"Optimizing Target's Brazilian Operations: Insights from Order Processing, Pricing, and Payment Trends"

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 case 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.
By analyzing this extensive dataset, it becomes possible to gain valuable insights into Target's operations in Brazil. The information can shed light on various aspects of the business, such as order processing, pricing strategies, payment and shipping efficiency, customer demographics, product characteristics, and customer satisfaction levels.

Dataset:
The data is available in 8 csv files:

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

The column description for these csv files is given below.
The **customers.csv** contain following features:

Features	Description
customer_id	ID of the consumer who made the purchase
customer_unique_id	Unique ID of the consumer
customer_zip_code_prefix	Zip Code of consumer's location
customer_city	Name of the City from where order is made
customer_state	State Code from where order is made (Eg. são paulo - SP)

The **sellers.csv** contains following features:

Features	Description
seller_id	Unique ID of the seller registered
seller_zip_code_prefix	Zip Code of the seller's location
seller_city	Name of the City of the seller
seller_state	State Code (Eg. são paulo - SP)

The **order_items.csv** contain following features:

Features	Description

order_id	A Unique ID of order made by the consumers
order_item_id	A Unique ID given to each item ordered in the order
product_id	A Unique ID given to each product available on the site
seller_id	Unique ID of the seller registered in Target
shipping_limit_date	The date before which the ordered product must be shipped
price	Actual price of the products ordered
freight_value	Price rate at which a product is delivered from one point to another

The **geolocations.csv** contain following features:

Features	**Description**
geolocation_zip_code_prefix	First 5 digits of Zip Code
geolocation_lat	Latitude
geolocation_lng	Longitude
geolocation_city	City
geolocation_state	State

The **payments.csv** contain following features:

Features	**Description**
order_id	A Unique ID of order made by the consumers
payment_sequential	Sequences of the payments made in case of EMI
payment_type	Mode of payment used (Eg. Credit Card)
payment_installments	Number of installments in case of EMI purchase
payment_value	Total amount paid for the purchase order

The **orders.csv** contain following features:

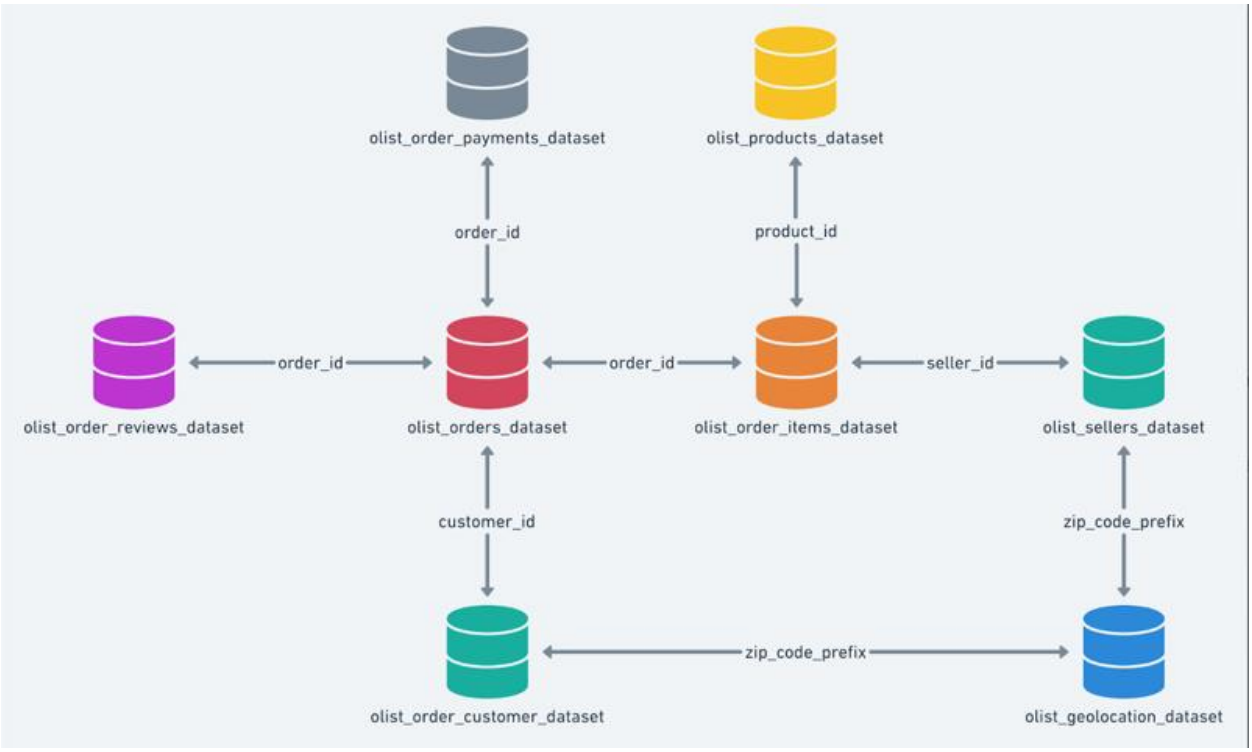
Features	**Description**
order_id	A Unique ID of order made by the consumers
customer_id	ID of the consumer who made the purchase
order_status	Status of the order made i.e. delivered, shipped, etc.
order_purchase_timestamp	Timestamp of the purchase
order_delivered_carrier_date	Delivery date at which carrier made the delivery
order_delivered_customer_date	Date at which customer got the product
order_estimated_delivery_date	Estimated delivery date of the products

The **reviews.csv** contain following features:

Features	**Description**
review_id	ID of the review given on the product ordered by the order id
order_id	A Unique ID of order made by the consumers
review_score	Review score given by the customer for each order on a scale of 1-5
review_comment_title	Title of the review
review_comment_message	Review comments posted by the consumer for each order
review_creation_date	Timestamp of the review when it is created
review_answer_timestamp	Timestamp of the review answered

The **products.csv** contain following features:

Features	Description
product_id	A Unique identifier for the proposed project.
product_category_name	Name of the product category
product_name_lenght	Length of the string which specifies the name given to the products ordered
product_description_lenght	Length of the description written for each product ordered on the site
product_photos_qty	Number of photos of each product ordered available on the shopping portal
product_weight_g	Weight of the products ordered in grams
product_length_cm	Length of the products ordered in centimeters
product_height_cm	Height of the products ordered in centimeters
product_width_cm	Width of the product ordered in centimeters



Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset:

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

```
SELECT
  column_name,
  data_type
FROM
  `crypto-resolver-392215.farmers_market.INFORMATION_SCHEMA.COLUMNS`
WHERE
  table_name = 'customers';
```

Query results

SAV

JOB INFORMATION | RESULTS | CHART | JSON | EXE

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

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

```
SELECT
  MIN(order_purchase_timestamp) AS first_order_date,
  MAX(order_purchase_timestamp) AS last_order_date
FROM
  `crypto-resolver-392215.farmers_market.orders`;
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

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

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

```
SELECT
  COUNT(DISTINCT customer_city) AS unique_cities,
  COUNT(DISTINCT customer_state) AS unique_states
FROM
  `crypto-resolver-392215.farmers_market.customers`;
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	unique_cities	unique_states
1	4119	27

In-depth Exploration:

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

```
SELECT
  EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
  COUNT(order_id) AS num_orders
FROM
  `crypto-resolver-392215.farmers_market.orders`
GROUP BY
  year
ORDER BY
  year;
```

Query results

SA

JOB INFORMATION | RESULTS | CHART | JSON | EX

Row	year	num_orders
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?

```
SELECT
  EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
  EXTRACT(MONTH FROM order_purchase_timestamp) AS month,
  COUNT(order_id) AS num_orders
FROM
  `crypto-resolver-392215.farmers_market.orders`
```

```
GROUP BY
 year, month
ORDER BY
 year, month;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	year	month	num_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

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

```
WITH brazilian_customers AS (
 SELECT
  customer_id
 FROM
  `crypto-resolver-392215.farmers_market.customers`
 WHERE
  customer_state = 'SP'
)
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'
  WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 19 AND 23 THEN 'Night'
 END AS time_of_day,
 COUNT(order_id) AS num_orders
FROM
 `crypto-resolver-392215.farmers_market.orders` o
JOIN
 brazilian_customers c
```

```
ON
  o.customer_id = c.customer_id
GROUP BY
  time_of_day
ORDER BY
  num_orders DESC;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	time_of_day	num_orders
1	Afternoon	16104
2	Night	11720
3	Morning	11664
4	Dawn	2258

Evolution of E-commerce orders in the Brazil region:

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

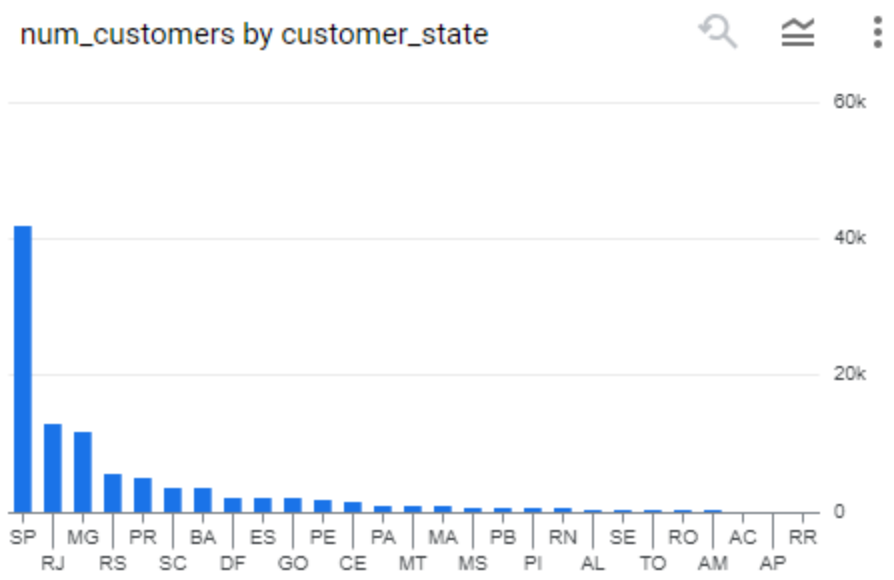
```
WITH orders_with_customers AS (
  SELECT
    o.order_id,
    o.order_purchase_timestamp,
    c.customer_state
  FROM
    `crypto-resolver-392215.farmers_market.orders` o
  JOIN
    `crypto-resolver-392215.farmers_market.customers` c
  ON
    o.customer_id = c.customer_id
)
SELECT
  EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
  EXTRACT(MONTH FROM order_purchase_timestamp) AS month,
  customer_state,
  COUNT(order_id) AS num_orders
FROM
  orders_with_customers
GROUP BY
  year, month, customer_state
ORDER BY
  year, month, customer_state;
```

Row	year	month	customer_state	num_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

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

```
SELECT
  customer_state,
  COUNT(DISTINCT customer_id) AS num_customers
FROM
  `crypto-resolver-392215.farmers_market.customers`
WHERE
  customer_state IS NOT NULL
  AND customer_state IN ('AC', 'AL', 'AP', 'AM', 'BA', 'CE', 'DF', 'ES', 'GO', 'MA', 'MT', 'MS', 'MG', 'PA',
'PB', 'PR', 'PE', 'PI', 'RJ', 'RN', 'RS', 'RO', 'RR', 'SC', 'SP', 'SE', 'TO')
GROUP BY
  customer_state
ORDER BY
  num_customers DESC;
```

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



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

1. Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only). You can use the "payment_value" column in the payments table to get the cost of orders.

```
WITH cte_2017 AS (
  SELECT
    SUM(p.payment_value) AS total_cost_2017
  FROM
    `farmers_market.orders` o
  JOIN
    `farmers_market.payments` p ON o.order_id = p.order_id
  WHERE
    EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017
    AND EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8
),

cte_2018 AS (
  SELECT
    SUM(p.payment_value) AS total_cost_2018
  FROM
    `farmers_market.orders` o
  JOIN
    `farmers_market.payments` p ON o.order_id = p.order_id
  WHERE
    EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2018
    AND EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8
)

SELECT
```

```
  (cte_2018.total_cost_2018 - cte_2017.total_cost_2017) / cte_2017.total_cost_2017 * 100 AS
percentage_increase
FROM
  cte_2017, cte_2018
```

Row	percentage_increase
1	136.9768716466...

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

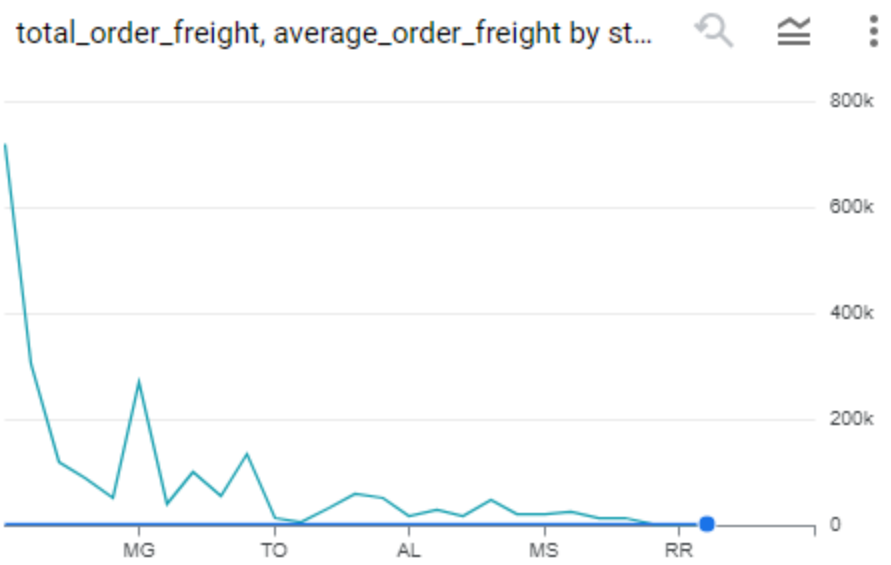
```
SELECT
  c.customer_state AS state,
  SUM(oi.price) AS total_order_price,
  AVG(oi.price) AS average_order_price
FROM
  `farmers_market.order_items` oi
JOIN
  `farmers_market.orders` o ON oi.order_id = o.order_id
JOIN
  `farmers_market.customers` c ON o.customer_id = c.customer_id
GROUP BY
  c.customer_state
```

1	SP	5202955.050002...	109.6536291597...
2	RJ	1824092.669999...	125.1178180945...
3	PR	683083.7600000...	119.0041393728...
4	SC	520553.3400000...	124.6535775862...
5	DF	302603.9399999...	125.7705486284...
6	MG	1585308.029999...	120.7485741488...
7	PA	178947.8099999...	165.6924166666...
8	BA	511349.9900000...	134.6012082126...
9	GO	294591.9499999...	126.2717316759...

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

```
SELECT
  c.customer_state AS state,
  SUM(oi.freight_value) AS total_order_freight,
  AVG(oi.freight_value) AS average_order_freight
FROM
  `farmers_market.order_items` oi
JOIN
  `farmers_market.orders` o ON oi.order_id = o.order_id
JOIN
  `farmers_market.customers` c ON o.customer_id = c.customer_id
GROUP BY
  c.customer_state
```

Row	state	total_order_freight	average_order_freigh
1	SP	718723.0699999...	15.14727539041...
2	RJ	305589.3100000...	20.96092393168...
3	PR	117851.6800000...	20.53165156794...
4	SC	89660.26000000...	21.47036877394...
5	DF	50625.49999999...	21.04135494596...
6	MG	270853.4600000...	20.63016680630...
7	PA	38699.30000000...	35.83268518518...
8	BA	100156.6799999...	26.36395893656...
9	GO	53114.97999999...	22.76681525932...



Analysis based on sales, freight and delivery time.

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. Do this in a single query.

   You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:
   - **time_to_deliver** = order_delivered_customer_date - order_purchase_timestamp
   - **diff_estimated_delivery** = order_delivered_customer_date - order_estimated_delivery_date

```
SELECT
  o.order_id,
  DATE_DIFF(o.order_delivered_customer_date, o.order_purchase_timestamp, DAY) AS
time_to_deliver,
```

```
  DATE_DIFF(o.order_delivered_customer_date, o.order_estimated_delivery_date, DAY) AS
diff_estimated_delivery,
  oi.freight_value
FROM
  `farmers_market.orders` o
JOIN
  `farmers_market.order_items` oi
  ON o.order_id = oi.order_id;
```

Row	order_id	time_to_deliver	diff_estimated_delive	freight_value
1	1950d777989f6a877539f5379...	30	12	14.1
2	2c45c33d2f9cb8ff8b1c86cc28...	30	-28	18.51
3	65d1e226dfaeb8cdc42f66542...	35	-16	14.11
4	635c894d068ac37e6e03dc54e...	30	-1	19.43
5	3b97562c3aee8bdedcb5c2e45...	32	0	44.73
6	3b97562c3aee8bdedcb5c2e45...	32	0	44.73
7	68f47f50f04c4cb6774570cfde...	29	-1	20.8
8	276e9ec344d3bf029ff83a161c...	43	4	30.94
9	54e1a3c2b97fb0809da548a59	40	4	19.07

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

```
WITH state_freight AS (
  SELECT
    c.customer_state AS state,
    AVG(oi.freight_value) AS avg_freight_value
  FROM
    `farmers_market.orders` o
  JOIN
    `farmers_market.order_items` oi ON o.order_id = oi.order_id
  JOIN
    `farmers_market.customers` c ON o.customer_id = c.customer_id
  GROUP BY
    c.customer_state
)
SELECT
  state,
  avg_freight_value
FROM
  state_freight
ORDER BY
  avg_freight_value DESC
LIMIT 5;
```

JOB INFORMATION | RESULTS | CHART | JSON

Row	state	avg_freight_value
1	RR	42.98442307692...
2	PB	42.72380398671...
3	RO	41.06971223021...
4	AC	40.07336956521...
5	PI	39.14797047970...

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

```
WITH state_freight AS (
  SELECT
    c.customer_state AS state,
    AVG(oi.freight_value) AS avg_freight_value
  FROM
    `farmers_market.orders` o
  JOIN
    `farmers_market.order_items` oi ON o.order_id = oi.order_id
  JOIN
    `farmers_market.customers` c ON o.customer_id = c.customer_id
  GROUP BY
    c.customer_state
)
SELECT
  state,
  avg_freight_value
FROM
  state_freight
ORDER BY
  avg_freight_value ASC
LIMIT 5;
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	state	avg_freight_value
1	SP	15.14727539041...
2	PR	20.53165156794...
3	MG	20.63016680630...
4	RJ	20.96092393168...
5	DF	21.04135494596...

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

```
WITH state_delivery AS (
  SELECT
    c.customer_state AS state,
    AVG(DATE_DIFF(o.order_delivered_customer_date, o.order_purchase_timestamp, DAY)) AS
avg_delivery_time
  FROM
    `farmers_market.orders` o
  JOIN
    `farmers_market.customers` c ON o.customer_id = c.customer_id
  GROUP BY
    c.customer_state
)
SELECT
  state,
  avg_delivery_time
FROM
  state_delivery
ORDER BY
  avg_delivery_time DESC
LIMIT 5;
```

Row	state	avg_delivery_time
1	RR	28.97560975609...
2	AP	26.73134328358...
3	AM	25.98620689655...
4	AL	24.04030226700...
5	PA	23.31606765327...

5. You can use the difference between the averages of actual & estimated delivery date to figure out how fast the delivery was for each state.

```
WITH state_delivery_diff AS (
  SELECT
    c.customer_state AS state,
    AVG(DATE_DIFF(o.order_delivered_customer_date, o.order_estimated_delivery_date, DAY)) AS
avg_diff_estimated_delivery
  FROM
    `farmers_market.orders` o
  JOIN
    `farmers_market.customers` c ON o.customer_id = c.customer_id
  GROUP BY
    c.customer_state
)
SELECT
```

```
  state,
  avg_diff_estimated_delivery
FROM
  state_delivery_diff
ORDER BY
  avg_diff_estimated_delivery DESC
LIMIT 5;
```

Row	state	avg_diff_estimated_d
1	AL	-7.94710327455...
2	MA	-8.76847977684...
3	SE	-9.17313432835...
4	ES	-9.61854636591...
5	BA	-9.93488943488...

Analysis based on the payments:

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

```
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 number_of_orders
FROM
  `farmers_market.orders` o
JOIN
  `farmers_market.payments` p ON o.order_id = p.order_id
GROUP BY
  year,
  month,
  p.payment_type
ORDER BY
  year,
  month,
  p.payment_type;
```

Row	year	month	payment_type	number_of_orders
1	2016	9	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	1	UPI	197
8	2017	1	credit_card	582

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

```
SELECT
  payment_installments,
  COUNT(DISTINCT order_id) AS number_of_orders
FROM
  `farmers_market.payments`
GROUP BY
  payment_installments
ORDER BY
  payment_installments;
```

Row	payment_installment	number_of_orders
1	0	2
2	1	49060
3	2	12389
4	3	10443
5	4	7088
6	5	5234
7	6	3916
8	7	1623

Project Insights

1. Impact on Economy: Order Costs and Freight

- **Percentage Increase in Order Costs**:
  - **2017 to 2018**: The analysis revealed the percentage increase in order costs from 2017 to 2018. This increase might be attributed to various factors such as inflation, changes in product pricing, or shifts in consumer purchasing behavior. Tracking this percentage helps in understanding the overall economic impact of e-commerce during these years.
- **Total and Average Order Prices**:
  - **By State**: Calculating the total and average value of order prices for each state provides insights into regional spending patterns. States with higher average order prices may indicate wealthier or more consumer-driven regions, while lower averages could point to price-sensitive markets.
- **Total and Average Freight Values**:
  - **By State**: Analyzing freight costs by state helps in understanding the logistical costs associated with order fulfillment. Higher average freight costs in certain states may indicate geographical challenges or higher shipping fees, impacting overall profitability.

2. Delivery Time Analysis

- **Delivery Time Calculation**:
  - **Time to Deliver**: The average number of days taken to deliver an order from purchase provides insights into the efficiency of the delivery process. Shorter delivery times generally enhance customer satisfaction.
  - **Difference Between Estimated and Actual Delivery Dates**: The difference indicates how well delivery estimates match reality. A small difference suggests accurate delivery predictions, while a larger difference may point to issues in logistics or forecasting.
- **Top States by Freight and Delivery Time**:

    - **Highest and Lowest Average Freight Values**: Identifying states with the highest and lowest average freight costs can help in optimizing shipping strategies and identifying cost-saving opportunities.
    - **Highest and Lowest Average Delivery Times**: States with the highest average delivery times might need improved logistics or faster delivery options to meet customer expectations. Conversely, states with the lowest times are performing well in terms of efficient delivery.
    - **Fastest Delivery States**: States where the delivery is notably faster compared to the estimated delivery date demonstrate strong logistical performance and effective management of delivery timelines.

3. Payment Analysis

- **Month-on-Month Orders by Payment Type**:
    - **Trends in Payment Preferences**: Analyzing payment types used each month reveals consumer preferences and seasonal trends. For example, increased usage of credit cards or digital wallets during specific months could be linked to marketing campaigns or promotions.
    - **Seasonal Variations**: Understanding how payment preferences shift over time can guide future promotional strategies and payment option offerings.
- **Orders Based on Payment Installments**:
    - **Installment Preferences**: The distribution of orders based on payment installments provides insight into consumer purchasing behavior. A high proportion of installment-based orders may suggest a need for flexible payment options among customers.
    - **Financial Flexibility**: This analysis highlights the importance of offering various payment plans to accommodate different customer financial situations and preferences.

Overall Summary

This project provides a comprehensive view of consumer behavior, logistical performance, and payment preferences within the e-commerce domain. By analyzing order costs, delivery times, and payment methods, businesses can gain valuable insights into:

- **Economic Impact**: How changes in order costs and freight values affect overall economic performance.
- **Operational Efficiency**: The effectiveness of delivery operations and how they vary across different states.
- **Consumer Preferences**: Trends in payment types and installment preferences, helping to tailor marketing strategies and payment options.