

# Business Case: Target – SQL



## Introduction

Target is a globally renowned brand and one of the leading retailers in the United States. Known for its commitment to outstanding value, innovation, and exceptional customer experiences, Target has established itself as a preferred shopping destination worldwide.

This business case study focuses on Target's operations in Brazil, analysing a dataset containing detailed information about 100,000 orders placed between 2016 and 2018. The dataset provides valuable insights across multiple dimensions, including order status, pricing, payment performance, shipping logistics, customer location, product attributes, and customer reviews.

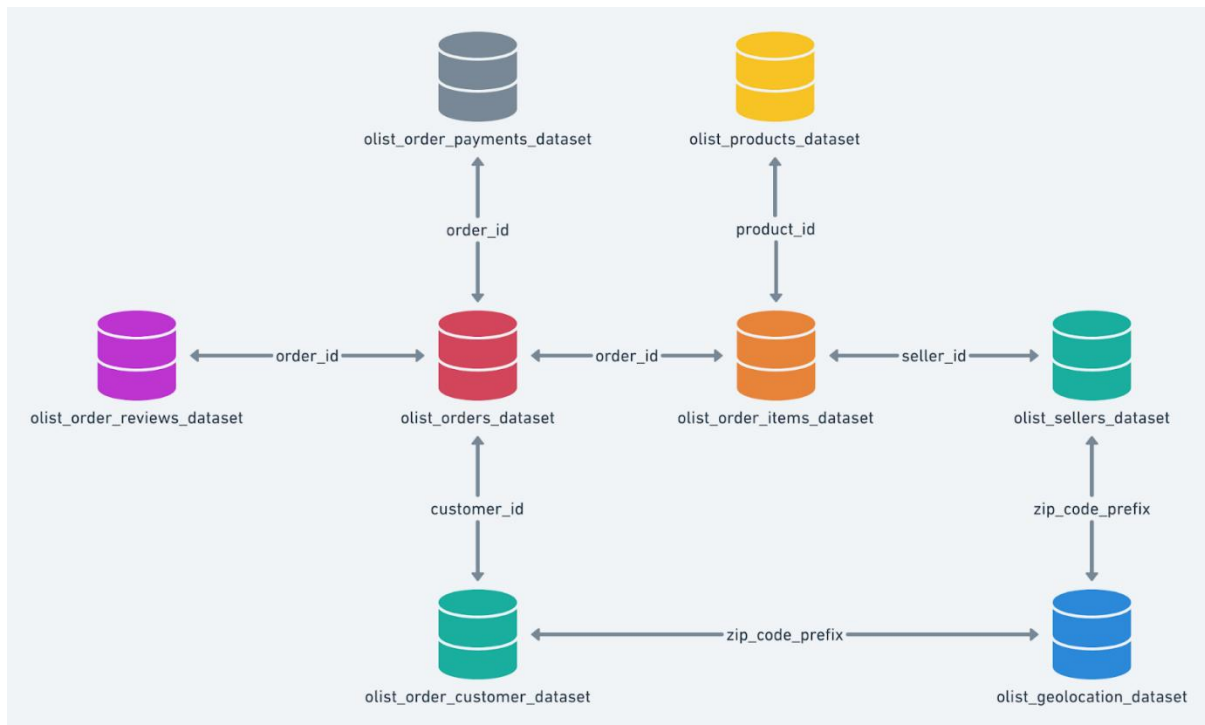
Through an in-depth exploration of this dataset, the study aims to uncover key insights into Target's operational performance in Brazil. By examining factors such as order processing efficiency, pricing strategies, payment trends, shipping performance, customer demographics, product characteristics, and customer satisfaction levels, this analysis seeks to highlight areas of strength and opportunities for improvement in Target's operations in the Brazilian market.

## Dataset

This dataset provides a robust foundation for analysing:

- **Customer demographics and purchasing behaviours.**
- **Order trends, pricing, and shipping performance.**
- **Product characteristics and customer satisfaction.**
- **Seller contributions and geolocation insights.**

By integrating and analysing these datasets, actionable insights can be derived to optimize operations, enhance customer experiences, and drive strategic growth for Target in Brazil.



The diagram showcases the relationships between datasets in the Olist system, with the **olist\_orders\_dataset** as the central hub. It connects to:

- **Customers:** Via **customer\_id** (**olist\_order\_customer\_dataset**).
- **Order Items:** Via **order\_id** (**olist\_order\_items\_dataset**), which links to products (**olist\_products\_dataset**) and sellers (**olist\_sellers\_dataset**).
- **Payments:** Via **order\_id** (**olist\_order\_payments\_dataset**).
- **Reviews:** Via **order\_id** (**olist\_order\_reviews\_dataset**).
- **Geolocation:** Linked through **zip\_code\_prefix** for customers and sellers (**olist\_geolocation\_dataset**).

This structure supports comprehensive analysis of orders, customers, products, payments, and geography.

## Lets do some High Level checks:

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

```
3
4 SELECT COLUMN_NAME, DATA_TYPE, IS_NULLABLE, COLUMN_DEFAULT
5 FROM `adroit-nectar-446912-b8.Target_Case_Study.INFORMATION_SCHEMA.COLUMNS`
6 WHERE TABLE_NAME = 'customers';
```

Press Alt+F1 for Accessibility Q

Query results [SAVE RESULTS](#) [OPEN IN](#)

JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS	EXECUTION GRAPH
Row	COLUMN_NAME	DATA_TYPE	IS_NULLABLE	COLUMN_DEFAULT	
1	customer_id	STRING	YES	NULL	
2	customer_unique_id	STRING	YES	NULL	
3	customer_zip_code_prefix	INT64	YES	NULL	
4	customer_city	STRING	YES	NULL	
5	customer_state	STRING	YES	NULL	

### 2.) The time range between which the orders were placed

```
1 SELECT max(order_purchase_timestamp) as earliest_order_stamp,
2 | min(order_purchase_timestamp) as latest_order_stamp
3 FROM `Target_Case_Study.orders`;
```

Press Alt+F1 for Acc

Query results [SAVE RESULTS](#) [OPEN IN](#)

JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS	EXECUTION GRAPH
Row	earliest_order_stamp	latest_order_stamp			
1	2018-10-17 17:30:18 UTC	2016-09-04 21:15:19 UTC			

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

```
1 select count(distinct customer_city) as cust_city_count,
2 | count(distinct customer_state) as cust_state_count
3 from(
4 | (SELECT customer_id FROM `Target_Case_Study.orders`) o
5 | join
6 | ((SELECT customer_id,
7 | customer_city,
8 | customer_state
9 | FROM `Target_Case_Study.customers`) c
10 | on o.customer_id=c.customer_id
11 | )
```

Query results [SAVE RESULTS](#)

JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS
Row	cust_city_count	cust_state_count		
1	4119	27		

## Summary

### 1. Order Timeline:

- The dataset captures orders placed over a period spanning from **September 4, 2016**, to **October 17, 2018**.
- This provides a **two-year snapshot** of Target's operations in Brazil, offering comprehensive insights into customer behaviour, operational efficiency, and market trends.

### 2. Customer Distribution:

- Orders were placed by customers from **4,119 unique cities** across **27 states** in Brazil.
- This highlights the **geographical diversity** of Target's customer base and underscores the company's extensive market reach within the country.

## Lets do some in-depth Exploration:

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

<pre>1 SELECT EXTRACT(YEAR FROM order_purchase_timestamp) AS order_year, 2 count(*) as order_count 3 FROM `Target_Case_Study.orders` 4 group by 1 5 order by 1;</pre>					Press Alt
Query results <a href="#">SAVE RESULTS</a>					
JOB INFORMATION RESULTS CHART JSON EXECUTION DETAILS EXECUTION GRAPH					
Row	order_year	order_count			
1	2016	329			
2	2017	45101			
3	2018	54011			

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

```
1 SELECT order_month_name, AVG(order_count) AS avg_order_count
2 FROM (
3     SELECT EXTRACT(YEAR FROM order_purchase_timestamp) AS order_year,
4             FORMAT_TIMESTAMP('%B', order_purchase_timestamp) AS order_month_name,
5             EXTRACT(MONTH FROM order_purchase_timestamp) AS order_month_number,
6             COUNT(*) AS order_count
7     FROM `Target_Case_Study.orders`
8     GROUP BY 1, 2, 3
9 )
10 WHERE order_year <> 2016
11 GROUP BY order_month_name, order_month_number
12 ORDER BY order_month_number;
```

Order Month	Average Order Count
January	4034.5
February	4254.0
March	4946.5
April	4671.5
May	5286.5
June	4706.0
July	5159.0
August	5421.5
September	2150.5
October	2317.5
November	7544.0
December	5673.0

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

```

1 SELECT case when EXTRACT(HOUR FROM order_purchase_timestamp) between 0 and 6 then 'Dawn'
2           | when EXTRACT(HOUR FROM order_purchase_timestamp) between 7 and 12 then 'Mornings'
3           | when EXTRACT(HOUR FROM order_purchase_timestamp) between 13 and 18 then 'Afternoon'
4           | when EXTRACT(HOUR FROM order_purchase_timestamp) between 19 and 23 then 'Night' end AS order_time,
5           COUNT(*) AS order_count
6 FROM `Target_Case_Study.orders`
7 GROUP BY 1
8 order by 1

```

Press Alt+F1 for Accessibility

Query results

[SAVE RESULTS](#)

[OPEN IN](#)

JOB INFORMATION		RESULTS	CHART	JSON	EXECUTION DETAILS	EXECUTION GRAPH
Row	order_time	order_count				
1	Afternoon	38135				
2	Dawn	5242				
3	Mornings	27733				
4	Night	28331				

## Summary

### 1. Yearly Order Trend:

- **2016:** The dataset contains 329 orders, reflecting partial data for the year.
- **2017:** The number of orders significantly increased to **45,101**, indicating strong growth.
- **2018:** Orders further rose to **54,011**, demonstrating consistent upward momentum.

## 2. Monthly Order Distribution:

- **Peak months:** November shows the highest average order count (**7,544**), followed by August (**5,421.5**) and December (**5,673**), likely driven by promotional events and holiday shopping.
- **Low activity months:** September (**2,150.5**) and October (**2,317.5**) have the lowest average order counts, suggesting potential opportunities to enhance sales during these periods.
- Overall, there is a **seasonal pattern**, with notable spikes in Q4 and steady activity during the first half of the year.

## 3. Order Time Analysis:

- **Afternoon** is the busiest period, with the highest order volume.
- **Dawn** sees the fewest orders, indicating minimal activity during this time.
- **Night and Morning** exhibit comparable order activity, highlighting consistent demand outside peak hours.

This analysis provides insights into **annual trends, seasonal demand patterns, and time-of-day preferences**, enabling Target to optimize operations, inventory, and marketing efforts effectively.

## Actionable Items for Business

### 1. Leverage Peak Sales Periods:

Focus marketing campaigns and promotional efforts during high-demand months such as **November, December, and August**. Introduce targeted discounts, bundle offers, and advertising to capitalize on the seasonal surge and maximize revenue during these periods.

### 2. Boost Engagement During Low-Activity Times:

Address the dip in sales during **September and October** by launching special promotions, loyalty rewards, or events to stimulate demand. Additionally, explore ways to incentivize purchases during off-peak hours like **Dawn** by offering limited-time deals or exclusive perks for early-bird shoppers.

## Evolution of E-commerce orders in the Brazil region:

1.) month on month no. of orders placed in each state.

```

1 select *,
2 order_count-lag(order_count) over(order by Year,Month) as month_on_month_orders from
3 (SELECT EXTRACT(YEAR FROM order_purchase_timestamp) as Year,
4         EXTRACT(MONTH FROM order_purchase_timestamp) as Month,
5         COUNT(*) AS order_count
6 FROM `Target_Case_Study.orders`
7 GROUP BY 1,2) t
8 order by 1,2

```

Press Alt+F1 for Accessibility (

Query results

[SAVE RESULTS](#) ▾

[OPEN IN](#) ▾

Year	Month	order_count	month_on_month_orders
2016	9	4	
2016	10	324	320
2016	12	1	-323
2017	1	800	799
2017	2	1780	980
2017	3	2682	902
2017	4	2404	-278
2017	5	3700	1296
2017	6	3245	-455
2017	7	4026	781
2017	8	4331	305
2017	9	4285	-46
2017	10	4631	346
2017	11	7544	2913
2017	12	5673	-1871
2018	1	7269	1596
2018	2	6728	-541
2018	3	7211	483
2018	4	6939	-272
2018	5	6873	-66
2018	6	6167	-706
2018	7	6292	125
2018	8	6512	220
2018	9	16	-6496
2018	10	4	-12

2.) How are the customers distributed across all the states?

```

1 SELECT customer_state,
2        count(distinct customer_id) as cust_count
3 FROM `Target_Case_Study.customers`
4 group by 1
5 order by 2 desc

```

Query results

customer_state	cust_count
SP	41746

RJ	12852
MG	11635
RS	5466
PR	5045
SC	3637
BA	3380
DF	2140
ES	2033
GO	2020
PE	1652
CE	1336
PA	975
MT	907
MA	747
MS	715
PB	536
PI	495
RN	485
AL	413
SE	350
TO	280
RO	253
AM	148
AC	81
AP	68
RR	46

## Summary

### 1. Order Trends

- **Growth:** Orders grew significantly from 2016 to 2018, peaking at **7544 orders in November 2017**.
- **Seasonality:** The highest growth occurred in **November 2017** (+2913), while **September 2018** saw a steep decline (-6496). Late 2018 recorded minimal orders, indicating seasonal or operational challenges.

### 2. Customer Distribution

- **Top States:** **São Paulo (SP)** leads with **41.75% (41,746 customers)**, followed by **Rio de Janeiro (RJ)** and **Minas Gerais (MG)**.
- **Opportunities:** Northern states like **Roraima (RR)**, **Amapá (AP)**, and **Acre (AC)** have minimal representation, presenting untapped market potential.
- **Regional Focus:** Southeastern and Southern regions dominate, requiring tailored strategies for these areas.



## Actionable Items for Business

### 1. Expand Operations in Untapped Regions

Invest in marketing campaigns, logistical infrastructure, and localized promotions targeting underrepresented states like **Roraima (RR)**, **Amapá (AP)**, and **Acre (AC)** to capture potential customers in these areas.

### 2. Leverage Peak Seasonal Demand

Focus resources on **November**, the busiest month, by launching targeted discounts, advertising campaigns, and ensuring optimal stock and delivery capacity to maximize sales during this high-demand period.

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

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

```
1 select round(((cost_2018-cost_2017)*100/cost_2017),4) as perc_increase from(
2   select sum(case when order_year=2018 then payment_value end) as cost_2018,
3     sum(case when order_year=2017 then payment_value end) as cost_2017
4   from
5     (SELECT order_id,payment_value FROM `Target_Case_Study.payments`) p
6     join
7     (SELECT order_id,order_purchase_timestamp,extract(YEAR from order_purchase_timestamp) as order_year
8      FROM `Target_Case_Study.orders`) o
9     on p.order_id=o.order_id
10    where extract(MONTH from order_purchase_timestamp) between 1 and 8)
```

Press Alt+F1 for Accessibility Op

Query results [SAVE RESULTS](#) [OPEN IN](#)

JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS	EXECUTION GRAPH
Row	perc_increase				
1	136.9769				

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

```
1 select customer_state,
2   round(sum(price),3) as Total_order_price,
3   round(avg(price),3) as Avg_order_price
4 from(
5   (SELECT order_id,price FROM `Target_Case_Study.order_items`) p
6   join
7   (SELECT order_id,customer_id FROM `Target_Case_Study.orders`) o
8   on p.order_id=o.order_id
9   join
10  (SELECT customer_id,customer_state FROM `Target_Case_Study.customers`) c
11  on c.customer_id=o.customer_id
12 group by 1
13 order by 2 desc
```

Press Alt+F1 for Accessibility Options

Query results [SAVE RESULTS](#) [OPEN IN](#)

customer_state	Total_order_price	Avg_order_price
SP	5202955.05	109.654

RJ	1824092.67	125.118
MG	1585308.03	120.749
RS	750304.02	120.337
PR	683083.76	119.004
SC	520553.34	124.654
BA	511349.99	134.601
DF	302603.94	125.771
GO	294591.95	126.272
ES	275037.31	121.914
PE	262788.03	145.508
CE	227254.71	153.758
PA	178947.81	165.692
MT	156453.53	148.297
MA	119648.22	145.204
MS	116812.64	142.628
PB	115268.08	191.475
PI	86914.08	160.358
RN	83034.98	156.966
AL	80314.81	180.889
SE	58920.85	153.041
TO	49621.74	157.529
RO	46140.64	165.974
AM	22356.84	135.496
AC	15982.95	173.728
AP	13474.3	164.321
RR	7829.43	150.566

### 3.) Total & Average value of order freight for each state

```

1  select customer_state,
2      round(sum(freight_value),3) as Total_order_freight,
3      round(avg(freight_value),3) as Avg_order_freight
4  from(
5      (SELECT order_id,freight_value FROM `Target_Case_Study.order_items`) p
6      join
7      (SELECT order_id,customer_id FROM `Target_Case_Study.orders`) o
8      on p.order_id=o.order_id
9      join
10     (SELECT customer_id,customer_state FROM `Target_Case_Study.customers`) c
11     on c.customer_id=o.customer_id)
12  group by 1
13  order by 2 desc

```

Press Alt+F1 for Accessibility Options

Query results

 SAVE RESULTS ▾

 OPEN IN ▾



customer_state	Total_order_freight	Avg_order_freight
SP	718723.07	15.147
RJ	305589.31	20.961
MG	270853.46	20.63
RS	135522.74	21.736
PR	117851.68	20.532
BA	100156.68	26.364

SC	89660.26	21.47
PE	59449.66	32.918
GO	53114.98	22.767
DF	50625.5	21.041
ES	49764.6	22.059
CE	48351.59	32.714
PA	38699.3	35.833
MA	31523.77	38.257
MT	29715.43	28.166
PB	25719.73	42.724
PI	21218.2	39.148
MS	19144.03	23.375
RN	18860.1	35.652
AL	15914.59	35.844
SE	14111.47	36.653
TO	11732.68	37.247
RO	11417.38	41.07
AM	5478.89	33.205
AC	3686.75	40.073
AP	2788.5	34.006
RR	2235.19	42.984

## Summary

### Year-on-Year Growth

The total cost of orders increased significantly by **137%** from 2017 to 2018, highlighting strong growth in overall sales volume and value.

#### 1. State-Wise Sales Analysis

- **São Paulo (SP)** is the top-performing state with a **Total Order Price** of **5.2M** and the highest number of orders, though its **Average Order Price (109.65)** is lower than many other states.
- States like **Pernambuco (PE)**, **Ceará (CE)**, and **Pará (PA)** exhibit the highest **Average Order Prices** at **145.51**, **153.76**, and **165.69**, respectively, indicating higher-value purchases in these regions.

#### 2. Freight Costs Insights

- **São Paulo (SP)** has the highest total freight cost (**718K**), but its **Average Freight Cost (15.15)** is the lowest among all states, likely due to proximity to warehouses or optimized logistics.
- States like **Roraima (RR)**, **Pernambuco (PE)**, and **Paraíba (PB)** have significantly higher **Average Freight Costs (42.98, 32.92, and 42.72)**, suggesting logistical challenges or less frequent delivery routes.

## Actionable Items for Business

### 1. Optimize Freight Logistics for High-Cost States

- Focus on improving delivery networks and partnerships in states with high average freight costs, such as **Roraima (RR)**, **Paraíba (PB)**, and **Pernambuco (PE)**.
- Establish regional warehouses or negotiate better shipping rates to reduce freight expenses and make purchases more attractive to customers in these areas.

### 2. Expand Marketing in High-Potential States with High Average Order Prices

- Invest in targeted marketing campaigns for underperforming but high-value states like **Acre (AC)**, **Amapá (AP)**, and **Roraima (RR)** to attract more customers and increase order volume.
- Highlight premium products and services tailored to the purchasing behavior of customers in these regions.

## Analysis based on sales, freight, and delivery time:

1.) No. of days taken to deliver each order from the order's purchase date as delivery time & The difference (in days) between the estimated & actual delivery date of an order

```
1 select round(avg(delv_time),4) as avg_delv_time,
2 round(avg(diff_est_time),4) as avg_diff_est_time from(
3 SELECT order_id,
4 date_diff(order_delivered_customer_date,order_purchase_timestamp,DAY) as delv_time,
5 date_diff(order_delivered_customer_date,order_estimated_delivery_date,DAY) as diff_est_time
6 FROM `Target_Case_Study.orders`
7 where order_delivered_customer_date is not null) t1
```

Press Alt+F1

Query results [SAVE RESULTS](#)

JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS	EXECUTION GRAPH
row	avg_delv_time	avg_diff_est_time			
1	12.0941	-10.958			

### 2.) Top 5 states with the highest & lowest average freight value

```
1 select customer_state as states_high_freight,
2 round(avg(freight_value),3) as Avg_order_freight
3 from(
4 (SELECT order_id,freight_value FROM `Target_Case_Study.order_items`) p
5 join
6 (SELECT order_id,customer_id FROM `Target_Case_Study.orders`) o
7 on p.order_id=o.order_id
8 join
9 (SELECT customer_id,customer_state FROM `Target_Case_Study.customers`) c
10 on c.customer_id=o.customer_id
11 group by 1
12 order by 2 desc
13 limit 5
```

Press Alt+F1 for Accessibility

Query results [SAVE RESULTS](#) [OPEN IN](#)

states_high_freight	Avg_order_freight
RR	42.984
PB	42.724
RO	41.07
AC	40.073
PI	39.148

states_low_freight	Avg_order_freight
SP	15.147
PR	20.532
MG	20.63
RJ	20.961
DF	21.041

### 3.) Top 5 states with the highest & lowest average delivery time

```

1 SELECT customer_state,
2    round(avg(date_diff(order_delivered_customer_date,order_estimated_delivery_date,DAY)),4) as avg_diff_est_time
3 FROM `Target_Case_Study.orders` o
4 join `Target_Case_Study.customers` c
5 on o.customer_id=c.customer_id
6 where order_delivered_customer_date is not null
7 group by 1
8 order by 2 desc
9 limit 5

```

#### Query results

customer_state	avg_delv_time
RR	28.9756
AP	26.7313
AM	25.9862
AL	24.0403
PA	23.3161

customer_state	avg_delv_time
SP	8.2981
PR	11.5267
MG	11.5438
DF	12.5091
SC	14.4796

### 4.) Top 5 states where the order delivery is really fast as compared to the estimated date of delivery

```

1 SELECT customer_state,
2    round(avg(date_diff(order_delivered_customer_date,order_estimated_delivery_date,DAY)),4) as avg_diff_est_time
3 FROM `Target_Case_Study.orders` o
4 join `Target_Case_Study.customers` c
5 on o.customer_id=c.customer_id
6 where order_delivered_customer_date is not null
7 group by 1
8 order by 2
9 limit 5

```

Press Alt+F1 for Accessibility Options

#### Query results

[SAVE RESULTS](#) [OPEN IN](#)

JOB INFORMATION	RESULTS	CHART	JSON	EXECUTION DETAILS	EXECUTION GRAPH
row	customer_state	avg_diff_est_time			
1	AC	-19.7625			
2	RO	-19.1317			
3	AP	-18.7313			
4	AM	-18.6069			
5	RR	-16.4146			

## Summary

### 1. Delivery Time Analysis:

- The average delivery time across all states is approximately **12.09 days**.

- On average, orders are delivered **10.96 days earlier** than the estimated delivery time.
- States with the highest average delivery times include **RR (28.98 days)**, **AP (26.73 days)**, and **AM (25.99 days)**.
- States with the lowest delivery times include **SP (8.30 days)**, **PR (11.53 days)**, and **MG (11.54 days)**.

## 2. Freight Cost Analysis:

- States with the highest average freight costs include **RR (₹42.98)**, **PB (₹42.72)**, and **RO (₹41.07)**.
- States with the lowest average freight costs include **SP (₹15.15)**, **PR (₹20.53)**, and **MG (₹20.63)**.

## 3. Delivery vs. Freight Observations:

- **RR** has the highest freight costs and longest delivery times, suggesting logistical inefficiencies.
- **SP**, with the lowest freight costs and shortest delivery times, is an operationally efficient state.

## Actionable Items for Business

### 1. Optimize Logistics for High-Cost States:

- Focus on states with high average delivery times and freight costs, such as **RR**, **AP**, and **AM**. Consider establishing localized distribution centers or negotiating better rates with shipping providers to reduce costs and delivery delays.

### 2. Leverage Efficient States for Best Practices:

- Analyse the logistics processes in states like **SP**, **PR**, and **MG**, which have low delivery times and freight costs. Apply similar practices or strategies in states with inefficiencies to improve overall performance.

## Analysis based on the payments:

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

```

1 select *,
2 lag(order_count) over(partition by payment_type order by order_year,order_month) - order_count as mom_orders from
3 (
4 select order_year,order_month,payment_type,count(*) as order_count from (
5 (select order_id,payment_type FROM `Target_Case_Study.payments`) p
6 join
7 (select order_id,
8 extract(MONTH from order_purchase_timestamp) as order_month,
9 extract(YEAR from order_purchase_timestamp) as order_year
10 from `Target_Case_Study.orders`) o
11 on p.order_id=o.order_id)
12 group by 1,2,3)
13 order by 1,2,3

```

Press Alt+F1 for Accessibility Option

Query results

 SAVE RESULTS ▾

 OPEN IN ▾



order_year	order_month	payment_type	order_count	mom_orders
2016	9	credit_card	3	
2016	10	UPI	63	
2016	10	credit_card	254	-251
2016	10	debit_card	2	
2016	10	voucher	23	
2016	12	credit_card	1	253
2017	1	UPI	197	-134
2017	1	credit_card	583	-582
2017	1	debit_card	9	-7
2017	1	voucher	61	-38
2017	2	UPI	398	-201
2017	2	credit_card	1356	-773
2017	2	debit_card	13	-4
2017	2	voucher	119	-58
2017	3	UPI	590	-192
2017	3	credit_card	2016	-660
2017	3	debit_card	31	-18
2017	3	voucher	200	-81
2017	4	UPI	496	94
2017	4	credit_card	1846	170
2017	4	debit_card	27	4
2017	4	voucher	202	-2
2017	5	UPI	772	-276
2017	5	credit_card	2853	-1007
2017	5	debit_card	30	-3
2017	5	voucher	289	-87
2017	6	UPI	707	65
2017	6	credit_card	2463	390
2017	6	debit_card	27	3
2017	6	voucher	239	50
2017	7	UPI	845	-138
2017	7	credit_card	3086	-623
2017	7	debit_card	22	5
2017	7	voucher	364	-125

2017	8	UPI	938	-93
2017	8	credit_card	3284	-198
2017	8	debit_card	34	-12
2017	8	voucher	294	70
2017	9	UPI	903	35
2017	9	credit_card	3283	1
2017	9	debit_card	43	-9
2017	9	voucher	287	7
2017	10	UPI	993	-90
2017	10	credit_card	3524	-241
2017	10	debit_card	52	-9
2017	10	voucher	291	-4
2017	11	UPI	1509	-516
2017	11	credit_card	5897	-2373
2017	11	debit_card	70	-18
2017	11	voucher	387	-96
2017	12	UPI	1160	349
2017	12	credit_card	4377	1520
2017	12	debit_card	64	6
2017	12	voucher	294	93
2018	1	UPI	1518	-358
2018	1	credit_card	5520	-1143
2018	1	debit_card	109	-45
2018	1	voucher	416	-122
2018	2	UPI	1325	193
2018	2	credit_card	5253	267
2018	2	debit_card	69	40
2018	2	voucher	305	111
2018	3	UPI	1352	-27
2018	3	credit_card	5691	-438
2018	3	debit_card	78	-9
2018	3	voucher	391	-86
2018	4	UPI	1287	65
2018	4	credit_card	5455	236
2018	4	debit_card	97	-19
2018	4	voucher	370	21
2018	5	UPI	1263	24
2018	5	credit_card	5497	-42
2018	5	debit_card	51	46
2018	5	voucher	324	46
2018	6	UPI	1100	163
2018	6	credit_card	4813	684
2018	6	debit_card	182	-131
2018	6	voucher	324	0
2018	7	UPI	1229	-129
2018	7	credit_card	4755	58



2018	7	debit_card	242	-60
2018	7	voucher	281	43
2018	8	UPI	1139	90
2018	8	credit_card	4985	-230
2018	8	debit_card	277	-35
2018	8	not_defined	2	
2018	8	voucher	295	-14
2018	9	not_defined	1	1
2018	9	voucher	15	280
2018	10	voucher	4	11

## Summary

### 1. Order Volume by Payment Type:

- **Credit Card** is the dominant payment method, consistently driving the highest order volume across the years.
- **UPI** adoption shows a steady increase, reflecting its growing popularity.
- **Voucher** and **Debit Card** represent smaller shares but remain consistent contributors.

### 2. Month-on-Month Performance:

- **Credit Card** orders exhibit significant fluctuations, peaking in **November 2017** (5,897 orders) and showing seasonal declines thereafter.
- **UPI** orders display steady growth, particularly in 2018, with noticeable peaks in **January 2018** (1,518 orders).
- **Vouchers** show irregular trends, with a significant rise in **June 2018** and a drop afterward.
- **Debit Card** usage remains relatively stable but contributes minimally to overall order counts.

## Actionable Items for Business

### 1. Promote Credit Card and UPI Discounts:

- Given the dominance of **Credit Card** and increasing **UPI** adoption, offer targeted promotions or cashback incentives to retain high-volume users and drive engagement.

### 2. Explore Voucher Potential:

- Assess the effectiveness of vouchers in driving customer engagement and order recovery. Consider offering seasonal or referral-based voucher programs to capitalize on consistent voucher usage.