:



- **Structure of the dataset**:

  The dataset contains 8 tables as mentioned above –

  1. customers
  2. sellers
  3. order_items
  4. geolocation
  5. payments
  6. reviews
  7. orders
  8. products

a) Data type of all columns in the "customers" table.

Query-

```
SELECT column_name, data_type
FROM Target.INFORMATION_SCHEMA.COLUMNS
WHERE table_name = 'Customers';
```

Output-

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

Insight-

All column data types were correct with no mismatches found.

b) Time Range between which the orders were place

Query-

```
SELECT
   MIN(DATE(order_purchase_timestamp)) AS First_Order_Date,
   MAX(DATE(order_purchase_timestamp)) AS Latest_Order_Date,
   DATE_DIFF(MAX(DATE(order_purchase_timestamp)),
MIN(DATE(order_purchase_timestamp)), MONTH)
   AS Time_Range_Months
FROM `Target.Orders`;
```

Output-

Row	First_Order_Date	Latest_Order_Date	Time_Range_Months
1	2016-09-04	2018-10-17	25

Insight-

The dataset contains order records from 2016 to 2018, covering a total of 25 months

c) Count the Number of Unique Cities & States in Orders

Query-1

```
SELECT
   COUNT(DISTINCT customer_city) AS unique_cities,
   COUNT(DISTINCT customer_state) AS unique_states
FROM `Target.Customers`;
```

Output-

Row	unique_cities	unique_states
1	4119	27

Query-2

```
SELECT
   COUNT(DISTINCT geolocation_city) AS No_of_cities,
   COUNT(DISTINCT geolocation_state) AS No_of_states
FROM `Target.geolocation`;
```

Output-

Row	No_of_cities	No_of_states
1	8011	27

Insights-

Dataset includes geolocation data for 8,011 cities and 27 states. However, the customers table shows that orders come from only 4,119 cities, meaning **about 50% (3,892 cities) have no customers, showing an opportunity for expansion.**

2. In-depth Exploration

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

Query-

```
SELECT
  EXTRACT(YEAR FROM order_purchase_timestamp) AS order_year,
  EXTRACT(MONTH FROM order_purchase_timestamp) AS order_month,
  COUNT(order_id) AS total_orders
FROM `Target.Orders`
GROUP BY order_year, order_month
ORDER BY order_year, order_month;
```

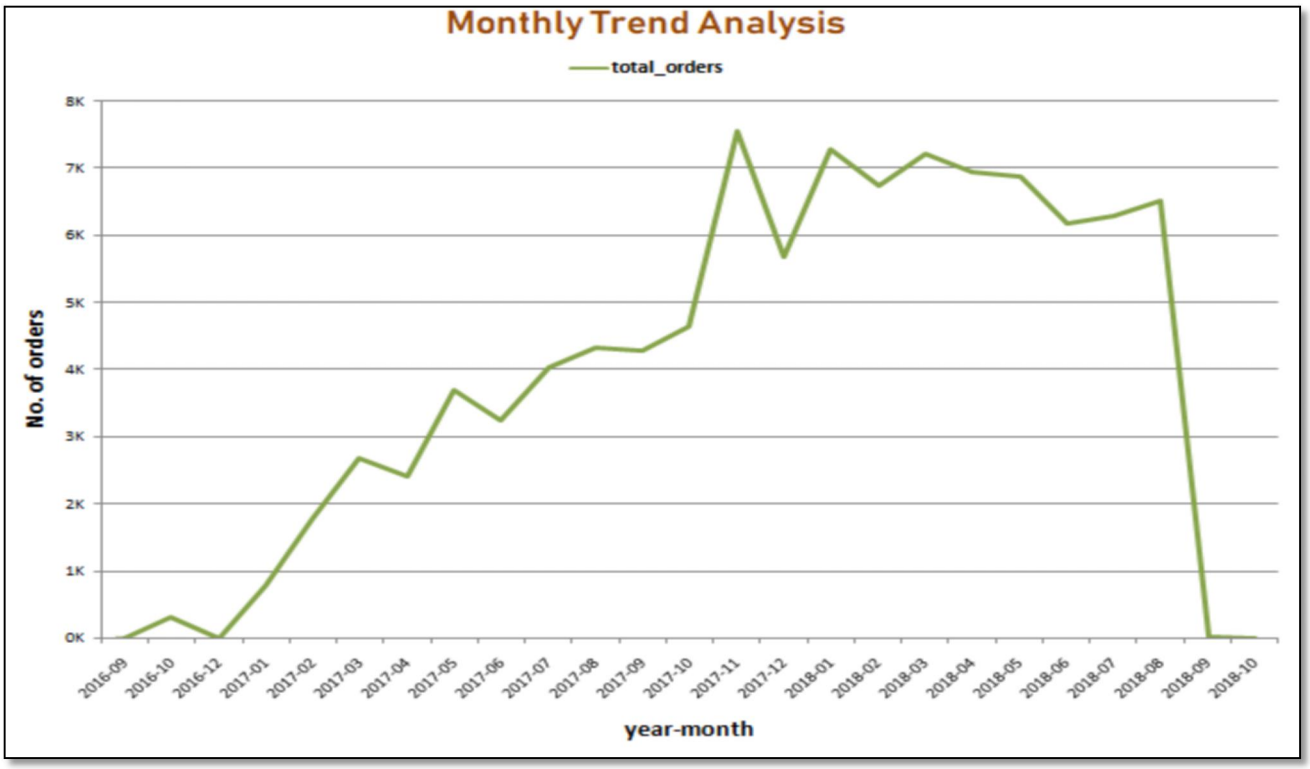
Output-

Row	order_year	order_month	total_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
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
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-

Analysis of the data shows that, after excluding cancelled and unavailable orders, the number of orders increased significantly from 2016 to 2017 before slowing down in 2018. Overall, the trend demonstrates a steady upward movement in order volume.

➢ **Visualisation-**



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

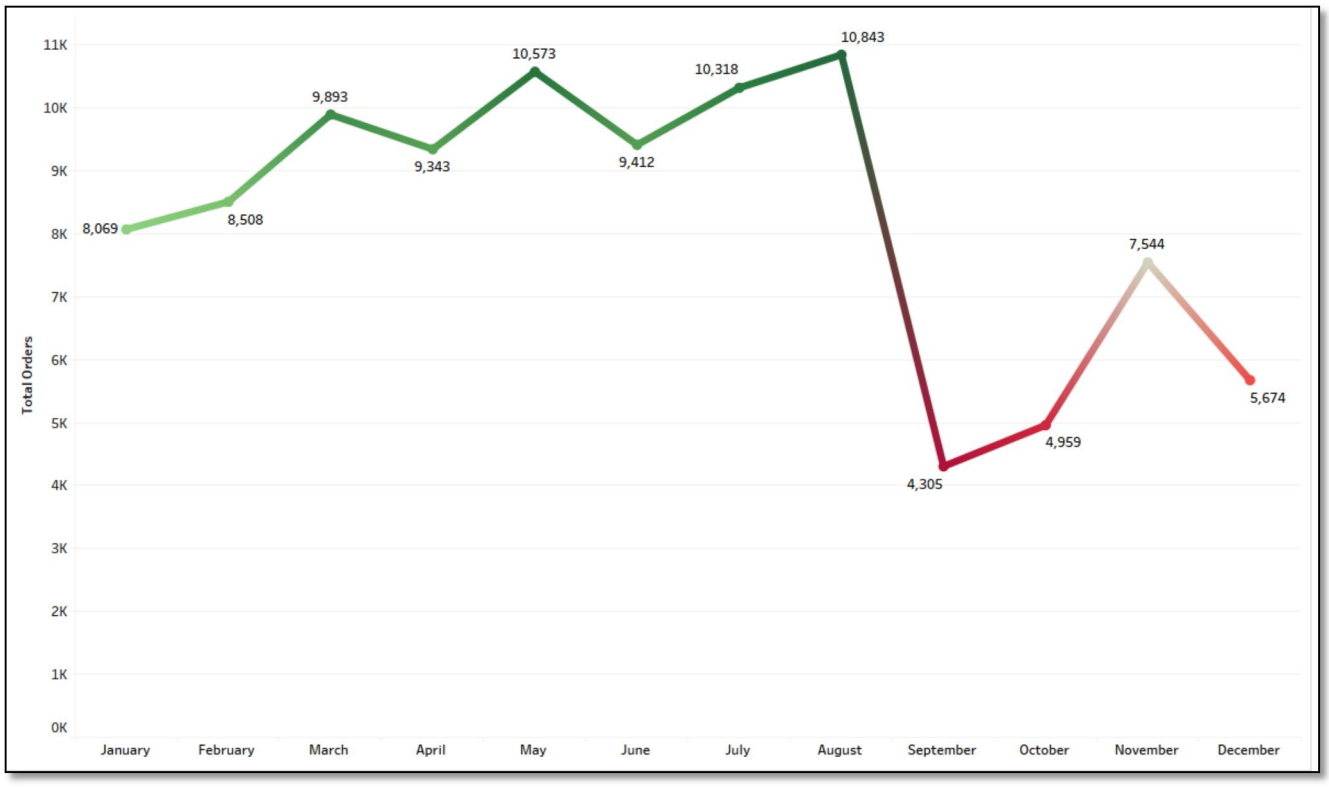
Query-

```
SELECT
  CASE EXTRACT(MONTH FROM order_purchase_timestamp)
     WHEN 1 THEN 'January'
     WHEN 2 THEN 'February'
     WHEN 3 THEN 'March'
     WHEN 4 THEN 'April'
     WHEN 5 THEN 'May'
     WHEN 6 THEN 'June'
     WHEN 7 THEN 'July'
     WHEN 8 THEN 'August'
     WHEN 9 THEN 'September'
     WHEN 10 THEN 'October'
     WHEN 11 THEN 'November'
     WHEN 12 THEN 'December'
  END AS order_month_name,
  COUNT(order_id) AS total_orders
FROM `Target.Orders`
GROUP BY order_month_name, EXTRACT(MONTH FROM order_purchase_timestamp)
ORDER BY EXTRACT(MONTH FROM order_purchase_timestamp);
```

Output-

Row	order_month_name	total_orders
1	January	8069
2	February	8508
3	March	9893
4	April	9343
5	May	10573
6	June	9412
7	July	10318
8	August	10843
9	September	4305
10	October	4959
11	November	7544
12	December	5674

➢ Visualisation-



MONTHLY SEASONALITY

Insights-

- The order count remains high during the first eight months of the year, with values above 8,000. However, it declines in the last four months.
- Orders begin to rise in March and continue increasing until August, which sees the highest number of orders in the year.
- **May, July**, and **August** record significantly **high order counts**, exceeding 10,000.
- The **lowest order counts** occur in **September** and **October**.

a) During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

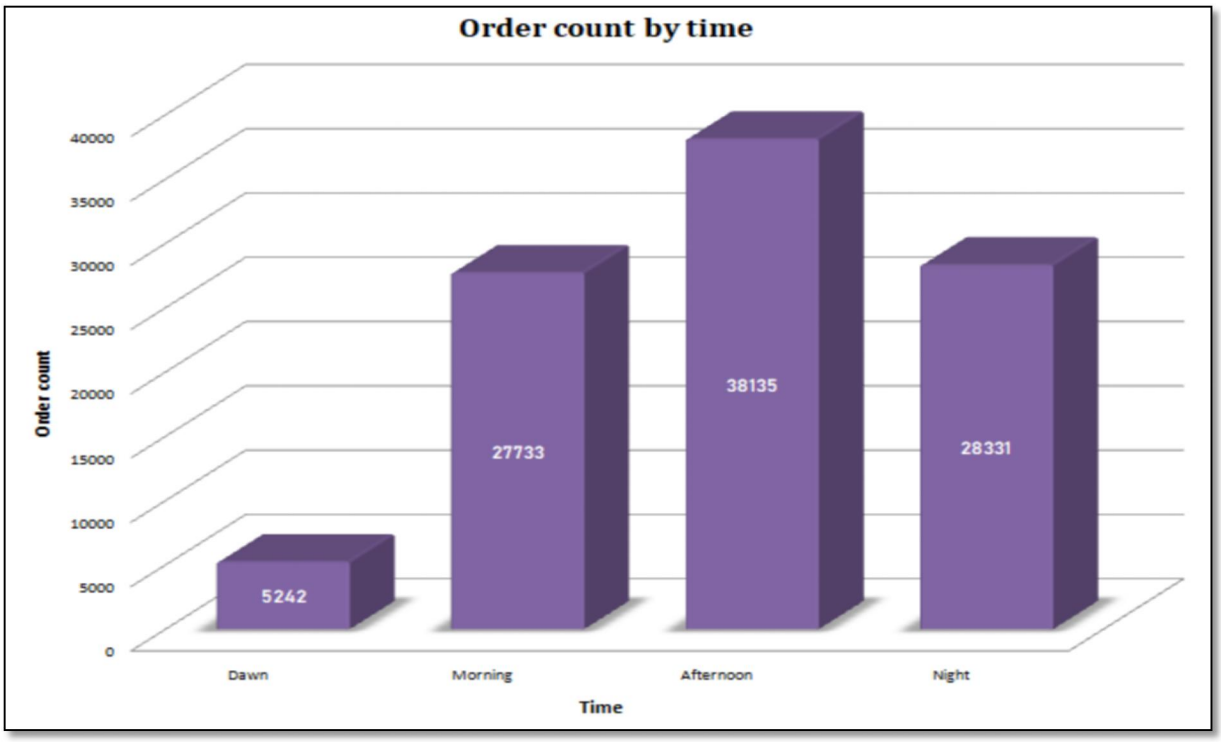
Query-

```
SELECT
  CASE
     WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 0 AND 6 THEN 'Dawn'
     WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 7 AND 12 THEN
'Morning'
     WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 13 AND 18 THEN
'Afternoon'
     ELSE 'Night'
  END AS time_of_day,
  COUNT(order_id) AS total_orders
FROM `Target.Orders`
GROUP BY time_of_day
ORDER BY total_orders DESC;
```

Output-

Row	time_of_day	total_orders
1	Afternoon	38135
2	Night	28331
3	Morning	27733
4	Dawn	5242

➢ **Visualisation-**



Insights-

Customers place the highest number of orders in the afternoon (13-18 hrs), followed by the night (19-23 hrs) and the morning (7-12 hrs). The lowest order activity occurs during dawn (0-6 hrs).

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

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

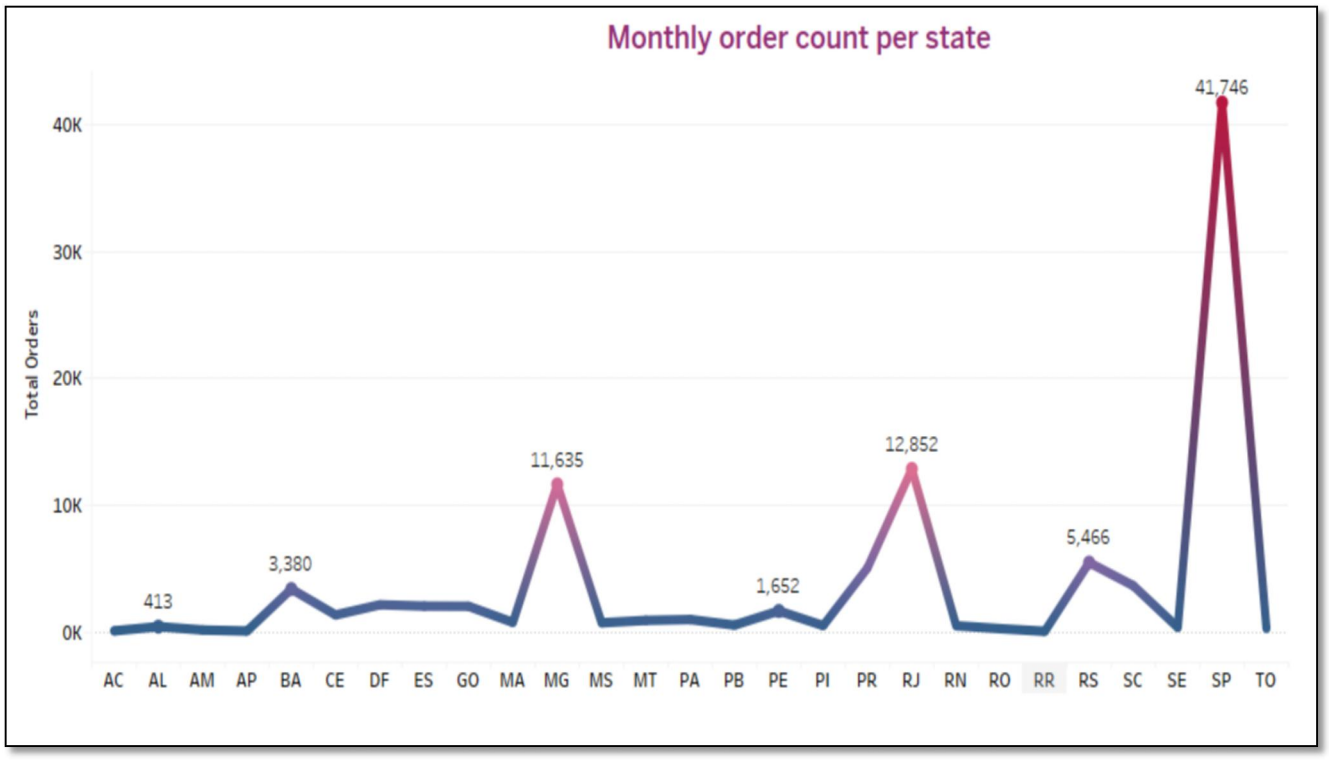
Query-

```
SELECT
   customer_state,
   EXTRACT(YEAR FROM order_purchase_timestamp) AS order_year,
   EXTRACT(MONTH FROM order_purchase_timestamp) AS order_month,
   COUNT(order_id) AS total_orders
FROM `Target.Orders` o
JOIN `Target.Customers` c
ON o.customer_id = c.customer_id
GROUP BY customer_state, order_year, order_month
ORDER BY customer_state, order_year, order_month;
```

Output(First 10rows)-

Row	customer_state	order_year	order_month	total_orders
1	AC	2017	1	2
2	AC	2017	2	3
3	AC	2017	3	2
4	AC	2017	4	5
5	AC	2017	5	8
6	AC	2017	6	4
7	AC	2017	7	5
8	AC	2017	8	4
9	AC	2017	9	5
10	AC	2017	10	6

➢ Visualisation-



Insights-

From the above analysis, **SP dominates e-commerce orders**, followed by **PR and MG**. The significant disparity in order volumes suggests **strong market presence in certain states** while others, like **RR and AC, show minimal activity**.

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

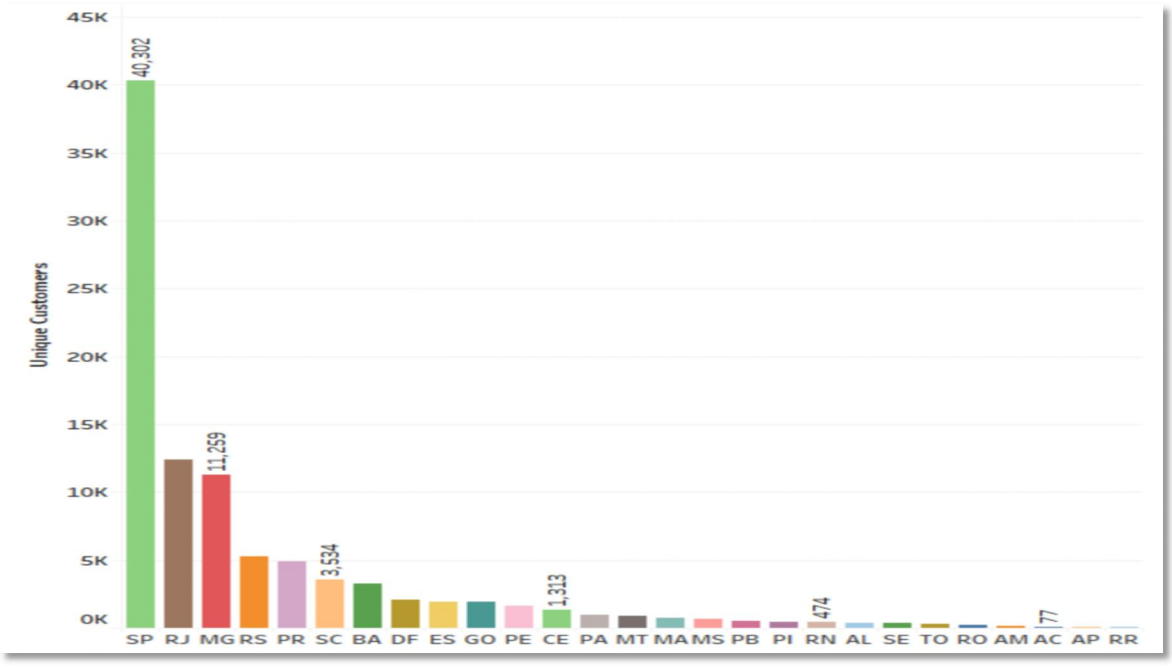
Query-

```
SELECT
   customer_state,
   COUNT(DISTINCT customer_unique_id) AS unique_customers
FROM `Target.Customers`
GROUP BY customer_state
ORDER BY unique_customers DESC;
```

Output(First 10rows)-

Row	customer_state	unique_customers
1	SP	40302
2	RJ	12384
3	MG	11259
4	RS	5277
5	PR	4882
6	SC	3534
7	BA	3277
8	DF	2075
9	ES	1964
10	GO	1952

➢ **Visualisation-**



Insights-

1. **Strong Presence in the Southeast & South**
   - SP has the highest number of customers (**40,302**), followed by **RJ and MG**.
   - The **Southern states** (RS, PR, SC) also have a notable customer base.
2. **Moderate Presence in Other States**
   - **BA, DF,ES , and GO** have a decent number of customers but much lower than SP.
3. **Low Presence in the North & Northeast**
   - States like **RR, AP, and AC** have very few customers, indicating low market reach.

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

a) The % increase in the cost of orders from 2017 to 2018(Months between Jan to Aug only).

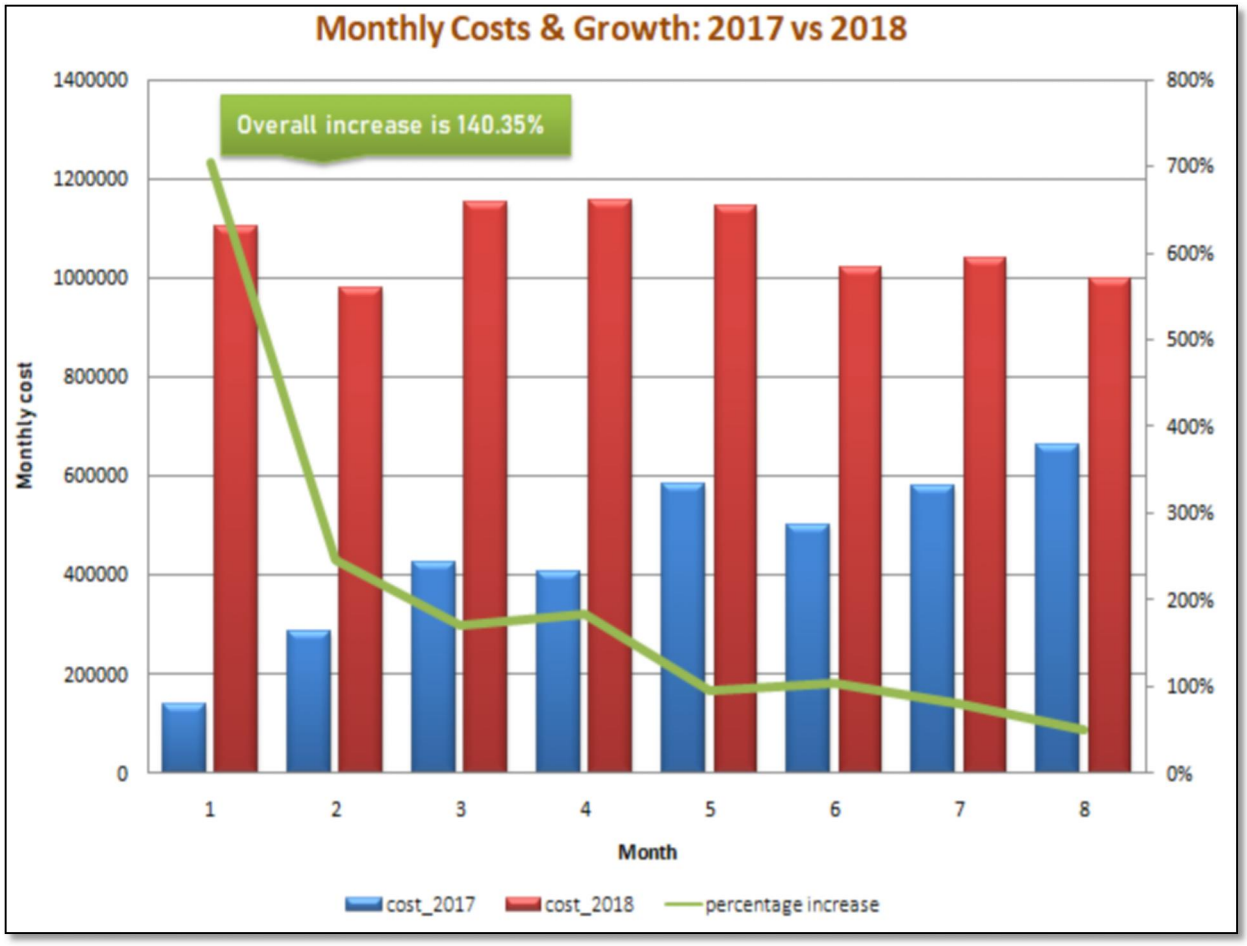
Query-

```
WITH monthly_costs AS (
  SELECT
    EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
    EXTRACT(MONTH FROM 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 order_purchase_timestamp) BETWEEN 1 AND 8
    AND EXTRACT(YEAR FROM order_purchase_timestamp) IN (2017, 2018)
    AND order_status NOT IN ('canceled', 'unavailable')
  GROUP BY year, month),
pivot_costs AS (
  SELECT month,
    MAX(CASE WHEN year = 2017 THEN total_cost END) AS cost_2017,
    MAX(CASE WHEN year = 2018 THEN total_cost END) AS cost_2018
  FROM monthly_costs
  GROUP BY month)
SELECT month,
  CONCAT(ROUND(((cost_2018 - cost_2017) / cost_2017) * 100, 2), '%') AS percent_increase,
  cost_2017,
  cost_2018
FROM pivot_costs
ORDER BY month;
```

Output(Jan to Aug)-

Row	month	percent_increase	cost_2017	cost_2018
1	1	704.81%	137006.7599999...	1102639.410000...
2	2	245.52%	283621.9400000...	979966.2300000...
3	3	170.81%	425656.4000000...	1152736.739999...
4	4	184.81%	405988.3799999...	1156303.909999...
5	5	96.55%	582926.1600000...	1145748.629999...
6	6	104.17%	499827.4700000...	1020494.289999...
7	7	79.64%	578858.5800000...	1039880.160000...
8	8	50.57%	662071.7700000...	996896.1499999...

➢ Visualisation-



Insights-

- **High Early Growth:** Month 1 shows a massive 704% increase compared to 2017.
- **Consistent 2018 Advantage:** Every month in 2018 exceeds the corresponding 2017 cost.
- **Declining Growth Trend:** The percent increase tapers from 704% in Month 1 down to 50% by Month 8.
- **Overall 140% Increase (Jan–Aug):** Summing all months, 2018's total cost is 140% higher than 2017's.

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

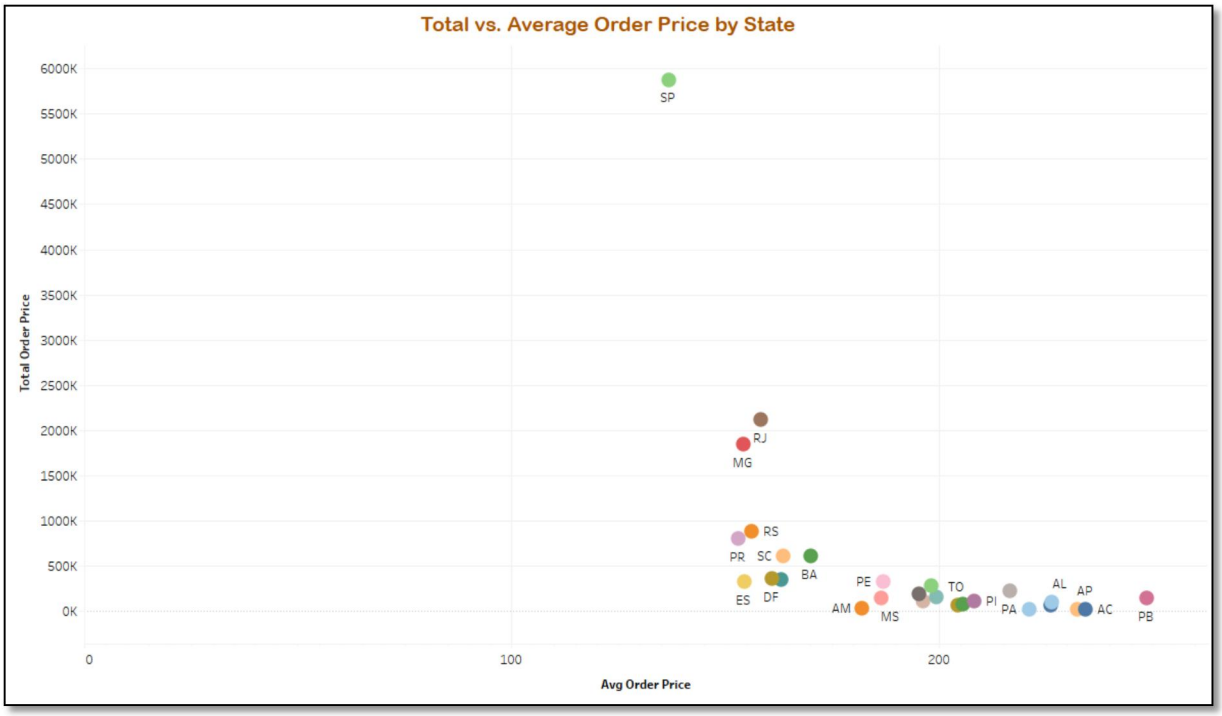
Query-

```
SELECT
  c.customer_state,
  SUM(p.payment_value) AS total_order_price,
  AVG(p.payment_value) AS avg_order_price
FROM `Target.Customers` c
JOIN `Target.Orders` o ON c.customer_id = o.customer_id
JOIN `Target.Payments` p ON o.order_id = p.order_id
WHERE order_status NOT IN ('canceled','unavailable')
GROUP BY c.customer_state
ORDER BY c.customer_state;
```

Output(First 10rows)-

Row	customer_state	total_order_price	avg_order_price
1	AC	19680.61999999...	234.2930952380...
2	AL	96252.69999999...	226.4769411764...
3	AM	27846.43999999...	182.0028758169...
4	AP	16262.79999999...	232.3257142857...
5	BA	607041.4500000...	169.9444148936...
6	CE	274549.8099999...	198.2309097472...
7	DF	351487.6299999...	160.9375595238...
8	ES	323105.7500000...	154.5221186035...
9	GO	340626.2400000...	163.0570799425...
10	MA	150691.7000000...	199.3276455026...
11	MG	1843460.370000...	154.2386521084...

➢ Visualisation-



Insights-

- **SP** stands out with the **highest total cost** but a **moderate average price**, implying **high order volume**.
- **MG and RJ** show **mid-range total cost** and **average price**.
- **Other states** cluster at **lower total** and **average prices**, indicating potential growth areas

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

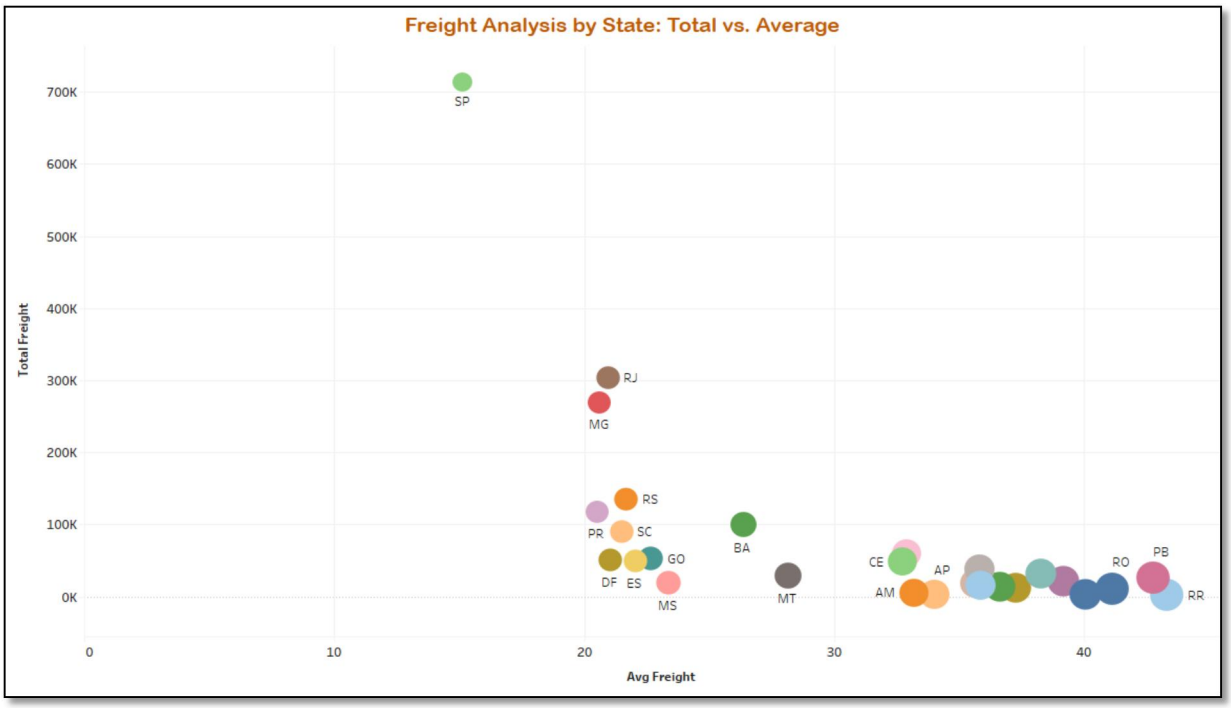
Query-

```
SELECT
  c.customer_state,
  SUM(oi.freight_value) AS total_freight,
  AVG(oi.freight_value) AS avg_freight
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
WHERE order_status NOT IN ('canceled','unavailable')
GROUP BY c.customer_state
ORDER BY c.customer_state;
```

Output(First 10rows)-

Row	customer_state	total_freight	avg_freight
1	AC	3686.7499999999991	40.073369565217391
2	AL	15914.589999999993	35.843671171171145
3	AM	5478.89	33.205393939393936
4	AP	2788.5000000000018	34.006097560975633
5	BA	99799.8299999991	26.36719418758258
6	CE	48258.5499999999	32.739857530529164
7	DF	50440.759999999769	21.04328744263664
8	ES	49548.899999999834	22.041325622775844
9	GO	52673.909999999822	22.674950495049519
10	MA	31396.340000000018	38.2882195121951

➢ Visualisation-



Insights-

- **SP Dominance:** SP is a clear outlier with very high total freight, indicating high shipping volume.
- **Moderate Mid-Ranges:** RJ and MG show moderate total freight with balanced average freight rates.
- **Low Volume Cluster:** Most states cluster at lower total and average freight, suggesting smaller order volumes overall.

5. Analysis based on sales, freight and delivery time.

a) Delivery Time & Difference Between Estimated and Actual Delivery.

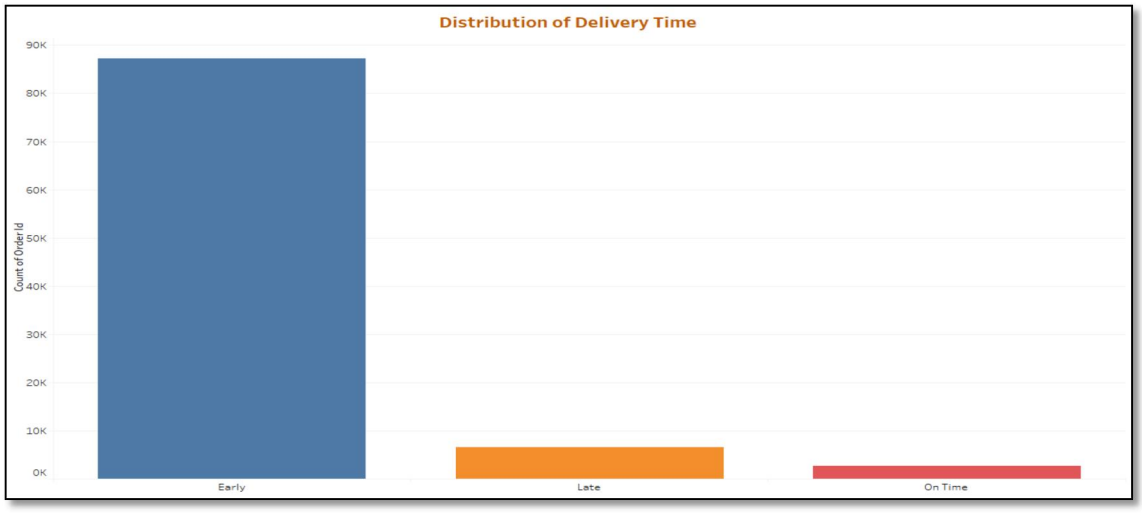
Query-

```
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_delivered_customer_date IS NOT NULL;
```

Output(First 10rows)-

Row	order_id	time_to_deliver	diff_estimated_delivery
1	65d1e226dfaeb8cdc42f66542...	35	16
2	2c45c33d2f9cb8ff8b1c86cc28...	30	28
3	1950d777989f6a877539f5379...	30	-12
4	bfbd0f9bdef84302105ad712db...	54	-36
5	98974b076b01553d49ee6467...	43	6
6	c4b41c36dd589e901f6879f25...	36	14
7	d2292ff2201e74c5db154d1b7...	29	20
8	95e01270fcbae986342340010...	30	19
9	ed8c7b1b3eb256c70ce0c7423...	44	5
10	5cc475c7c03290048eb2e742c...	68	-18

➢ **Visualisation-**



Insights-

- **Variable Delivery Times:** Durations range from about 30 to over 100 days, indicating significant variability in order processing and shipping times.
- **Mostly Early Deliveries:** Most orders arrive before the estimated date, which shows the system often delivers faster than expected, with only a few orders coming in late or exactly on time.
- **Operational Implications:** This pattern suggests that while the majority of deliveries are efficient, the high variability indicates potential areas for process improvement to reduce extreme delays.

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

Query-

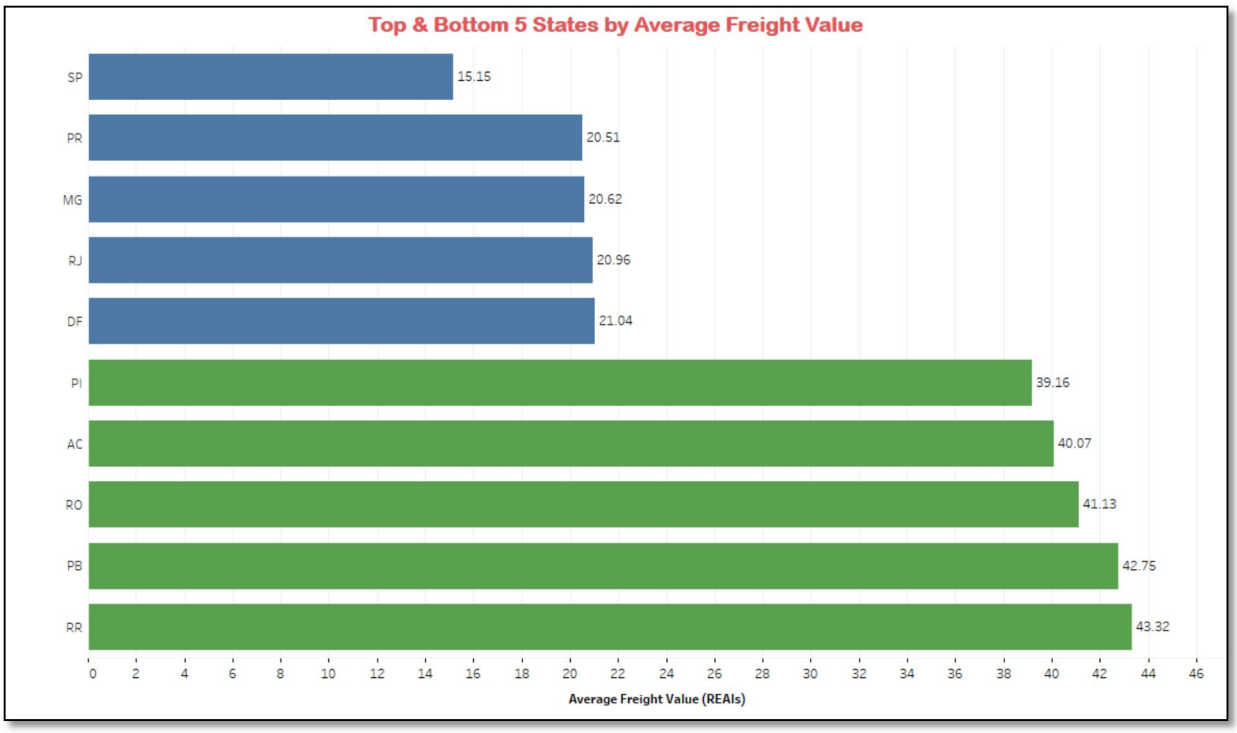
```
WITH state_freight AS (
  SELECT
    c.customer_state,
    AVG(oi.freight_value) AS avg_freight
  FROM `target-case-study-453312.Target.Customers` c
  JOIN `target-case-study-453312.Target.Orders` o
    ON c.customer_id = o.customer_id
  JOIN `target-case-study-453312.Target.Order-items` oi
    ON o.order_id = oi.order_id
  WHERE order_status NOT IN ('canceled','unavailable')
  GROUP BY c.customer_state
),
bottom5 AS (
  SELECT
    customer_state,
    ROUND(avg_freight, 2) AS avg_freight,
    'Bottom 5' AS Category
  FROM state_freight
  ORDER BY avg_freight ASC
  LIMIT 5
),
top5 AS (
  SELECT
    customer_state,
    ROUND(avg_freight, 2) AS avg_freight,
    'Top 5' AS Category
  FROM state_freight
  ORDER BY avg_freight DESC
  LIMIT 5
```

```
)
SELECT *
FROM (
  SELECT * FROM bottom5
  UNION ALL
  SELECT * FROM top5
) AS combined
ORDER BY Category, avg_freight ASC;
```

Output-

Row	customer_state	avg_freight	Category
1	SP	15.15	Bottom 5
2	PR	20.51	Bottom 5
3	MG	20.62	Bottom 5
4	RJ	20.96	Bottom 5
5	DF	21.04	Bottom 5
6	PI	39.16	Top 5
7	AC	40.07	Top 5
8	RO	41.13	Top 5
9	PB	42.75	Top 5
10	RR	43.32	Top 5

➢ Visualisation-



Insights-

- **Large Cost Gap:** Bottom 5 states (SP, PR, MG, RJ, DF) have average freight costs around 15–21 REAIs, while top 5 states (PI, AC, RO,PB, RR) range from about 39–43 REAIs, indicating a significant difference in shipping expenses.
- **Possible Geographic/Infrastructure Influences:** States with lower freight costs may be closer to distribution centers or have better logistics networks, whereas states with higher costs might face longer distances, remote locations, or less-developed transport routes.

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

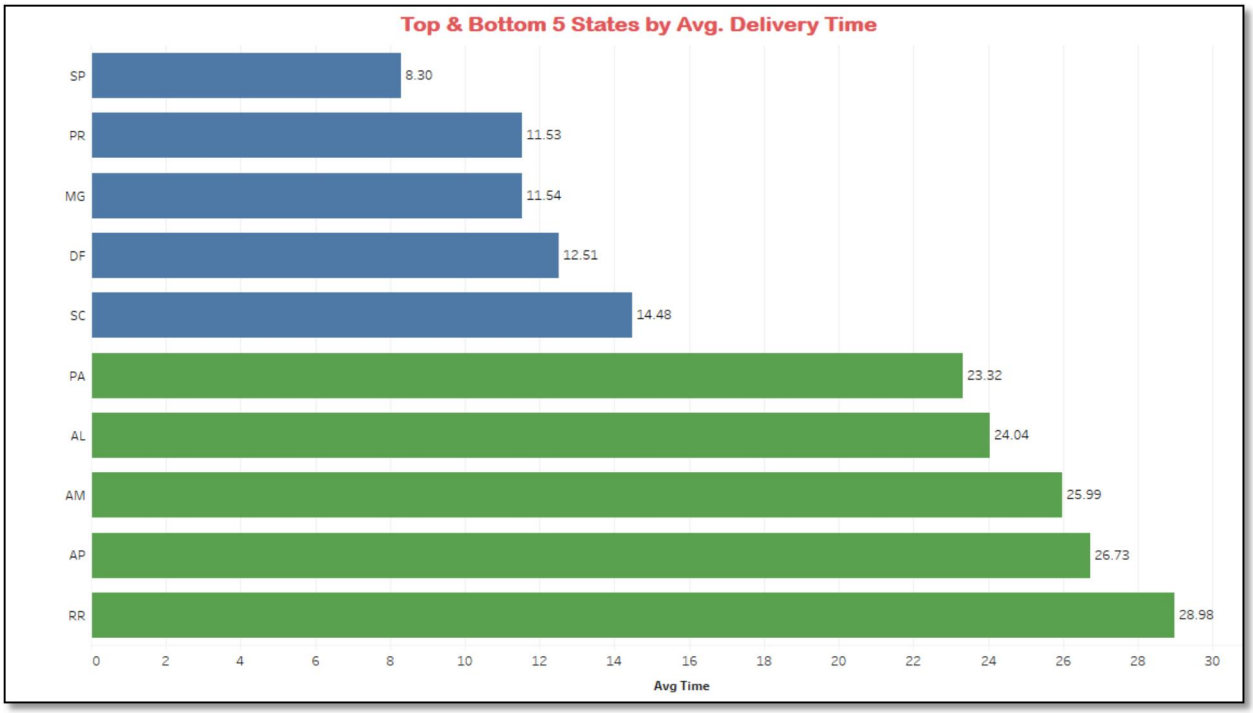
Query-

```
WITH sd AS (
  SELECT
    c.customer_state,
    ROUND(AVG(DATE_DIFF(o.order_delivered_customer_date, o.order_purchase_timestamp,
DAY)), 2) AS avg_time
  FROM `Target.Orders` o
  JOIN `Target.Customers` c ON o.customer_id = c.customer_id
  WHERE o.order_status = 'delivered'
  GROUP BY c.customer_state
),
ranked AS (
  SELECT
    customer_state,
    avg_time,
    ROW_NUMBER() OVER (ORDER BY avg_time ASC) AS rn_asc,
    ROW_NUMBER() OVER (ORDER BY avg_time DESC) AS rn_desc
  FROM sd
)
SELECT customer_state, avg_time, 'Bottom 5' AS Category
FROM ranked
WHERE rn_asc <= 5
UNION ALL
SELECT customer_state, avg_time, 'Top 5' AS Category
FROM ranked
WHERE rn_desc <= 5
ORDER BY avg_time ASC;
```

Output-

Row	customer_state	avg_time	Category
1	SP	8.3	Bottom 5
2	PR	11.53	Bottom 5
3	MG	11.54	Bottom 5
4	DF	12.51	Bottom 5
5	SC	14.48	Bottom 5
6	PA	23.32	Top 5
7	AL	24.04	Top 5
8	AM	25.99	Top 5
9	AP	26.73	Top 5
10	RR	28.98	Top 5

➢ Visualisation-



Insights-

- **Significant Gap Between Fastest and Slowest:** The top five states have an average delivery time between 8 to 15 days, while the bottom five range from 23 to 29 days. This shows a difference of about 15 to 20 days.
- **Fastest State:** SP has the shortest average delivery time of 8 days, indicating efficient logistics or proximity to distribution centers.
- **Slowest State:** RR has the longest average delivery time of 29 days, suggesting possible delays due to logistics challenges or distance.

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

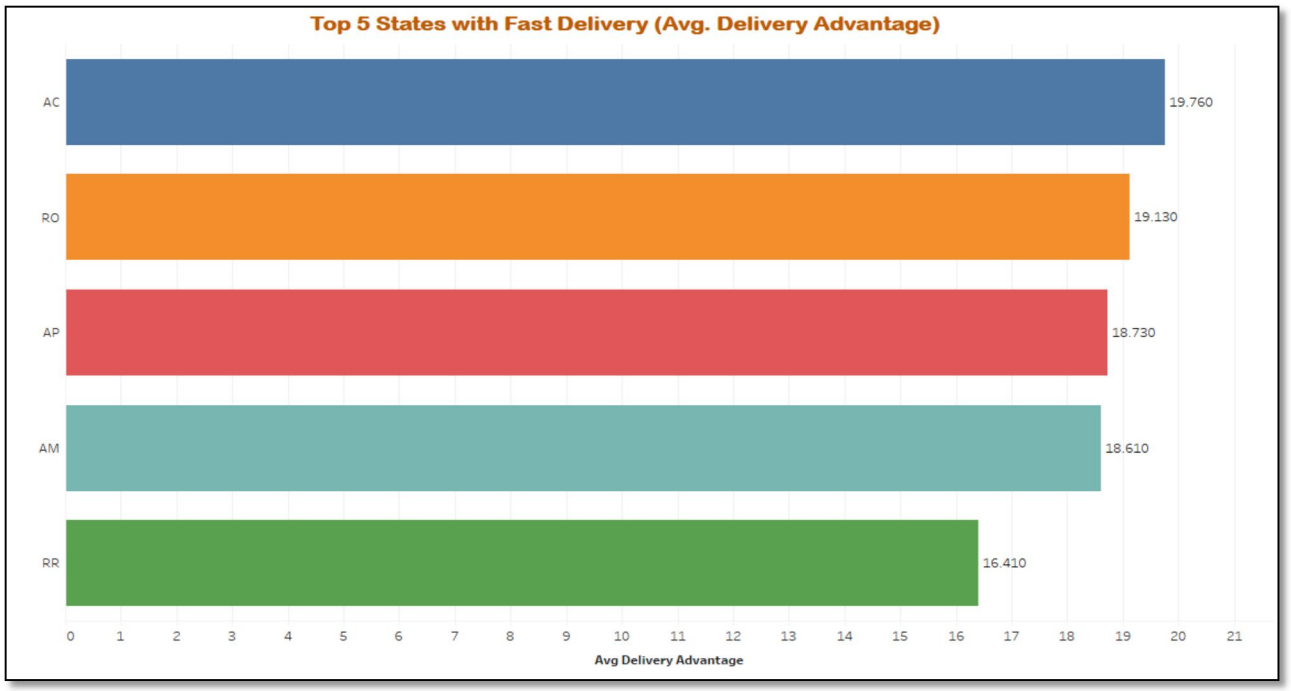
Query-

```
SELECT
  c.customer_state,
  ROUND(AVG(DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date,
DAY)), 2) AS avg_delivery_advantage
FROM `Target.Customers` c
JOIN `Target.Orders` o ON c.customer_id = o.customer_id
WHERE order_status = 'delivered'
GROUP BY c.customer_state
ORDER BY avg_delivery_advantage DESC
LIMIT 5;
```

Output-

Row	customer_state	avg_delivery_advantage
1	AC	19.76
2	RO	19.13
3	AP	18.73
4	AM	18.61
5	RR	16.41

➢ Visualisation-



Insights-

- **AC Leads with the Greatest Advantage:** AC delivers almost 20 days earlier than estimated, standing out as the fastest.
- **Close Performance Among the Top 5:** RO, AP, AM, and RR also show high averages (16 to 19 days early), indicating consistently fast delivery.
- **Significantly Ahead of Estimates:** All five states deliver more than two weeks earlier than expected, suggesting very efficient logistics or shorter shipping distances in these regions

6. Analysis based on the payments.

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

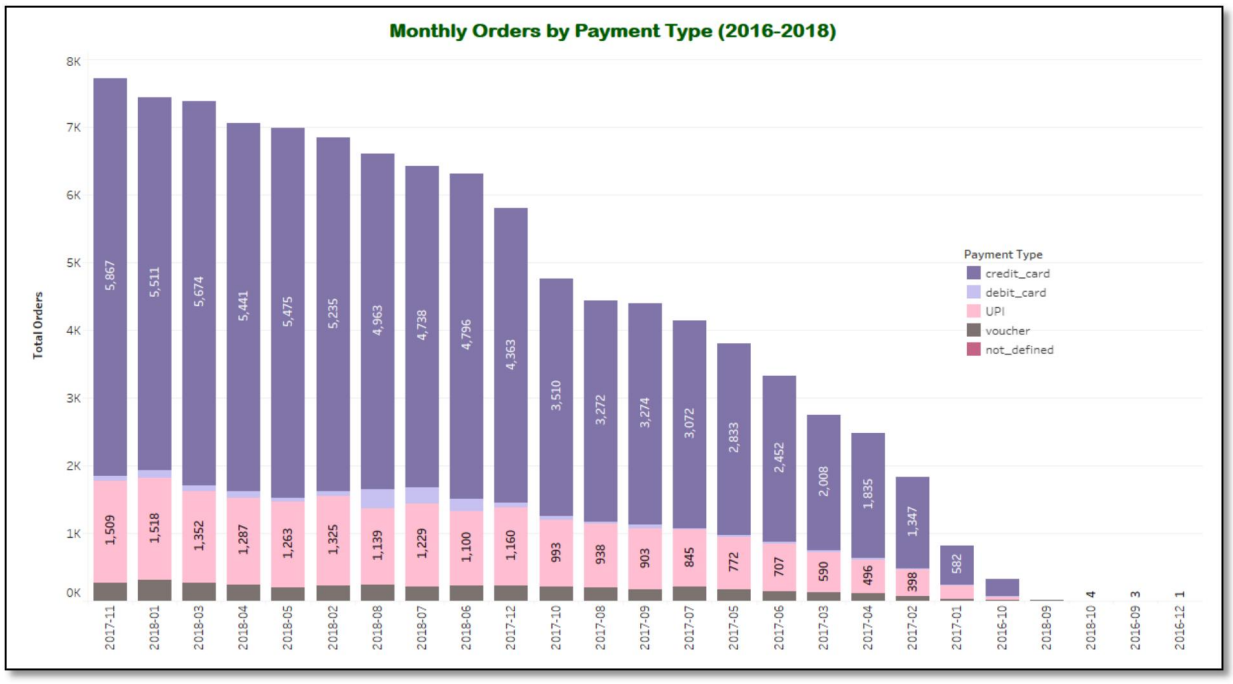
Query-

```
SELECT
  FORMAT_TIMESTAMP('%Y-%m', orders.order_purchase_timestamp) AS order_month,
  payments.payment_type,
  COUNT(DISTINCT orders.order_id) AS total_orders
FROM `target-case-study-453312.Target.Orders` AS orders
JOIN `target-case-study-453312.Target.Payments` AS payments
  ON orders.order_id = payments.order_id
GROUP BY order_month, payments.payment_type
ORDER BY order_month, payments.payment_type;
```

Output-

Row	order_month	payment_type	total_orders
1	2016-09	credit_card	3
2	2016-10	UPI	63
3	2016-10	credit_card	253
4	2016-10	debit_card	2
5	2016-10	voucher	11
6	2016-12	credit_card	1
7	2017-01	UPI	197
8	2017-01	credit_card	582
9	2017-01	debit_card	9
10	2017-01	voucher	33

➢ Visualisation-



Insights-

- **Credit cards** were the most used payment method throughout the period.
- **UPI payments** showed a rising trend, indicating increased digital adoption.
- **Peak orders occurred in November 2017**, likely due to festive shopping.
- **Order volume declined after mid-2018**, showing a downward trend.
- **Debit cards and vouchers** had significantly lower usage compared to credit cards and UPI.
- The data suggests a **gradual shift towards digital payment methods** over time.

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

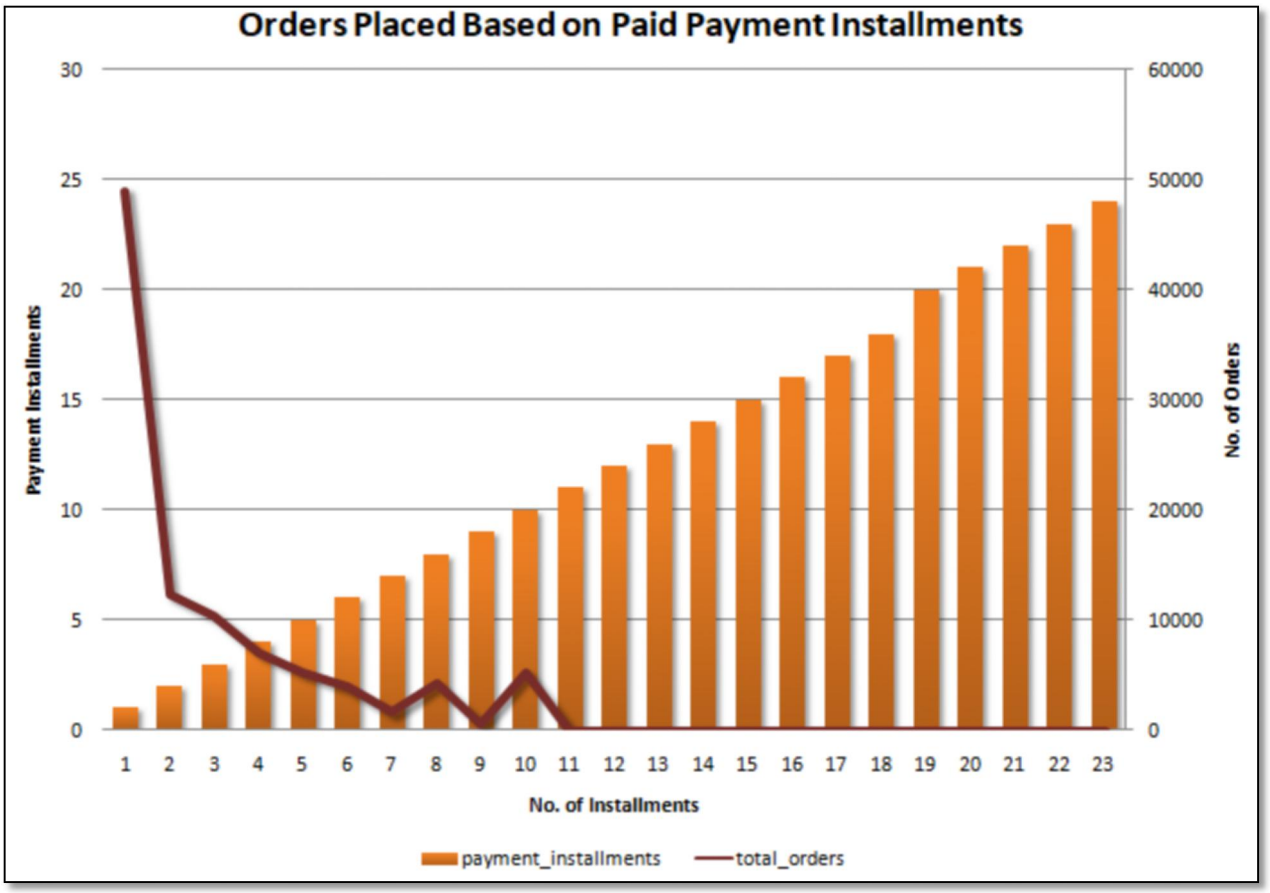
Query-

```
SELECT
   payment_installments,
   COUNT(DISTINCT order_id) AS No_of_Orders
FROM `target-case-study-453312.Target.Payments`
WHERE payment_installments <> 0
GROUP BY payment_installments
ORDER BY payment_installments;
```

Output-

Row	payment_installments	total_orders
10	10	5315
11	11	23
12	12	133
13	13	16
14	14	15
15	15	74
16	16	5
17	17	8
18	18	27
19	20	17
20	21	3
21	22	1
22	23	1
23	24	18

➢ Visualisation-



Insights-

- **Most Orders in Single Installment** – Majority of customers prefer one-time payments.
- **Drop in Orders for Fewer Installments** – Orders decrease sharply for 2-5 installments.
- **Gradual Rise for Higher Installments** – Orders increase from 10+ installments.
- **Inverse Trend** – Higher installments see fewer users, but total orders rise.
- **Peak at 23 Installments** – Long-term financing is used for high-value purchases.

Recommendations-

- **Expand Reach**: Currently serving 4113 cities; target full coverage across Brazil (8011+ cities, 27 states) for broader accessibility.
- **Boost Low-Order Months**: Use market research, partnerships, promotions, and seasonal marketing (New Year, Black Friday, Carnival, FIFA, etc.).
- **Optimize Logistics**: Improve shipping, negotiate costs, and partner with local carriers to reduce delivery times and enhance operations.
- **Enhance Customer Communication**: Provide real-time order updates and proactive delivery notifications.
- **Secure & Flexible Payments**: Support diverse payment methods, educate customers, and promote installment options with incentives.
- **Leverage Customer Insights**: Collect feedback, analyze preferences, and use branding to enhance customer experience and loyalty.
- **Targeted Strategies**: Focus on engagement in high-customer states and growth strategies in low-customer states.