Case Study SQL

As a committed analyst, I *NISHTHA* has explored this case study comprehensively, diving into Target's operational intricacies as a global retail leader celebrated for value, innovation, and exceptional customer experiences. Alongside addressing core case study questions, I've performed supplementary analysis.

Context:

A renowned global brand, recognized for its prominence in the retail sector in the United States, has established itself as a preferred shopping destination by consistently delivering exceptional value, innovation, and an unparalleled guest experience. The case study delves into the operational aspects of this industry leader in a specific international market, referred to as "Country A." The dataset under examination provides comprehensive insights into a dataset of 100,000 orders spanning from 2016 to 2018. This dataset encompasses a wide array of dimensions, including order statuses, pricing structures, payment and logistics performance, customer locations, product attributes, and customer feedback.

This ***preliminary assessment*** includes an examination of metadata, an evaluation of the customer dataset's dimensions, and the identification of any poten1tial data gaps before starting with case study specific questions.

- **Get metadata for the Target dataset**

Query:

```
# Get metadata for tables in Target dataset
SELECT *
FROM `Target.INFORMATION_SCHEMA.TABLES`;
```

Output:

table_catalog	table_schema	table_name	table_type	is_insertable_i	is_typed	creation_time	base_table_catalog	base_table_schema	base_table_name
scaler-sql-dsml-390...	Target	order_items	BASE TABLE	YES	NO	2023-07-31 01:41:14...	null	null	null
scaler-sql-dsml-390...	Target	sellers	BASE TABLE	YES	NO	2023-07-31 01:50:48...	null	null	null
scaler-sql-dsml-390...	Target	geolocation	BASE TABLE	YES	NO	2023-07-31 01:39:20...	null	null	null

- **How many rows and columns do we have in customer table?**

Query:

```
# Total number of columns in customer table (Similar operation can be performed for other tables)
SELECT count(distinct column_name) as Total_Col
FROM  `Target.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'customers';

# Total number of rows in customer table (Similar operation can be performed for other tables)
SELECT count(*) AS Total_Row
FROM `Target.customers`;
```

Output:

Row	Total_Col
1	5

Row	Total_Row
1	99441

- **Check for missing values in customer table, If any.**

Query:

```
SELECT SUM
(
  CASE WHEN
   customer_zip_code_prefix IS NULL
  Then 1
  ELSE 0
  END
)  AS missingval_zipcodeprefix
FROM `Target.customers`;

SELECT SUM
(
  CASE WHEN
   customer_unique_id IS NULL
  Then 1
  ELSE 0
  END
)  AS missingval_uniqueid
FROM `Target.customers`;
```

Output:

Row	missingval_zipcodep
1	0

Row	missingval_uniqueid
1	0

Important Note

Disclaimer: Handling of Alias Names in GROUP BY Clause

In the presented business case study, the choice was made not to use alias names in the GROUP BY clause, despite the capability of the chosen database management system, Google BigQuery, to recognize alias names in this context. While it's true that BigQuery supports using aliases directly in the GROUP BY clause, a design decision was made to ensure portability and compatibility across various database systems.

The SQL standard and behavior can vary between different database management systems. Although BigQuery does allow the use of aliases in the GROUP BY clause, there are other widely-used database systems that adhere to a more restrictive standard, wherein column aliases are not recognized in the GROUP BY clause. By avoiding the use of alias names in the GROUP BY clause, the queries created for this case study are designed to be functional and optimized in a broader range of database environments.

This approach aligns with the principle of writing SQL queries that are not only efficient in the chosen platform but are also future-proof and adaptable to different systems. It also demonstrates a consideration for potential migration or integration efforts with other databases that may not support the same behavior as BigQuery.

Therefore, the decision to refrain from using alias names in the GROUP BY clause was a strategic one, aimed at creating SQL queries that maintain consistency across various database platforms and enhance the overall robustness of the solution.

Case Study Questions:

Exploratory Data Analysis

PART A. 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.

A.1 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

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

A.2 query:

```
SELECT distinct(FIRST_VALUE(order_purchase_timestamp) OVER(ORDER BY order_purchase_timestamp))
 AS earliest_order_timestamp,
FIRST_VALUE(order_purchase_timestamp) OVER(ORDER BY order_purchase_timestamp DESC)
 AS latest_order_timestamp
FROM `Target.orders`;
```

[Same output can be generated using MIN (order_purchase_timestamp) and MAX (order_purchase_timestamp) as well]

Output:

earliest_order_timestamp	latest_order_timestamp
2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC

Insights: *Orders spans from September 4, 2016, to October 17, 2018 which represents the time range during which orders were made.*

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

A.3 query:

```
SELECT COUNT(distinct(c.customer_city)) CustomerCity_count, COUNT(DISTINCT(c.customer_state)) CustomerState_count
FROM `Target.customers` c
LEFT JOIN `Target.orders` o
ON c.customer_id = o.customer_id
WHERE order_purchase_timestamp IS NOT NULL;
```

Output:

CustomerCity_count	CustomerState_count
4119	27

Insights: *The output indicates that orders were placed by customers hailing from a diverse range of 27 states and 4,119 individual cities, highlighting the broad regional distribution of orders.*

In-Depth Exploration

Part B.

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

B.1 query:

```
WITH CTE AS(
  SELECT FORMAT_TIMESTAMP('%Y-%m', order_purchase_timestamp) as year_month, COUNT(order_id) Count_of_orderplaced
  FROM `Target.orders`
  GROUP BY FORMAT_TIMESTAMP('%Y-%m', order_purchase_timestamp)
)
SELECT year_month, Count_of_orderplaced
FROM CTE
ORDER BY year_month;
```

Output:

Row	year_month	Count_of_orderplace
1	2016-09	4
2	2016-10	324
3	2016-12	1
4	2017-01	800
5	2017-02	1780
6	2017-03	2682
7	2017-04	2404
8	2017-05	3700
9	2017-06	3245
10	2017-07	4026

(In an effort to illustrate diverse querying strategies, I have explored alternative methods to address the same query by extracting year and month components separately and calculating changes in order counts over the years.)

Alternate approach:

```
WITH CTE AS(
   SELECT *,  EXTRACT(Year FROM order_purchase_timestamp) Year_of_orderplaced,
   EXTRACT(Month FROM order_purchase_timestamp) Month_of_orderplaced
   FROM `Target.orders`
)
SELECT *, (Count_of_orderplaced- lag(Count_of_orderplaced) OVER(ORDER BY Year_of_orderplaced, Month_of_orderplaced)) Increase_in_Order
FROM
(
SELECT Year_of_orderplaced, Month_of_orderplaced, COUNT(order_id) Count_of_orderplaced
FROM CTE
GROUP BY  Year_of_orderplaced, Month_of_orderplaced
)
ORDER BY Year_of_orderplaced, Month_of_orderplaced;
```

Row	Year_of_orderplaced	Month_of_orderplace	Count_of_orderplace	Increase_in_Order
1	2016	9	4	*null*
2	2016	10	324	320
3	2016	12	1	-323
4	2017	1	800	799
5	2017	2	1780	980
6	2017	3	2682	902
7	2017	4	2404	-278
8	2017	5	3700	1296
9	2017	6	3245	-455
10	2017	7	4026	781

In this query increase/decrease of no. of order is shown over every month over the past years

Insights: *The examination of monthly order placement trends reveals distinct patterns in customer engagement. In December 2016, there was a single order. In January 2017, there was a remarkable surge of 799 orders, indicating a strong start to the year. February and March 2017 showed substantial increases of 980 and 902 orders, respectively. May 2017 had a significant leap of 1296 orders, while June had a dip in orders. November 2017 saw substantial growth of 2913 orders, emphasizing dynamic seasonal variation. December 2017 experienced a dip of 1871 orders, potentially reflecting a post-holiday slowdown.*

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

B.2 Query:

```
WITH t1 AS(
  SELECT EXTRACT(MONTH FROM order_purchase_timestamp) AS Order_Placed_Month, COUNT(order_id) AS Count_of_Orders
  FROM `Target.orders`
  GROUP BY EXTRACT(MONTH FROM order_purchase_timestamp)
),
t2 AS(
  SELECT EXTRACT(DATE FROM order_purchase_timestamp) AS Order_date,
  EXTRACT(MONTH FROM order_purchase_timestamp) AS Order_Placed_Month
  FROM `Target.orders`
)
SELECT DISTINCT(t1.Order_Placed_Month), FORMAT_DATETIME("%B", Order_date) AS Month, Count_of_Orders
FROM t1, t2
WHERE t1.Order_Placed_Month = t2.Order_Placed_Month
ORDER BY Count_of_Orders DESC;
```

Output:

Row	Order_Placed_Month	Month	Count_of_Orders
1	8	August	10843
2	5	May	10573
3	7	July	10318
4	3	March	9893
5	6	June	9412
6	4	April	9343
7	2	February	8508
8	1	January	8069
9	11	November	7544
10	12	December	5674

Month name column is added for better visibility. If name of Months not needed, only one CTE is enough

Insight: *August, May, and July prominently stand out as peak months for order placements, recording order counts of 10,843, 10,573, and 10,318, respectively. This observed trend aligns with the summer season. Following closely are the months of March, June, and April, each exhibiting a consistent order count. In contrast, the winter months of January and February demonstrate relatively lower order counts of 8,069 and 8,508, respectively. Notably, there is a surge in orders during November, likely indicating heightened demand associated with holiday shopping. However, this trend sees a decline in December, potentially influenced by the conclusion of the holiday season.*

3. During what time of the day, do the country A 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**

B.3 Query:

```
WITH CTE1 AS(
  SELECT DISTINCT(EXTRACT(HOUR FROM order_purchase_timestamp)) AS hours, customer_id, order_id, order_status
  FROM `Target.orders`
  ORDER BY hours
), CTE2 AS(
SELECT
CASE
WHEN hours BETWEEN 0 AND 6
THEN 'Dawn'
WHEN hours BETWEEN 7 AND 12
THEN 'Mornings'
WHEN hours BETWEEN 13 AND 18
THEN 'Afternoon'
ELSE 'Night'
END AS timeofday, order_id
FROM CTE1
),
CTE3 AS( SELECT Timeofday, count(order_id) AS Order_count
    FROM CTE2
    GROUP BY timeofday
    ORDER BY order_count DESC
)
SELECT * FROM CTE3;
```

Output:

Row	Timeofday	Order_count
1	Afternoon	38135
2	Night	28331
3	Mornings	27733
4	Dawn	5242

Insights: *Analysis showed that customers in country A have different ordering preferences depending on the time of day. In particular, the afternoon segment (13:00 - 18:00 hours) is the most popular time slot, with 38,135 orders. The night sector (19:00 - 23:00 hours) followed closely with 28,331 orders. The morning segment (07:00-12:00) and the dawn segment (00:00-06:00) also contributed significantly, recording 27,733 orders respectively and he recorded 5,242 orders.*

Evolution of E-commerce orders in the country A region

Part C

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

C.1 Query:

```
WITH CTE AS(
  SELECT DISTINCT(c.customer_state), EXTRACT(MONTH FROM o.order_purchase_timestamp) AS Month, Count(order_id) AS Order_count
  FROM `Target.customers` c
  INNER JOIN `Target.orders` o
  ON c.customer_id = o.customer_id
  GROUP BY c.customer_state, EXTRACT(MONTH FROM o.order_purchase_timestamp)
)
SELECT *, (Order_count - lag(Order_count) OVER (PARTITION BY customer_state ORDER BY Month)) AS MoM_Order_Change
FROM CTE
ORDER BY customer_state, MONTH;
```

Output:

Row	customer_state	Month	Order_count	MoM_Order_Change
1	AC	1	8	null
2	AC	2	6	-2
3	AC	3	4	-2
4	AC	4	9	5
5	AC	5	10	1
6	AC	6	7	-3
7	AC	7	9	2
8	AC	8	7	-2
9	AC	9	5	-2
10	AC	10	6	1

Insight: *In November, the state 'SP' had a remarkable MoM increase of 1104 orders. There were also substantial growths in March (690 orders), May (665 orders), and August (601 orders). 'MG' saw a significant boost of 343 orders in November, while 'RJ' observed a noteworthy rise of 323 orders during the same period. In July, 'SP' displayed a MoM increase of 277 orders, and in October, there was a noteworthy expansion of 260 orders. The state 'RJ' experienced moderate growth of 186 orders in February.*

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

C.2 Query:

```
SELECT customer_state, COUNT(DISTINCT(customer_id)) AS customer_count,
COUNT(DISTINCT(customer_unique_id)) AS customer_uniqueid_count
FROM `Target.customers`
GROUP BY customer_state
ORDER BY customer_count DESC;
```

Output:

Row	customer_state	customer_count	customer_uniqueid_count
1	SP	41746	40302
2	RJ	12852	12384
3	MG	11635	11259
4	RS	5466	5277
5	PR	5045	4882
6	SC	3637	3534
7	BA	3380	3277
8	DF	2140	2075
9	ES	2033	1964
10	GO	2020	1952

Insight: *Significantly, the state 'SP' boasts the most substantial customer count, totalling 41,746 individuals. This commanding figure is closely pursued by the states of 'RJ' and 'MG,' where customer counts reach 12,852 and 11,635, respectively. The pattern of considerable customer presence persists, with 'RS' hosting 5,466 customers and 'PR' accounting for 5,045. Similarly, the states of 'SC' and 'BA' accommodate 3,637 and 3,380 customers, respectively. This discernible trend further extends to the state of 'DF,' which registers 2,140 customers, while 'ES' and 'GO' mirror each other closely, with 2,033 and 2,020 customers, respectively.*

Recommendations:

1. **State-Specific Marketing Strategies**:
    - Capitalize on the substantial month-on-month increases observed in specific states like 'SP' and 'MG'. Tailor marketing campaigns, promotions, and product launches to coincide with these high-growth periods.
2. **Supply Management Optimization**:
    - Recognize the surge in orders and the subsequent dips in certain states. Adjust inventory and supply chain processes to accommodate these fluctuations effectively.
3. **Localized Seasonal Offers**:
    - Create targeted seasonal campaigns in states like 'RJ' where order count experiences a significant boost during specific months. Offer time-limited promotions and exclusive products to capitalize on heightened interest.
4. **Customer Retention Strategies**:
    - Address the fluctuating order patterns in 'SP' and 'RJ' with customer retention strategies. Implement loyalty programs or personalized discounts to encourage repeat business during months of slower order growth.
5. **Geo-Targeted Content**:
    - Design marketing materials and content that resonate with each state's peak order months. Tailor the messaging to align with local festivities, holidays, and preferences, enhancing customer engagement.
6. **Strategic Marketing in High Customer Count States:**
    - In states like 'SP', which have a high customer count, focus on targeted marketing campaigns to maintain brand loyalty. Implement personalized email campaigns, and loyalty rewards of new products to engage and retain customers.
7. **Segmented Offers in High Customer Count States:**
    - In 'SP', 'RJ', and 'MG', create customer segments based on demographics and purchase history. Craft tailored offers and promotions to cater to distinct preferences and increase average order values.

8. **Regional Partnerships in High Customer Count States:**

    - Establish collaborations with local businesses in states like 'SP' to cross-promote products or services. Leverage the existing customer base for joint marketing efforts and mutually beneficial partnerships.

9. **Localized Content for State Preferences:**

    - Tailor website content and marketing materials for states like 'RS' and 'PR' to resonate with local culture and preferences. Use regional references and imagery to create a more personalized shopping experience.

10. **Customer Feedback Surveys in Low Customer Count States:**

    - In states like 'DF', 'ES', and 'GO', encourage customers to provide feedback on their shopping experience. Use to enhance user experience and address any pain points specific to these regions.

11. **Localized Payment and Delivery Options:**

    - Offer preferred local payment methods and faster delivery options in states with lower customer counts like 'BA' and 'SC'. Enhance the checkout process to provide a seamless experience that aligns with customer expectations.

12. **Regional Influencer Collaborations:**

    - Collaborate with local influencers in states with varying customer counts to promote products. Influencers can help create localized content and connect with their followers in these regions.

13. **Community Building in Low Customer Count States:**

    - In states with lower customer counts like 'SC' and 'BA', focus on community-building efforts. Create online forums, social media groups, or events that foster engagement and bring customers together.

14. **Collaborative Discounts in Low Customer Count States:**

    - Offer joint discounts with complementary businesses in states with lower customer counts. For example, partner with local restaurants or entertainment venues to provide added value to customers in 'ES' and 'GO'.

Impact on Economy

Part D. Analyse 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).

D.1 Query:

```
WITH t1 AS(
SELECT EXTRACT(YEAR FROM order_purchase_timestamp) AS purchase_year, EXTRACT(MONTH FROM order_purchase_timestamp) AS purchase_month, SUM(payment_value) AS payment_sum
FROM `Target.orders` o
  INNER JOIN `Target.payments` p
  ON o.order_id = p.order_id
GROUP BY EXTRACT(YEAR FROM order_purchase_timestamp), EXTRACT(MONTH FROM order_purchase_timestamp)
),
t2 AS(
SELECT distinct(purchase_year), SUM(payment_sum) OVER(PARTITION BY purchase_year) as Total_cost
FROM t1
WHERE purchase_year BETWEEN 2017 AND 2018
  AND purchase_month BETWEEN 1 AND 8
ORDER BY purchase_year
)
SELECT purchase_year, IFNULL((((Total_cost- lag(Total_cost) OVER(ORDER BY purchase_year)) / lag(Total_cost) OVER(ORDER BY purchase_year) ) *100),0) AS percentage_cost_increase
FROM t2;
```

Output:

Row	purchase_year	percentage_cost_increase
1	2017	0.0
2	2018	136.97687164666064

Insight: *The analysis of order cost increase from 2017 to 2018 (spanning January to August) unveils a significant growth trend. The year 2018 experienced an impressive surge of approximately 136.98% in order costs. This remarkable increase highlights the substantial financial growth over the specified time period, reflecting a substantial uptick in order values and potentially indicating shifts in customer purchasing behaviour or business strategies.*

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

D.2 Query:

```
SELECT DISTINCT(c.customer_state), ROUND(SUM(oi.price) OVER(PARTITION BY c.customer_state),2) AS Total_value,
ROUND(AVG(oi.price) OVER(PARTITION BY c.customer_state),2) AS Average_value
FROM `Target.customers` c
LEFT JOIN `Target.orders` o
ON c.customer_id = o.customer_id
LEFT JOIN `Target.order_items` oi
ON o.order_id = oi.order_id
ORDER BY c.customer_state;
```

Output:

Row	customer_state	Total_value	Average_value
1	AC	15982.95	173.73
2	AL	80314.81	180.89
3	AM	22356.84	135.5
4	AP	13474.3	164.32
5	BA	511349.99	134.6
6	CE	227254.71	153.76
7	DF	302603.94	125.77
8	ES	275037.31	121.91
9	GO	294591.95	126.27
10	MA	119648.22	145.2

Insight: *The variation in total and average order values across states reflects differing levels of economic activity and consumer spending. States with higher total and average values, such as 'BA' and 'CE', suggest potentially stronger purchasing power and a willingness to spend on higher-value items. States with lower average order values, like 'AM' and 'AP', might represent untapped market potential. Strategies that tailor offerings to suit local preferences and budgets could potentially yield growth opportunities.*

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

D.3 Query:

```
SELECT DISTINCT(c.customer_state), ROUND(SUM(oi.freight_value) OVER(PARTITION BY c.customer_state),2) AS Total_value,
 ROUND(AVG(oi.freight_value) OVER(PARTITION BY c.customer_state),2) AS Average_value
FROM `Target.customers` c
INNER JOIN `Target.orders` o
ON c.customer_id = o.customer_id
INNER JOIN `Target.order_items` oi
ON o.order_id = oi.order_id
ORDER BY c.customer_state;
```

Output:

Row	customer_state	Total_value	Average_value
1	AC	3686.75	40.07
2	AL	15914.59	35.84
3	AM	5478.89	33.21
4	AP	2788.5	34.01
5	BA	100156.68	26.36
6	CE	48351.59	32.71
7	DF	50625.5	21.04
8	ES	49764.6	22.06
9	GO	53114.98	22.77
10	MA	31523.77	38.26

Insight: *The variation in total and average freight values across states signifies the varying costs of transportation and delivery services. States with higher average freight values, such as 'AC' and 'MA', might indicate logistical challenges or remote locations that incur higher shipping costs. The lower average freight values in states like 'BA' and 'CE' suggest that customers in these regions might be more sensitive to shipping costs. This could reflect preferences for affordable or locally available products. : Higher freight values in certain states, such as 'DF' and 'ES', could be due to their proximity to major distribution centres or efficient transportation networks.*

Recommendations:

1. **Price Optimization for High-Performing Months:**
    - Capitalize on the substantial increase in order costs observed in November 2018 (136.98% rise) compared to 2017. Introduce exclusive or higher-priced products during this peak season to capture the willingness of customers to spend more.
2. **Targeted Marketing for High-Value States:**
    - Focus marketing efforts on states like 'SP' and 'RJ', which exhibit high average order values. Design campaigns that highlight premium offerings and emphasize quality and value to resonate with these states' preferences.
3. **Shipping Cost Subsidies in High Freight States:**
    - Address the relatively higher freight costs observed in states like 'DF' and 'ES' with targeted shipping cost subsidies or discounts. Implement promotional periods offering free or reduced-cost shipping to incentivize purchases in these states.
4. **Localized Product Assortment:**
    - Leverage insights on customer distribution and average order values to tailor product assortments for each state. Curate products that align with the preferences of customers in different states, enhancing relevance and driving conversions.

5. **Supply Chain Efficiency in High Freight States:**

    - Strategically place warehouses or distribution centers in states with higher average freight values to reduce shipping costs. Optimize inventory management and fulfillment processes to minimize delivery expenses.

6. **Customer Engagement in States with Low Average Values:**

    - Implement loyalty programs or targeted promotions in states like 'AM' and 'AP', which exhibit lower average order values. Encourage repeat purchases and larger basket sizes through incentives that resonate with budget-conscious customers.

7. **Localized Seasonal Campaigns:**

    - Capitalize on regional seasonal trends observed in states like 'BA' and 'CE' by tailoring promotions to align with their specific peak shopping periods. Create marketing content that resonates with the festive and buying behavior of these states.

8. **Cross-Selling and Up-Selling in High-Value States:**

    - Implement cross-selling and up-selling strategies in states with higher average order values and suggest complementary products or upgrades during the purchase process to maximize revenue per transaction.

9. **Local Partnerships for Freight Optimization:**

    - Collaborate with local shipping partners in regions with higher freight costs to negotiate better rates and enhance delivery efficiency. Offer premium shipping options for faster deliveries in high-value states to cater to customer preferences.

Analysis based on sales, freight and delivery time

PART E.

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.

E.1 Query:

```
SELECT order_id, TIMESTAMP_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) AS Time_to_deliver,
 TIMESTAMP_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY) AS Diff_estimated_delivery
FROM `Target.orders`;
```

Output:

Row	order_id	Time_to_deliver	Diff_estimated_delivery
1	1950d777989f6a877539f5379...	30	-12
2	2c45c33d2f9cb8ff8b1c86cc28...	30	28
3	65d1e226dfaeb8cdc42f66542...	35	16
4	635c894d068ac37e6e03dc54e...	30	1
5	3b97562c3aee8bdedcb5c2e45...	32	0
6	68f47f50f04c4cb6774570cfde...	29	1
7	276e9ec344d3bf029ff83a161c...	43	-4
8	54e1a3c2b97fb0809da548a59...	40	-4
9	fd04fa4105ee8045f6a0139ca5...	37	-1
10	302bb8109d097a9fc6e9cefc5...	33	-5

Insights: *The analysis of order delivery metrics provides valuable insights into the efficiency and accuracy of the delivery process. By calculating the time taken to deliver each order from the purchase date, businesses can gauge their operational performance in terms of order fulfilment.*

The "Time_to_deliver" column highlights the number of days taken to deliver each order from the date of purchase. This metric is crucial for evaluating the business's ability to fulfill orders promptly. For instance, orders with varying delivery times, such as 30, 35, or 29 days, signify discrepancies in delivery speed. the "Diff_estimated_delivery" column showcases the difference (in days) between the estimated and actual delivery dates. Negative values, such as -12, -4, or -5, indicate instances where orders were delivered before the estimated delivery date. Conversely, positive values like 28 or 16 denote orders that were delivered after the initially estimated timeframe.

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

E.2 Query:

```
WITH CTE AS(
SELECT c.customer_state, AVG(oi.freight_value) AS Average_freight
FROM `Target.customers` c
INNER JOIN `Target.orders` o
ON c.customer_id = o.customer_id
INNER JOIN `Target.order_items` oi
ON o.order_id = oi.order_id
GROUP BY c.customer_state
ORDER BY c.customer_state
),
 High AS(
  SELECT customer_state, dense_rank() OVER(ORDER BY CTE.Average_freight ASC) AS Highest_freight,
FROM CTE
),
Low AS(
  SELECT customer_state, dense_rank() OVER(ORDER BY CTE.Average_freight DESC) AS Lowest_freight,
FROM CTE
)
SELECT Highest_freight AS Sno, h.customer_state AS Top5State, l.customer_state AS Bottom5State
FROM HIGH h
INNER JOIN LOW l
ON h.Highest_freight = l.Lowest_freight
where  h.Highest_freight<=5 and l.Lowest_freight<=5
ORDER BY h.Highest_freight;
```

Output:

Row	Sno	Top5State	Bottom5State
1	1	SP	RR
2	2	PR	PB
3	3	MG	RO
4	4	RJ	AC
5	5	DF	PI

Insights: *The analysis of average freight values across states reveals notable differences. Among the top 5 states with the lowest average freight values, 'SP' stands out as the leader, indicating cost-efficient shipping operations. On the contrary, 'RR', 'PB', and 'RO' show higher average freight values, suggesting potential areas for optimization in their logistical networks. The disparities in freight costs could indicate differences in geographic location, transportation infrastructure, and distance from distribution centers. This insight could prompt the company to optimize its supply chain and distribution strategies for higher-cost regions. The states with lower freight costs might indicate better supply chain efficiency or proximity to distribution centers. The company could investigate replicating these efficiency factors in states with higher freight costs to reduce operational expenses.*

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

E.3 Query:

```
WITH CTE AS(
  SELECT c.customer_state, AVG(TIMESTAMP_DIFF(o.order_delivered_customer_date, o.order_purchase_timestamp, DAY)) AS Average_delivery_time
  FROM `Target.customers` c
  INNER JOIN `Target.orders` o
  ON o.customer_id = c.customer_id
  GROUP BY c.customer_state
  order by Average_delivery_time
),
High AS(
  SELECT customer_state, DENSE_RANK() OVER(ORDER BY Average_delivery_time ASC) AS Highest_delivery_time
  FROM CTE
),
Low AS(
  SELECT customer_state, DENSE_RANK() OVER(ORDER BY Average_delivery_time DESC) AS Lowest_delivery_time
  FROM CTE
)
SELECT Highest_delivery_time AS Sno, h.customer_state AS Top5State, l.customer_state AS Bottom5State
FROM HIGH h
INNER JOIN LOW l
ON h.Highest_delivery_time = l.Lowest_delivery_time
where  h.Highest_delivery_time<=5 and l.Lowest_delivery_time<=5
ORDER BY h.Highest_delivery_time;
```

Output:

Row	Sno	Top5State	Bottom5State
1	1	SP	RR
2	2	PR	AP
3	3	MG	AM
4	4	DF	AL
5	5	SC	PA

Insights: *The analysis of average delivery times across states reveals significant variations. Among the top 5 states with the lowest average delivery times, 'SP' exhibits the quickest deliveries, reflecting efficient logistics and a well-established delivery network. Conversely, 'RR', 'AP', and 'AM' display longer average delivery times, indicating potential areas for optimization to enhance speed and reliability.*

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

E.4 Query:

```
WITH CTE AS
(
  SELECT c.customer_state,  AVG(TIMESTAMP_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY)) AS delivery_time,
  DENSE_RANK() OVER(ORDER BY AVG(TIMESTAMP_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY))) AS Ranking
  FROM `Target.customers` c
  INNER JOIN `Target.orders` o
  ON o.customer_id = c.customer_id
  WHERE o.order_status = 'delivered' OR o.order_delivered_customer_date IS NOT NULL
  GROUP BY c.customer_state
  ORDER BY delivery_time ASC
) SELECT customer_state AS Top5State, delivery_time, Ranking
FROM CTE
WHERE Ranking<=5;
```

Output:

Row	Top5State	delivery_time	Ranking
1	AL	7.9471032745592	1
2	MA	8.768479776847...	2
3	SE	9.173134328358...	3
4	ES	9.618546365914...	4
5	BA	9.934889434889...	5

Insights: *The analysis of average delivery times across states highlights states with efficient delivery performance. States such as 'AL', 'MA', and 'SE' demonstrate remarkably quick average delivery times, with delivery times as low as 7.95 days. This suggests that these regions have well-optimized logistical operations and effective coordination, resulting in prompt order deliveries. Conversely, 'ES' and 'BA' also showcase competitive average delivery times, further reflecting the effectiveness of their delivery systems.*

Recommendations

1. **Optimize Delivery Time Management:**
    - Analyze the calculated delivery times for each order to identify outliers and patterns.
    - Implement strategies to reduce delivery times, such as optimizing warehouse locations, streamlining logistics, and enhancing last-mile delivery efficiency.
2. **Enhance Freight Value Management:**
    - Focus on states with higher average freight values, such as 'RR', 'PB', 'RO', 'AC', and 'PI', to identify cost-saving opportunities.
    - Explore options for negotiated shipping rates, partnerships with local carriers, or route optimization to lower shipping costs.
3. **Improve Delivery Time in Certain States:**
    - In states with higher average delivery times, like 'AL', 'MA', and 'SE', consider enhancing distribution networks and partnering with reliable carriers. Implement real-time tracking and updates to manage customer expectations and offer accurate delivery estimates.
4. **Fast Delivery States for Targeted Expansion:**
    - Identify states like 'DF', 'ES', and 'MG' with faster-than-estimated delivery times, indicating operational excellence.
    - Leverage these successful models to optimize operations in other states and potentially expand operations.
5. **Segmented Approach to Customer Engagement:**
    - Tailor marketing efforts based on states with higher and lower average freight values and delivery times.
    - For states with higher freight values, emphasize value-added offerings or incentives to mitigate potential cart abandonment due to shipping costs.
6. **Customer Feedback for Improvement:**
    - Collect feedback from customers in states with longer delivery times to understand pain points and areas for improvement.
    - Use feedback to refine processes and enhance customer satisfaction.

7. **Localized Strategies for Regional Preferences:**
    - Customize marketing campaigns and product assortments based on regional preferences identified from the analysis.
    - Offer promotions or products that align with each state's unique customer behavior.
8. **Efficiency Sharing Among States:**
    - States with efficient delivery times, like 'SP', can share best practices and insights with states that require optimization.
    - Foster a culture of continuous improvement and knowledge sharing.
9. **Data-Driven Insights for Decision-Making:**
    - Continuously monitor and analyse delivery times, freight values, and customer feedback to make informed decisions.
    - Use data-driven insights to adjust strategies, allocate resources, and enhance overall operational efficiency.
10. **Investment in Technology and Automation:**
    - Consider investing in advanced tracking systems, route optimization software, and automation to streamline processes and reduce delivery times

Analysis based on the payments

Part F

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

F.1 Query:

```
WITH CTE AS(
SELECT EXTRACT(Year from o.order_purchase_timestamp) AS Year, EXTRACT(Month from o.order_purchase_timestamp) AS Month,
COUNT(o.order_id) AS Order_count, p.payment_type AS Payment_type
FROM `Target.orders` o
INNER JOIN `Target.payments` p
ON o.order_id = p.order_id
GROUP BY EXTRACT(Year from o.order_purchase_timestamp), EXTRACT(Month from o.order_purchase_timestamp), p.payment_type
)
SELECT *
FROM CTE
ORDER BY Year, Month, Payment_type;
```

Output:

Row	Year	Month	Order_count	Payment_type
3	2016	10	254	credit_card
4	2016	10	2	debit_card
5	2016	10	23	voucher
6	2016	12	1	credit_card
7	2017	1	197	UPI
8	2017	1	583	credit_card
9	2017	1	9	debit_card
10	2017	1	61	voucher

Insights: *The analysis of month-on-month order placement, categorized by different payment types, reveals distinct trends in customer preferences over time. For example, in September 2016, only 3 orders were placed using credit cards. However, in the subsequent month of October 2016, UPI transactions surged to 63 orders, while credit card payments reached 254 orders, indicating a notable shift. Additionally, debit card and voucher payments contributed to the mix, reflecting diverse payment behaviours.*

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

F.6 Query:

```
SELECT payment_installments, count(DISTINCT(order_id)) AS Order_count
FROM `Target.payments`
WHERE payment_sequential>=1
GROUP BY payment_installments
HAVING payment_installments >=1;
```

Output:

Row	payment_installments	Order_count
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: *The data shows that most customers (49,060 orders) prefer a single payment installment, indicating a convenience or financial consideration. The number of orders decreases as the number of installments increases, suggesting a decreasing preference for more installments.*

Recommendations

Month-on-Month Orders by Payment Types:

1. **Capitalizing on UPI Usage:** Given the steady increase in UPI usage from 2016 to 2017, consider partnering with UPI providers to offer exclusive discounts or cashback for customers using this payment method. Highlight the convenience and security of UPI transactions.
2. **Credit Card Consistency:** Credit card orders show consistency across the years. Focus on maintaining this trend by continuously enhancing the credit card payment experience, ensuring swift processing, and reinforcing the security measures.
3. **Voucher Utilization:** The consistent use of vouchers could suggest customer interest in promotional campaigns. Plan targeted voucher-based promotions aligned with key shopping seasons or events to further engage customers.
4. **Debit Card Boost:** Although debit card orders are comparatively lower, consider launching debit card-specific promotions to encourage more customers to use this payment method. These could include discounts or cashback on purchases.

Orders by Payment Installments:

1. **Leveraging Single Installment Dominance:** Since the majority of customers opt for a single payment installment, ensure that this option is displayed prominently during the checkout process. Highlight its convenience and simplicity.
2. **Encouraging 2-3 Installments:** There is a considerable drop in orders beyond 1 or 2 instalments. Offer targeted promotions for 2-3 instalment plans, emphasizing the balance between flexibility and ease of payment.
3. **Discounts for Higher Installments:** For installment counts beyond 3, introduce progressive discounts to incentivize customers to choose more installments. The data suggests an opportunity to make longer-term payment plans more appealing.
4. **Limited-time Offer for High Installments:** For higher installment counts like 10 or more, create a limited-time campaign offering exclusive benefits such as premium customer service, priority shipping, or access to loyalty programs.
5. **Monitoring High Installments:** Keep an eye on installment counts like 11 and above. If these numbers continue to grow, consider investigating why customers prefer such lengthy payment plans and whether adjustments are needed.

Conclusion

In conclusion, this comprehensive exploration of the business operations in country A through meticulous SQL analysis has unveiled a wealth of insights that contribute to a holistic understanding of the business landscape. The amalgamation of structured queries, insightful outputs, and additional analysis showcases a nuanced view of various facets, ranging from customer behaviour and payment preferences to order trends and regional dynamics. By delving beyond the provided case study questions and venturing into supplementary analysis, we have enriched our perspective, allowing us to craft actionable recommendations that align with business strategic goals. This analytical journey underscores the power of data-driven decision-making and its potential to drive growth, enhance operational efficiency, and elevate customer satisfaction. As we wrap up this exploration, the synthesized insights and actionable takeaways stand poised to guide the business endeavours in the A market and beyond.