**Target**

**Gist of the case study**

Target case study analyzes its Brazilian e-commerce business from 2016 to 2018. The data includes details on customers, orders, products, payments, and reviews.

Analysts aim to understand customer behaviour, pricing, efficiency, and areas for improvement. Exploration will focus on order trends, seasonality, delivery times, and payment methods to help Target optimize their Brazilian e-commerce strategy through data-driven decisions.

1. **Import the dataset and do the usual exploratory analysis steps like checking the structure & characteristics of the dataset:**
   1. Data type of all columns in the “customers” table.
   2. Get the time range between which the orders were placed.
   3. Count the Cities & States of customers who ordered during the given period.

1. Question: Can you explain what INFORMATION\_SCHEMA.COLUMNS is and how it can be useful?

Answer: The INFORMATION\_SCHEMA.COLUMNS view provides information about the columns within a specific table in a BigQuery dataset. It retrieves details like column names and data types without directly querying the table itself.

2. Question: Can you explain the purpose of the DISTINCT keyword and the INNER JOIN in this query?

Answer: - The DISTINCT keyword ensures we count each unique city and state only once. The INNER JOIN combines data from both the orders and customers tables based on the matching customer\_id. This allows us to efficiently retrieve city and state information for customers who placed orders during the given period.

3. Question: This query counts distinct cities and states for customers who placed orders. How would you modify it to find the top 5 most frequent cities or states?

Answer: We can use the ORDER BY and LIMIT clauses. For example, to find the top 5 most frequent cities, we can order by city\_cnt descending and limit the results to 5.

Follow-up: Right now, the query counts distinct cities and states. What if you wanted to count the total number of customers (including duplicates) from each city or state?

Answer: We can remove the DISTINCT keyword from the COUNT function. This will count all occurrences of each city and state, potentially including customers who placed multiple orders.

4. Question: Why is it important to know the time range of the orders?

Answer: Knowing the time range helps in understanding the period over which data is collected, which is crucial for trend analysis, seasonality studies, and ensuring data completeness.

1. **In-depth Exploration:**
   1. Is there a growing trend in the no. of orders placed over the past years?
   2. Can we see some kind of monthly seasonality in terms of the no. of orders being placed?
   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

5. Question: Why did you choose to group by both year and month rather than just year?

Answer: Grouping by both year and month provides more granular data, allowing us to observe not only annual trends but also monthly variations, which can reveal seasonal effects and more detailed growth patterns.

6. Question: What are some limitations of using monthly data for trend analysis? Would daily or even hourly data provide more insights?

Answer: Monthly data provides a high-level view, but it might miss shorter-term trends or specific seasonal variations within months. Daily or even hourly data could reveal more granular trends, but it would also increase the complexity of the analysis and visualization.

7. Question: This query investigates potential monthly seasonality in order placement. Can you think of other ways to visualize seasonality besides just counting orders by month?

Answer: Yes, we could calculate the monthly average number of orders can also provide a more nuanced view of seasonality, potentially revealing underlying trends beyond just raw counts.

8. Question: What kind of patterns would indicate monthly seasonality?

Answer: Repeating peaks and troughs in the number of orders for the same months across different years would indicate seasonality.

9. Question: How would you adjust your analysis if you wanted to account for different time zones within Brazil?

Answer: To account for different time zones, you could convert all timestamps to a common time zone (e.g., UTC) before categorizing them into time-of-day intervals.

10. Question: Could there be alternative ways to segment the time of day, and why did you not use them?

Answer: An alternative approach could be segmenting the day into finer intervals (e.g., every hour). However, using broader intervals like dawn, morning, afternoon, and night provides a more general understanding that is easier to interpret and communicate.

1. **Evolution of E-commerce orders in the Brazil region:** 
   1. Get the month-on-month no. of orders placed in each state.
   2. How are the customers distributed across all the states?

11. Question: Why this method (INNER JOIN) used & Why not (Left / Right) Joins?

Answer: An INNER JOIN is used here because we're only interested in orders that have corresponding customer information in the customers table. This ensures we only analyze orders from valid customers with a state associated.

Alternatives (LEFT/RIGHT JOIN):

Answer: A LEFT JOIN could be used if we want to include states with no orders in the results. However, for this specific analysis, focusing on states with actual orders is more relevant.

Answer: A RIGHT JOIN wouldn't be suitable here as we're primarily interested in orders data and want to filter customers based on having placed orders.

1. Follow-up Question: Can you explain a scenario where using a LEFT JOIN would be more appropriate than an INNER JOIN?

Answer: A LEFT JOIN would be more appropriate if we need to include all records from the left table (e.g., customers) and only the matching records from the right table (e.g., orders). This is useful for identifying customers who have not placed any orders, as it will return all customers and NULLs for those without matching orders.

1. Follow-up Question: What is the difference between a LEFT JOIN and a RIGHT JOIN, and when would you use each?

Answer: A LEFT JOIN returns all records from the left table and the matched records from the right table. If no match is found, NULL values are returned for columns from the right table. A RIGHT JOIN, conversely, returns all records from the right table and the matched records from the left table, with NULLs for non-matching records from the left table. The choice between the two depends on which table's records need to be fully included.

3. Follow-up Question: How would you handle a situation where you need to include all records from both tables, regardless of whether there are matches or not?

Answer: To include all records from both tables regardless of matches, you would use a FULL OUTER JOIN. This join returns all records when there is a match in either left or right table records. When there is no match, the result is NULL from the side of the table without a match.

12. Question: Why COUNT() & Why not AVG()?

Answer: COUNT() is used here to determine the total number of orders placed in each state for each month. This provides a basic understanding of the volume of orders.

AVG() could be used to calculate the average number of orders per month for each state. This might be helpful if order volume fluctuates significantly within a state. However, raw counts can be easier to interpret for initial analysis.

Follow-up: This query uses a COUNT(DISTINCT(customer\_unique\_id)) to find the number of customers in each state. Why is the DISTINCT keyword necessary here?

Answer: The DISTINCT keyword is used to ensure we count each customer only once, even if they placed multiple orders. Without DISTINCT, the query would count each order from a customer, potentially inflating the number of customers in each state.

13. Question: What insights can you derive from the customer distribution data?

Answer: This data can highlight the states with the highest and lowest number of customers, helping to identify key markets and potential areas for growth.

1. **Impact on the 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 the 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.

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

14. Question: What insights can be drawn from the percentage increase in the cost of orders?

Answer: This can indicate economic trends, consumer behaviour changes, and potential inflation or increased spending in e-commerce.

15. Question: Why did you use a CTE (Common Table Expression) instead of subqueries or temporary tables?

Answer: CTEs make the query more readable and maintainable by breaking it into logical steps. They are especially useful for complex transformations and aggregations. We could potentially achieve the same result using self-joins or subqueries, but it might be less efficient and more error-prone.

16. Question: Why did you use the LEAD() window function in your query?

Answer: The LEAD() function allows us to access the cost of the next year within the same result set, which simplifies the calculation of the percentage increase.

Follow-up Question: What is the difference between LEAD() and LAG() functions?

Answer: LEAD() retrieves data from the following row, whereas LAG() retrieves data from the preceding row. Both are useful for comparing current row values with previous or subsequent row values within the same result set.

Follow-up Question: Provide a scenario where the LAG() function would be more appropriate than LEAD().

Answer: LAG() would be more appropriate when you need to compare current values with previous values. For instance, if you want to calculate the monthly change in sales, you could use LAG() to get the previous month's sales and subtract it from the current month's sales.

Follow-up Question: Explain the purpose of the PARTITION BY clause in window functions.

Answer: The PARTITION BY clause divides the result set into partitions to which the window function is applied independently. It’s like a GROUP BY for window functions, allowing you to compute values within subsets of your data.

17. Question: How would you modify this query to calculate the increase for the entire year (2017 vs. 2018) instead of just Jan-Aug?

Answer: We could remove the AND EXTRACT(MONTH FROM a.order\_purchase\_timestamp) BETWEEN 1 AND 8 condition in the base\_1 CTE to include all months in 2017 and 2018.

18. Question: Why did you choose INNER JOIN instead of LEFT JOIN?

Answer: INNER JOIN is used to ensure that we only consider orders that have both associated order items and customer information. LEFT JOIN would include orders even if they lack order items or customer data, which could skew the results.

19. Question: How can variations in average freight cost per state impact e-commerce businesses?

Answer: Variations in freight cost can impact shipping strategies, pricing models, and overall customer satisfaction. Higher freight costs might deter customers, while lower costs could increase competitiveness.

20. Question: Why did you use COUNT(DISTINCT o.order\_id) instead of just COUNT(o.order\_id)?

Answer: Using COUNT(DISTINCT o.order\_id) ensures that each order is only counted once. Not using DISTINCT could result in double-counting orders if there are multiple items per order, leading to inaccurate average price calculations.

Follow-up: This query calculates the total and average order price for each state. It uses a CTE to join relevant tables and group data by state. Can you explain how the DISTINCT keyword is used in this context?

Answer: The DISTINCT keyword is used within COUNT(DISTINCT(o.order\_id)) to count the number of unique orders for each state. This ensures we don't inflate the order count if a customer placed multiple orders in the same state.

21. Question: What additional insights can we gain from analyzing the average order price along with total price?

Answer: Average order price can indicate the typical spending pattern for each state. By comparing total price and average price, we can identify states with high overall sales volume but potentially lower average order values (e.g., higher volume of smaller orders).

1. **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\_timestam.
* **diff\_estimated\_delivery** = order\_estimated\_delivery\_date - order\_delivered\_customer\_date
  1. Find out the top 5 states with the highest & lowest average freight value.
  2. Find out the top 5 states with the highest & lowest average delivery time.
  3. Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery.

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

22. Question: What insights can be drawn from analyzing the difference between the estimated and actual delivery dates?

Answer: This can provide insights into the accuracy of delivery estimates, customer satisfaction, and potential areas for improving logistics and delivery processes.

23. Question: Why did you use TIMESTAMP\_DIFF instead of DATE\_DIFF?

Answer: TIMESTAMP\_DIFF is more versatile as it can handle both timestamp and date data types, making it suitable for cases where the columns may include time components.

24. Question: What would be the impact if you didn't filter by order\_status='delivered'?

Answer: Including orders that are not delivered would result in inaccurate calculations of delivery time and differences, as undelivered orders do not have a delivery date.

25. Question: How can analyzing average freight value by state be helpful for business decisions?

Answer: Identifying states with high average freight costs might indicate areas where optimizing shipping strategies or negotiating with carriers could be beneficial.

Follow-up: Why is it important to analyze average freight values by state?

Answer: This can provide insights into regional shipping costs, help optimize logistics, and influence pricing strategies to balance cost and customer satisfaction.

26. Question: Why did you use INNER JOIN for this query?

Answer: INNER JOIN ensures we only include orders with matching customer and order item data, providing accurate and complete information for the calculation.

27. Question: What can high or low average delivery times indicate about the states?

Answer: High average delivery times might indicate logistical challenges or inefficiencies, while low average delivery times could indicate efficient delivery processes and better infrastructure.

28. Question: Why did you use AVG() for calculating delivery times?

Answer: AVG() provides the average number of days taken to deliver orders, which helps compare the efficiency of delivery processes across different states.

29. Question: Why did you include a WHERE clause for order\_status='delivered'?

Answer: Filtering by delivered orders ensures we only analyze completed deliveries, providing accurate delivery time calculations.

30. Question: Why did you choose to order by the difference between actual and estimated delivery times?

Answer: Ordering by this difference highlights the states where the actual delivery time is significantly better than the estimated time, which is crucial for understanding and replicating efficient delivery practices.

31. Question: This query calculates the average actual and estimated delivery times and then subtracts them to identify states with the "fastest" deliveries relative to estimates. Can you explain the logic behind this approach?

Answer: The query assumes that a positive difference between average actual and estimated delivery times indicates faster deliveries compared to estimates. In other words, states with a higher positive difference might have a tendency to deliver earlier than estimated.

1. **Analysis based on the payments:**
   1. Find the month-on-month no. of orders placed using different payment types**.**
   2. Find the no. of orders placed on the basis of the payment installments that have been paid.z

32. Question: This query looks at the number of orders placed each month categorized by payment type. It joins the payments and orders tables. Can you explain how this helps analyze payment trends?

Answer: By grouping orders by payment type, month, and year, we can see how the popularity of different payment methods (e.g., credit card, debit card, UPI and installments) varies throughout the year. This might reveal seasonal trends or preferences for specific payment methods.

33. Question: How would you visualize the results to compare payment trends across different payment types?

Answer: We could use charts like stacked bar charts or line charts to visualize the number of orders for each payment type over time. This would allow for easy comparison and identification of trends.

34. Question: Why did you use inner join & which method will you use to get the same result without using join?

Answer: A JOIN is necessary here to connect order data with payment information. The payments table likely has the order\_id to link it to the corresponding order details in the orders table. It would be difficult to analyze payment types without joining the tables. We could potentially write a subquery to achieve a similar result, but a JOIN is generally more efficient and readable.

35. Question: Why might it be useful to analyze the number of orders by payment installments?

Answer: Analyzing the number of orders by payment installments can provide insights into customer payment preferences and financial behaviour, helping tailor payment plans and financing options to meet customer needs better.

36. Question: Why did you include a WHERE clause to filter by payment\_installments >= 1?

Answer: The WHERE clause ensures we only consider orders with valid payment installments, excluding any erroneous or non-installment payments that might distort the analysis.

37. Question: Why did you use COUNT() instead of SUM() for the number of orders?

Answer: COUNT() accurately counts the number of orders for each installment category, whereas SUM() would aggregate a numeric value, which is not appropriate for counting occurrences.

38. Question: What would be the impact of not using GROUP BY in your queries?

Answer: Not using GROUP BY would result in aggregating all data together without differentiation, leading to a single total count rather than counts by payment type, month, or installments. This would obscure the insights that come from comparing different groups.

Insights:

* Order volume has grown over the past years, indicating a flourishing e-commerce market in Brazil. Orders peak in afternoons and during certain months, suggesting targeted promotions during these times.
* There's a variation in average order value across states. Target may explore optimizing pricing strategies based on location and customer segments.
* Order value has increased year-over-year (2017 vs 2018, Jan-Aug). This suggests the potential for upselling or introducing higher-priced products.
* Delivery times vary by state. Target can prioritize improvements in regions with slower delivery times. A comparison of actual vs. estimated delivery times can help identify areas for improvement in logistics or communication.
* Understanding popular payment methods (credit card, debit card, installments) can help tailor the checkout process. A significant portion of customers uses installments, indicating a preference for spreading purchase costs. Target might explore offering more attractive instalment plans or partnering with financing companies.

Recommendations**:**

* Implement data-driven promotions based on seasonality, customer location, and order history.
* Focus on improving delivery times in states with slower performance.
* Develop location-specific pricing strategies considering customer segments and average order values.
* Offer a variety of payment options including instalments to cater to customer preferences.
* Segment customers based on demographics, purchase behaviour, and location to personalize marketing efforts.
* Optimize inventory levels in warehouses across Brazil to ensure faster delivery times.