

# SQL BUSINESS CASE

# TARGET



Submitted By:

Deependu Ghosh

DSML Feb24 Beginner Mon 1

[deependughosh@yahoo.in](mailto:deependughosh@yahoo.in)

# Table of Contents

<b>Project Overview.....</b>	<b>4-4</b>
1. Context .....	4
2. Dataset Files .....	4
3. Dataset Schema .....	4
 <b>Problem Statement.....</b>	 <b>5-40</b>
1. Import the dataset and do the usual exploratory analysis steps like checking the structure & characteristics of the dataset:.....	5-6
1.1: Data type of all columns in the “customer” table .....	5
1.2: Get the time range between which the orders were placed .....	6
1.3: Count the Cities & States of customers who ordered during the given period ....	6
 2. In-depth Exploration: .....	7-9
2.1: Is there a growing trend in the no. of orders placed over the past years? .....	7
2.2: Can we see some monthly seasonality in terms of the no. of orders being placed? .....	8
2.3: During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night) .....	9
 3. Evolution of E-commerce orders in the Brazil region:.....	10-11
3.1: Get the month-on-month no. of orders placed in each state.....	10
3.2: How are the customers distributed across all the states? .....	11
 4. Impact on the Economy: Analyse the money movement by e-commerce by looking at order prices, freight, and others. ....	12-14
4.1: Get the % increase in the cost of orders from the year 2017 to 2018 (include Months between Jan to Aug only). ....	12
4.2: Calculate the Total & Average value of the order price for each state.....	13
4.3: Calculate the Total & Average value of order freight for each state.....	14

<i>5. Analysis based on sales, freight, and delivery time.....</i>	<i>15-18</i>
5.1: 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 .....	15
5.2: Find out the top 5 states with the highest & lowest average freight value..	16-17
5.3: Find out the top 5 states with the highest & lowest average delivery time..	18-19
5.4: Find out the top 5 states where the order delivery is fast as compared to the estimated date of delivery.....	20
<i>6. Analysis based on the payments:.....</i>	<i>21-22</i>
6.1: Find the month-on-month no. of orders placed using different payment types. ...	21
6.2: Find the no. of orders placed based on the payment instalments that have been paid.....	22
<i>7. Recommendations (based on insights shared on every query):.....</i>	<i>23</i>

# PROJECT OVERVIEW

## CONTEXT:

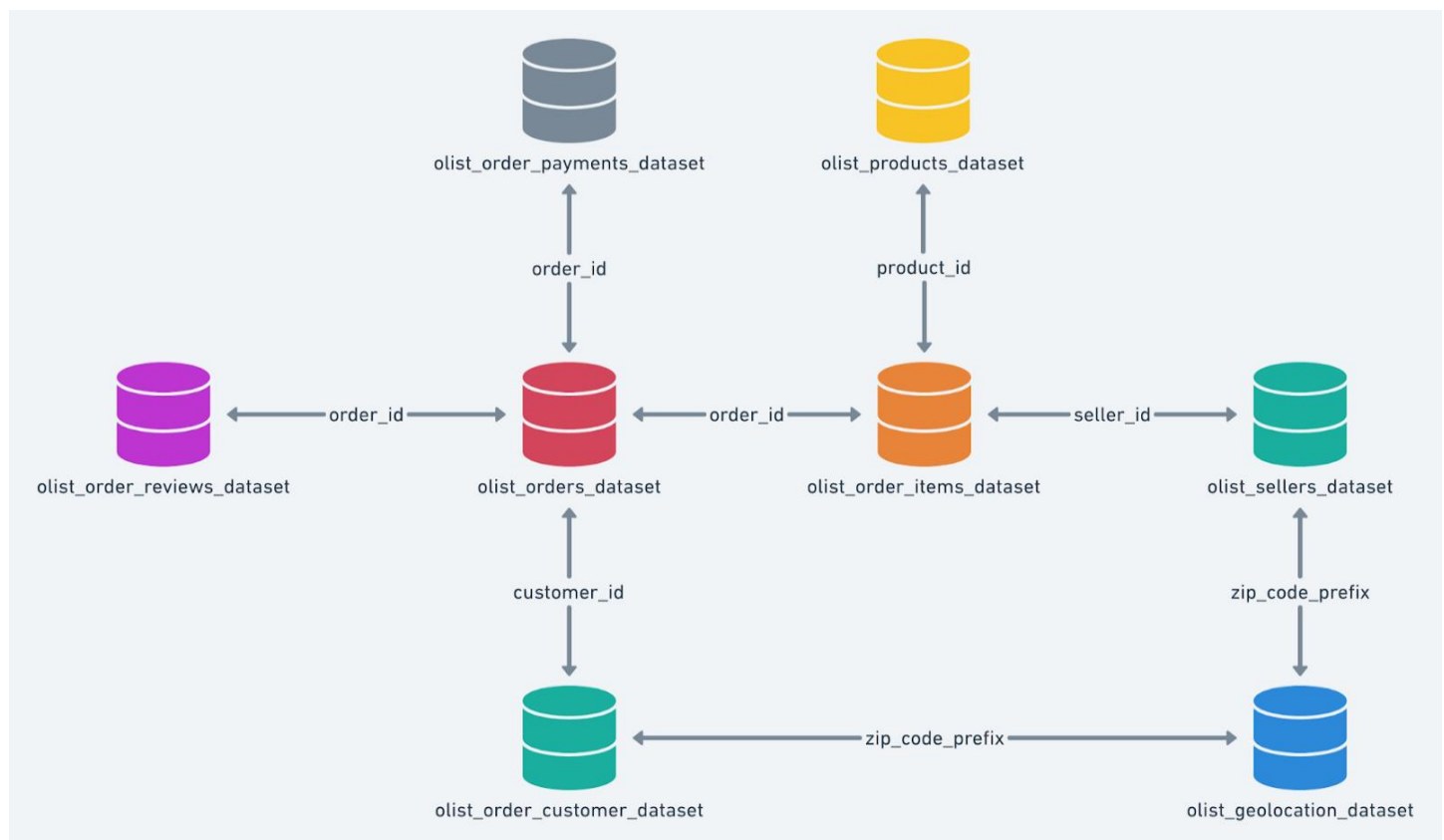
Target, a leading US retailer known for its exceptional value, innovation, and guest experience, is expanding its operations to Brazil. A detailed analysis of 100,000 orders from 2016 to 2018 provides insights into Target's performance in Brazil, including order processing, pricing, payment and shipping efficiency, customer demographics, product features, and satisfaction levels.

## DATASET FILES:

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

## DATASET SCHEMA:



## PROBLEM STATEMENT

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

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

Code -	<pre>SELECT column_name, data_type  FROM target.INFORMATION_SCHEMA.COLUMNS  WHERE table_name = 'customers';</pre>																				
Output -	<table><tr><th>Row</th><th>column_name</th><th>data_type</th></tr><tr><td>1</td><td>customer_id</td><td>STRING</td></tr><tr><td>2</td><td>customer_unique_id</td><td>STRING</td></tr><tr><td>3</td><td>customer_zip_code_prefix</td><td>INT64</td></tr><tr><td>4</td><td>customer_city</td><td>STRING</td></tr><tr><td>5</td><td>customer_state</td><td>STRING</td></tr></table>			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
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																			
Insights -	<p>We executed a query to get the <i>data type for each column</i> in the 'customers' table. This helps maintain data integrity, know the table's structure, create queries, and perform data analysis.</p> <ol style="list-style-type: none"><li>1. <u>Customer ID</u>: A unique identifier for customers used to manage customer information, purchase history, and track orders and shipments.</li><li>2. <u>Customer Unique ID</u>: A distinct ID used internally within the organization for operational purposes to differentiate between customers in a more detailed manner.</li></ol>																				

### 1.2. Get the time range between which the orders were placed.

Code -	SELECT MIN(order_purchase_timestamp) AS first_order_placed, MAX(order_purchase_timestamp) AS last_order_placed  FROM target.orders;								
Output -	<table><tr><th>Row</th><th>first_order_placed</th><th>last_order_placed</th></tr><tr><td>1</td><td>2016-09-04 21:15:19 UTC</td><td>2018-10-17 17:30:18 UTC</td></tr></table>			Row	first_order_placed	last_order_placed	1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC
Row	first_order_placed	last_order_placed							
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC							
Insights -	<ul style="list-style-type: none"><li>• We have successfully executed a query to extract the <i>timestamp of orders purchased between September 4th, 2016, and October 17th, 2018</i>, from the 'orders' table. This valuable information enables us to meticulously analyze order performance, evaluate customer patterns, trends, feedback, and statuses over time, and further improve operational efficiency.</li><li>• By utilizing this data, we can confidently ensure <i>customer satisfaction</i>, identify <i>seasonal trends</i>, and increase overall <i>sales growth</i>.</li></ul>								

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

Code -	<pre>SELECT COUNT(DISTINCT customer_city) AS Cities, COUNT(DISTINCT customer_state) AS States  FROM target.customers;</pre>											
Output -	<table><tr><th>Row</th><th>Cities</th><th>States</th><th></th></tr><tr><td>1</td><td>4119</td><td>27</td><td></td></tr></table>				Row	Cities	States		1	4119	27	
Row	Cities	States										
1	4119	27										
Insights -	<ul style="list-style-type: none"><li>It is imperative to consider the cities and states as a crucial source of information for conducting an effective analysis of the active customer base, identifying potential opportunities, and making informed decisions on resource allocation and strategic planning.</li><li>Ignoring this critical criterion could result in missed opportunities and suboptimal outcomes.</li></ul>											

## 2.In-depth Exploration:

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

Code -	<pre>WITH Growth_Trend AS (  SELECT  EXTRACT(YEAR FROM order_purchase_timestamp) AS Order_Year, COUNT(order_id) AS Order_Count  FROM target.orders  GROUP BY 1  ORDER BY 1 )  SELECT Order_Year, Order_Count,  CONCAT(ROUND(((Order_Count - Back_Track)/Back_Track) * 100, 2),' %') as Order_Growth_Pcnt  FROM (  SELECT *,  LAG(Order_Count) OVER (ORDER BY Order_Year)AS Back_Track  FROM Growth_Trend)  ORDER BY 1, 2;</pre>																			
Output -	<table><tr><th>Row</th><th>Order_Year</th><th>Order_Count</th><th>Order_Growth_Pcnt</th></tr><tr><td>1</td><td>2016</td><td>329</td><td>null</td></tr><tr><td>2</td><td>2017</td><td>45101</td><td>13608.51 %</td></tr><tr><td>3</td><td>2018</td><td>54011</td><td>19.76 %</td></tr></table>				Row	Order_Year	Order_Count	Order_Growth_Pcnt	1	2016	329	null	2	2017	45101	13608.51 %	3	2018	54011	19.76 %
Row	Order_Year	Order_Count	Order_Growth_Pcnt																	
1	2016	329	null																	
2	2017	45101	13608.51 %																	
3	2018	54011	19.76 %																	
Insights -	<ul style="list-style-type: none"><li>The analysis of queries indicates a consistent increase in the number of orders placed each year. There was a <u>substantial boost in order growth</u> between 2016 and 2017, with a <u>subsequent decrease in growth</u> rate in 2018.</li><li>These findings provide a clear insight into the <u>patterns and trends</u> in order placements over the specified duration (i.e., from 2016 to 2018).</li><li>Overall, the <u>orders</u> show a <i>consistent and significant upward trend</i>, indicating positive business performance and increasing customer demand for the organisation.</li></ul>																			

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

Code -	<pre>WITH Growth_Trend AS(  SELECT EXTRACT(MONTH FROM order_purchase_timestamp) AS Order_Month,  EXTRACT(YEAR FROM order_purchase_timestamp) AS Order_Year,  COUNT(order_id) AS Order_Count  FROM target.orders  GROUP BY 2, 1  ORDER BY 2, 1)  SELECT Order_Month, Order_Year, Order_Count  FROM Growth_Trend  ORDER BY 2, 1;</pre>																																															
Output -	<table><tr><th>Row</th><th>Order_Month</th><th>Order_Year</th><th>Order_Count</th></tr><tr><td>1</td><td>9</td><td>2016</td><td>4</td></tr><tr><td>2</td><td>10</td><td>2016</td><td>324</td></tr><tr><td>3</td><td>12</td><td>2016</td><td>1</td></tr><tr><td>4</td><td>1</td><td>2017</td><td>800</td></tr><tr><td>5</td><td>2</td><td>2017</td><td>1780</td></tr><tr><td>6</td><td>3</td><td>2017</td><td>2682</td></tr><tr><td>7</td><td>4</td><td>2017</td><td>2404</td></tr><tr><td>8</td><td>5</td><td>2017</td><td>3700</td></tr><tr><td>9</td><td>6</td><td>2017</td><td>3245</td></tr><tr><td>10</td><td>7</td><td>2017</td><td>4026</td></tr></table>				Row	Order_Month	Order_Year	Order_Count	1	9	2016	4	2	10	2016	324	3	12	2016	1	4	1	2017	800	5	2	2017	1780	6	3	2017	2682	7	4	2017	2404	8	5	2017	3700	9	6	2017	3245	10	7	2017	4026
Row	Order_Month	Order_Year	Order_Count																																													
1	9	2016	4																																													
2	10	2016	324																																													
3	12	2016	1																																													
4	1	2017	800																																													
5	2	2017	1780																																													
6	3	2017	2682																																													
7	4	2017	2404																																													
8	5	2017	3700																																													
9	6	2017	3245																																													
10	7	2017	4026																																													
Insights -	<ul style="list-style-type: none"><li>The query's objective is to calculate the total number of orders for each month over the years, to gain insights and manage the business's order patterns. There is a clear monthly seasonal pattern in the order count, with fluctuations throughout the year, likely influenced by holidays or specific shopping periods.</li><li>The data indicates that <i>November 2017</i> had the <u>highest number of orders</u>, while <i>September 2016</i> had the <u>lowest</u>. There are distinct monthly seasonality hotspots as shown by the pattern of growth and decline in the number of orders across different years. There was a <i>significant increase in orders from 2016 to 2017</i>, although there was a <i>decrease in the average number of orders in 2018</i> compared to the previous year.</li></ul>																																															



2.3. 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

Code -	<pre>WITH Pref_Time AS (  SELECT CASE  WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 0 and 6 THEN 'DAWN'  WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 7 and 12 THEN 'MORNING'  WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 13 and 18 THEN 'AFTERNOON'  ELSE 'NIGHT' END AS Time_of_Day,  COUNT(order_id) AS Order_Count  FROM target.orders  GROUP BY Time_of_Day)  SELECT * FROM Pref_Time  GROUP BY 1, 2  ORDER BY 2 DESC;</pre>																	
Output -	<table><tr><th>Row</th><th>Time_of_Day</th><th>Order_Count</th></tr><tr><td>1</td><td>AFTERNOON</td><td>38135</td></tr><tr><td>2</td><td>NIGHT</td><td>28331</td></tr><tr><td>3</td><td>MORNING</td><td>27733</td></tr><tr><td>4</td><td>DAWN</td><td>5242</td></tr></table>			Row	Time_of_Day	Order_Count	1	AFTERNOON	38135	2	NIGHT	28331	3	MORNING	27733	4	DAWN	5242
Row	Time_of_Day	Order_Count																
1	AFTERNOON	38135																
2	NIGHT	28331																
3	MORNING	27733																
4	DAWN	5242																
Insights -	<ul style="list-style-type: none"><li>• According to the data, it appears that Brazilian customers tend to place the <u>highest number of orders</u> in the <i>afternoon</i> and <i>night</i>, with a <u>substantial volume of orders</u> also being placed in the <i>morning</i>. However, there are <u>relatively fewer orders</u> placed during the <i>dawn hours</i>.</li><li>• This implies that Brazilian customers are quite <i>active in online shopping</i> during their leisure time in the afternoon and night, with a significant number of orders being placed in the morning as well.</li></ul>																	

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

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

Code -	<pre>SELECT  EXTRACT(MONTH FROM order_purchase_timestamp) AS Month,  EXTRACT(YEAR FROM order_purchase_timestamp) AS Year,  COUNT(order_id) AS Order_Count,  customer_state AS Order_State  FROM target.orders AS o JOIN target.customers AS c  USING (customer_id)  GROUP BY 1,2,4  ORDER BY 2,4;</pre>																																																												
Output -	<table><tr><th>Row</th><th>Month</th><th>Year</th><th>Order_Count</th><th>Order_State</th></tr><tr><td>1</td><td>10</td><td>2016</td><td>2</td><td>AL</td></tr><tr><td>2</td><td>10</td><td>2016</td><td>4</td><td>BA</td></tr><tr><td>3</td><td>10</td><td>2016</td><td>8</td><td>CE</td></tr><tr><td>4</td><td>10</td><td>2016</td><td>6</td><td>DF</td></tr><tr><td>5</td><td>10</td><td>2016</td><td>4</td><td>ES</td></tr><tr><td>6</td><td>10</td><td>2016</td><td>9</td><td>GO</td></tr><tr><td>7</td><td>10</td><td>2016</td><td>4</td><td>MA</td></tr><tr><td>8</td><td>10</td><td>2016</td><td>40</td><td>MG</td></tr><tr><td>9</td><td>10</td><td>2016</td><td>3</td><td>MT</td></tr><tr><td>10</td><td>10</td><td>2016</td><td>4</td><td>PA</td></tr></table>						Row	Month	Year	Order_Count	Order_State	1	10	2016	2	AL	2	10	2016	4	BA	3	10	2016	8	CE	4	10	2016	6	DF	5	10	2016	4	ES	6	10	2016	9	GO	7	10	2016	4	MA	8	10	2016	40	MG	9	10	2016	3	MT	10	10	2016	4	PA
Row	Month	Year	Order_Count	Order_State																																																									
1	10	2016	2	AL																																																									
2	10	2016	4	BA																																																									
3	10	2016	8	CE																																																									
4	10	2016	6	DF																																																									
5	10	2016	4	ES																																																									
6	10	2016	9	GO																																																									
7	10	2016	4	MA																																																									
8	10	2016	40	MG																																																									
9	10	2016	3	MT																																																									
10	10	2016	4	PA																																																									
Insights -	<ul style="list-style-type: none"><li>The query provides information on the number of orders placed in different states. By comparing <i>order counts</i>, we can identify states that consistently <u>perform well or poorly</u>. This information can help prioritize targeted strategies to increase orders in states with <i>low counts</i> month over month. Looking at state-wise order counts also indicates overall order trends, <u>highlighting states with steady or increasing orders</u>.</li><li>By analyzing this data, we can <u>identify growth opportunities</u> and develop suitable marketing strategies for different states. This will help <i>optimize operations</i> and <i>resource allocation</i> across Brazil.</li></ul>																																																												

### 3.2. How are the customers distributed across all the states?

<i>Code -</i>	<pre>SELECT customer_state AS States,  COUNT(customer_unique_id) AS Unique_Customers  FROM target.customers  GROUP BY 1  ORDER BY 2 ASC;</pre>																																	
<i>Output -</i>	<table><tr><th>Row</th><th>States</th><th>Unique_Customers</th></tr><tr><td>1</td><td>RR</td><td>46</td></tr><tr><td>2</td><td>AP</td><td>68</td></tr><tr><td>3</td><td>AC</td><td>81</td></tr><tr><td>4</td><td>AM</td><td>148</td></tr><tr><td>5</td><td>RO</td><td>253</td></tr><tr><td>6</td><td>TO</td><td>280</td></tr><tr><td>7</td><td>SE</td><td>350</td></tr><tr><td>8</td><td>AL</td><td>413</td></tr><tr><td>9</td><td>RN</td><td>485</td></tr><tr><td>10</td><td>PI</td><td>495</td></tr></table>	Row	States	Unique_Customers	1	RR	46	2	AP	68	3	AC	81	4	AM	148	5	RO	253	6	TO	280	7	SE	350	8	AL	413	9	RN	485	10	PI	495
Row	States	Unique_Customers																																
1	RR	46																																
2	AP	68																																
3	AC	81																																
4	AM	148																																
5	RO	253																																
6	TO	280																																
7	SE	350																																
8	AL	413																																
9	RN	485																																
10	PI	495																																
<i>Insights -</i>	<ul style="list-style-type: none"><li>• This query calculates the total number of unique customers across all states in Brazil (as customer_id's and customer_unique_id's are the same total count). By analyzing customer distribution across states in Brazil, we found that <i>RR</i> has the <u>lowest number of customers</u> (46), while <i>SP</i> has the <u>highest number of customers</u> (41746).</li><li>• This analysis can help identify potential hotspots for market opportunities for expansion and growth, and states with a higher number of regular customers suggest higher customer satisfaction.</li><li>• This data can be used to make informed decisions regarding marketing campaigns, expansion, customer retention, and investment opportunities.</li></ul>																																	

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

- 4.1. 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.

Code -	<pre>WITH Order_Cost_2017 AS (   SELECT ROUND(SUM(p.payment_value),2) AS Total_Cost_2017   FROM target.payments AS p JOIN target.orders AS o   USING (order_id)   WHERE EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017   AND EXTRACT(MONTH FROM o.order_purchase_timestamp)   BETWEEN 1 AND 8 ),   Order_Cost_2018 AS (   SELECT ROUND(SUM(p.payment_value),2) AS Total_Cost_2018   FROM target.payments AS p JOIN target.orders AS o   USING(order_id)   WHERE EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2018   AND EXTRACT(MONTH FROM o.order_purchase_timestamp)   BETWEEN 1 AND 8)    SELECT *, CONCAT(ROUND(((Total_Cost_2018 - Total_Cost_2017)   / Total_Cost_2017) * 100, 2),' %') AS Pcnt_Inc    FROM Order_Cost_2017, Order_Cost_2018;</pre>											
Output -	<table><tr><th>Row</th><th>Total_Cost_2017</th><th>Total_Cost_2018</th><th>Pcnt_Inc</th></tr><tr><td>1</td><td>3669022.12</td><td>8694733.84</td><td>136.98 %</td></tr></table>				Row	Total_Cost_2017	Total_Cost_2018	Pcnt_Inc	1	3669022.12	8694733.84	136.98 %
Row	Total_Cost_2017	Total_Cost_2018	Pcnt_Inc									
1	3669022.12	8694733.84	136.98 %									
Insights -	<ul style="list-style-type: none"><li>This query calculates the <i>percentage increase in the total cost</i> yearly (Jan to Aug period for 2017 and 2018). It identifies the <u>rise in order costs of products</u>, highlighting the pressing <i>need for pricing adjustments or changes in product offerings</i>. The increase could be due to <u>procurement, raw material costs, and demand-supply dynamics</u>.</li></ul>											

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

Code -	<pre>SELECT c.customer_state AS State,  ROUND(SUM(p.payment_value),2) AS Total_Value,  ROUND(AVG(p.payment_value),2) AS Avg_Value  FROM target.customers AS c JOIN target.orders AS o USING(customer_id)  JOIN target.payments AS p USING(order_id)  GROUP BY 1  ORDER BY 1;</pre>																																															
Output -	<table><tr><th>Row</th><th>State</th><th>Total_Value</th><th>Avg_Value</th></tr><tr><td>1</td><td>AC</td><td>19680.62</td><td>234.29</td></tr><tr><td>2</td><td>AL</td><td>96962.06</td><td>227.08</td></tr><tr><td>3</td><td>AM</td><td>27966.93</td><td>181.6</td></tr><tr><td>4</td><td>AP</td><td>16262.8</td><td>232.33</td></tr><tr><td>5</td><td>BA</td><td>616645.82</td><td>170.82</td></tr><tr><td>6</td><td>CE</td><td>279464.03</td><td>199.9</td></tr><tr><td>7</td><td>DF</td><td>355141.08</td><td>161.13</td></tr><tr><td>8</td><td>ES</td><td>325967.55</td><td>154.71</td></tr><tr><td>9</td><td>GO</td><td>350092.31</td><td>165.76</td></tr><tr><td>10</td><td>MA</td><td>152523.02</td><td>198.86</td></tr></table>				Row	State	Total_Value	Avg_Value	1	AC	19680.62	234.29	2	AL	96962.06	227.08	3	AM	27966.93	181.6	4	AP	16262.8	232.33	5	BA	616645.82	170.82	6	CE	279464.03	199.9	7	DF	355141.08	161.13	8	ES	325967.55	154.71	9	GO	350092.31	165.76	10	MA	152523.02	198.86
Row	State	Total_Value	Avg_Value																																													
1	AC	19680.62	234.29																																													
2	AL	96962.06	227.08																																													
3	AM	27966.93	181.6																																													
4	AP	16262.8	232.33																																													
5	BA	616645.82	170.82																																													
6	CE	279464.03	199.9																																													
7	DF	355141.08	161.13																																													
8	ES	325967.55	154.71																																													
9	GO	350092.31	165.76																																													
10	MA	152523.02	198.86																																													
Insights -	<ul style="list-style-type: none"><li>To determine both the total and average order amounts, we queried the <i>payment_value</i> column in the <i>payments</i> table.</li><li>Utilizing this data is crucial in <u>creating effective marketing campaigns</u> and gaining <u>valuable insights</u> into <u>regional customer behaviour</u> and <u>preferences</u>.</li></ul>																																															

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

Code -	<pre>SELECT c.customer_state AS State,  ROUND(SUM(oi.freight_value),2) AS Total_Freight_Value,  ROUND(AVG(oi.freight_value),2) AS Avg_Freight_Value  FROM target.customers AS c JOIN target.orders AS o USING(customer_id)  JOIN target.order_items AS oi USING(order_id)  GROUP BY 1  ORDER BY 1;</pre>																																												
Output -	<table><tr><th>Row</th><th>State</th><th>Total_Freight_Value</th><th>Avg_Freight_Value</th></tr><tr><td>1</td><td>AC</td><td>3686.75</td><td>40.07</td></tr><tr><td>2</td><td>AL</td><td>15914.59</td><td>35.84</td></tr><tr><td>3</td><td>AM</td><td>5478.89</td><td>33.21</td></tr><tr><td>4</td><td>AP</td><td>2788.5</td><td>34.01</td></tr><tr><td>5</td><td>BA</td><td>100156.68</td><td>26.36</td></tr><tr><td>6</td><td>CE</td><td>48351.59</td><td>32.71</td></tr><tr><td>7</td><td>DF</td><td>50625.5</td><td>21.04</td></tr><tr><td>8</td><td>ES</td><td>49764.6</td><td>22.06</td></tr><tr><td>9</td><td>GO</td><td>53114.98</td><td>22.77</td></tr><tr><td>10</td><td>MA</td><td>31523.77</td><td>38.26</td></tr></table>	Row	State	Total_Freight_Value	Avg_Freight_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
Row	State	Total_Freight_Value	Avg_Freight_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																																										
Insights -	<ul style="list-style-type: none"><li>States like SP and MG have <u>lower average freight values</u>, indicating more <i>efficient logistics networks</i> or <i>economies of scale</i> driving down shipping costs per order.</li><li>It's important to note that states like PB and PI have significantly <i>higher average freight values</i>, which indicates that their <u>shipping costs per order are relatively higher</u>. These valuable insights can be used to effectively <i>optimize logistics operations</i> and gain a <i>better understanding of regional shipping trends</i>.</li></ul>																																												

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

5.1. 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.

Code -	<pre>SELECT order_id,  DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) AS Time_to_Deliver,  DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY) AS Diff_Estimated_Delivery  FROM target.orders  WHERE order_status = 'delivered'  AND DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) IS NOT NULL  ORDER BY 1 ASC;</pre>																																												
Output -	<table><tr><th>Row</th><th>order_id</th><th>Time_to_Deliver</th><th>Diff_Estimated_Delivery</th></tr><tr><td>1</td><td>00010242fe8c5a6d1ba2dd792...</td><td>7</td><td>8</td></tr><tr><td>2</td><td>00018f77f2f0320c557190d7a1...</td><td>16</td><td>2</td></tr><tr><td>3</td><td>000229ec398224ef6ca0657da...</td><td>7</td><td>13</td></tr><tr><td>4</td><td>00024acbcd0a6daa1e931b03...</td><td>6</td><td>5</td></tr><tr><td>5</td><td>00042b26cf59d7ce69dfabb4e...</td><td>25</td><td>15</td></tr><tr><td>6</td><td>00048cc3ae777c65dbb7d2a06...</td><td>6</td><td>14</td></tr><tr><td>7</td><td>00054e8431b9d7675808bcb8...</td><td>8</td><td>16</td></tr><tr><td>8</td><td>000576fe39319847cbb9d288c...</td><td>5</td><td>15</td></tr><tr><td>9</td><td>0005a1a1728c9d785b8e2b08...</td><td>9</td><td>0</td></tr><tr><td>10</td><td>0005f50442cb953dcd1d21e1f...</td><td>2</td><td>18</td></tr></table>	Row	order_id	Time_to_Deliver	Diff_Estimated_Delivery	1	00010242fe8c5a6d1ba2dd792...	7	8	2	00018f77f2f0320c557190d7a1...	16	2	3	000229ec398224ef6ca0657da...	7	13	4	00024acbcd0a6daa1e931b03...	6	5	5	00042b26cf59d7ce69dfabb4e...	25	15	6	00048cc3ae777c65dbb7d2a06...	6	14	7	00054e8431b9d7675808bcb8...	8	16	8	000576fe39319847cbb9d288c...	5	15	9	0005a1a1728c9d785b8e2b08...	9	0	10	0005f50442cb953dcd1d21e1f...	2	18
Row	order_id	Time_to_Deliver	Diff_Estimated_Delivery																																										
1	00010242fe8c5a6d1ba2dd792...	7	8																																										
2	00018f77f2f0320c557190d7a1...	16	2																																										
3	000229ec398224ef6ca0657da...	7	13																																										
4	00024acbcd0a6daa1e931b03...	6	5																																										
5	00042b26cf59d7ce69dfabb4e...	25	15																																										
6	00048cc3ae777c65dbb7d2a06...	6	14																																										
7	00054e8431b9d7675808bcb8...	8	16																																										
8	000576fe39319847cbb9d288c...	5	15																																										
9	0005a1a1728c9d785b8e2b08...	9	0																																										
10	0005f50442cb953dcd1d21e1f...	2	18																																										
Insights -	<ul style="list-style-type: none"><li>Through a thorough analysis of this data, we can easily pinpoint orders that have surpassed the <i>average delivery timeline</i> and <i>evaluate their delivery efficiency</i>. The "diff_estimated_delivery" column showcases <u>negative values that represent delayed deliveries</u>, and <u>positive values that indicate early deliveries</u>.</li><li>By digging deeper into the <u>reasons for variances</u>, we can <i>optimize delivery timelines</i> and <i>minimize the gap</i> between estimated and actual delivery dates. Consequently, we can significantly <i>enhance logistics and delivery processes</i>.</li></ul>																																												

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

<i>Code_1 -</i>	<pre> SELECT c.customer_state, ROUND(AVG(oi.freight_value), 2) AS High_Avg_Freight_Value  FROM target.customers AS c JOIN target.orders AS o USING(customer_id)  JOIN target.order_items AS oi USING (order_id)  GROUP BY 1  ORDER BY 2 DESC  LIMIT 5; </pre>
-----------------	---

Output -	Row	customer_state	High_Avg_Freight_Va
	1	RR	42.98
	2	PB	42.72
	3	RO	41.07
	4	AC	40.07
	5	PI	39.15

<i>Code_2 -</i>	<pre> SELECT c.customer_state, ROUND(AVG(oi.freight_value), 2) AS Low_Avg_Freight_Value  FROM target.customers AS c JOIN target.orders AS o USING(customer_id)  JOIN target.order_items AS oi USING (order_id)  GROUP BY 1  ORDER BY 2 ASC  LIMIT 5; </pre>
-----------------	---

Output -	Row	customer_state	Low_Avg_Freight_Val
	1	SP	15.15
	2	PR	20.53
	3	MG	20.63
	4	RJ	20.96
	5	DF	21.04



*Insights -*

- It is important to note that different states have *varying average freight values*, which demonstrates that each state has *unique characteristics or logistical challenges* that significantly impact freight costs. Some states have *lower average freight values*, indicating that they have more favourable logistics infrastructure or other factors that contribute to reduced freight costs.
- By analyzing the average freight costs, businesses can easily identify significant variations in the average freight value among different states. This information is crucial to *optimize shipping and logistics operations*, as it enables businesses to *make informed decisions* regarding *pricing strategies, supply chain optimization, and resource allocation*.
- In states with *lower freight costs*, businesses can focus on further optimizing freight costs by *negotiating better rates, consolidating shipments, and improving logistics routes*.
- States with *higher freight costs*, can explore opportunities to mitigate factors contributing to higher costs, including *partnering with local logistics providers, optimizing transportation routes, and leveraging technology for efficient freight management*.

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

Code_1 -	<pre>SELECT c.customer_state,  ROUND(AVG(DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp,DAY)), 2) AS High_Avg_Del_Time  FROM target.orders AS o  JOIN target.customers AS c USING(customer_id)  GROUP BY 1  ORDER BY 2 DESC  LIMIT 5;</pre>																				
Output -	<table><tr><th>Row</th><th>customer_state</th><th>High_Avg_Del_Time</th></tr><tr><td>1</td><td>RR</td><td>46.17</td></tr><tr><td>2</td><td>AP</td><td>45.71</td></tr><tr><td>3</td><td>AM</td><td>44.76</td></tr><tr><td>4</td><td>AC</td><td>40.77</td></tr><tr><td>5</td><td>RO</td><td>38.41</td></tr></table>			Row	customer_state	High_Avg_Del_Time	1	RR	46.17	2	AP	45.71	3	AM	44.76	4	AC	40.77	5	RO	38.41
Row	customer_state	High_Avg_Del_Time																			
1	RR	46.17																			
2	AP	45.71																			
3	AM	44.76																			
4	AC	40.77																			
5	RO	38.41																			
Code_2 -	<pre>SELECT c.customer_state,  ROUND(AVG(DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp,DAY)), 2) AS Low_Avg_Del_Time  FROM target.orders AS o  JOIN target.customers AS c USING(customer_id)  GROUP BY 1  ORDER BY 2 ASC  LIMIT 5;</pre>																				
Output -	<table><tr><th>Row</th><th>customer_state</th><th>Low_Avg_Del_Time</th></tr><tr><td>1</td><td>SP</td><td>18.81</td></tr><tr><td>2</td><td>DF</td><td>24.06</td></tr><tr><td>3</td><td>MG</td><td>24.22</td></tr><tr><td>4</td><td>PR</td><td>24.25</td></tr><tr><td>5</td><td>ES</td><td>25.27</td></tr></table>			Row	customer_state	Low_Avg_Del_Time	1	SP	18.81	2	DF	24.06	3	MG	24.22	4	PR	24.25	5	ES	25.27
Row	customer_state	Low_Avg_Del_Time																			
1	SP	18.81																			
2	DF	24.06																			
3	MG	24.22																			
4	PR	24.25																			
5	ES	25.27																			

*Insights -*

- It is imperative to identify and address the factors causing these prolonged delivery durations. Businesses must rigorously evaluate their *logistics network, transportation routes, and last-mile delivery* processes to optimize efficiency. Collaborating with local logistics providers or establishing strategic partnerships is vital to *improving delivery performance* in these states. Implementing advanced technology solutions like real-time tracking systems and efficient delivery scheduling tools can significantly streamline last-mile operations and ensure timely deliveries.
- Open communication and collaborative relationships with *shipping carriers and logistics partners* are crucial for monitoring and addressing any issues promptly. By *optimizing the last-mile delivery process* and maintaining *efficient relationships with partners*, businesses can *reduce delivery times, enhance customer experience, and improve overall satisfaction.*

5.4. 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 date to figure out how fast the delivery was for each state.

Code -	<pre>SELECT c.customer_state as State,  ROUND(AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_d elivered_customer_date,DAY)), 2) as Fast_Avg_Odr_Del  from target.customers as c JOIN target.orders as o USING(customer_id)  GROUP BY 1  ORDER BY 2 DESC  LIMIT 5;</pre>																		
Output -	<table><tr><th>Row</th><th>State</th><th>Fast_Avg_Odr_Del</th></tr><tr><td>1</td><td>AC</td><td>19.76</td></tr><tr><td>2</td><td>RO</td><td>19.13</td></tr><tr><td>3</td><td>AP</td><td>18.73</td></tr><tr><td>4</td><td>AM</td><td>18.61</td></tr><tr><td>5</td><td>RR</td><td>16.41</td></tr></table>	Row	State	Fast_Avg_Odr_Del	1	AC	19.76	2	RO	19.13	3	AP	18.73	4	AM	18.61	5	RR	16.41
Row	State	Fast_Avg_Odr_Del																	
1	AC	19.76																	
2	RO	19.13																	
3	AP	18.73																	
4	AM	18.61																	
5	RR	16.41																	
Insights -	<ul style="list-style-type: none"><li>According to our query analysis, the top 5 states in Brazil have an outstanding record of <i>delivering orders within the estimated time frame</i>, with an <u>average difference of only 16 to 19 days</u>. This remarkable feat can be attributed to their <i>efficient delivery processes, well-established transportation networks, optimized routing strategies, and proactive coordination with shipping carriers</i>. By consistently meeting and exceeding customer expectations in terms of delivery time, Target can increase <i>customer satisfaction</i>, foster loyalty, and ultimately drive sales growth in Brazil.</li><li>The key to customer satisfaction is <u>reliable and prompt delivery</u>. Businesses can leverage this information to demonstrate their <i>effective logistics operations</i> and <i>gain a competitive advantage</i>. By comparing delivery times with other states that have longer times, companies can identify areas for <i>improving delivery efficiency</i> and ensure they are delivering at their best.</li></ul>																		

## 6. Analysis based on the payments:

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

Code -	<pre>SELECT  EXTRACT(YEAR FROM o.order_purchase_timestamp) AS Year,  EXTRACT(MONTH FROM o.order_purchase_timestamp) AS Month,  p.payment_type AS Payment_Type,  COUNT(order_id) AS Order_Count  FROM target.orders AS o JOIN target.payments AS p USING(order_id)  WHERE p.payment_type IS NOT NULL  GROUP BY 1,2,3  ORDER BY 1,2;</pre>																																																											
Output -	<table><tr><th>Row</th><th>Year</th><th>Month</th><th>Payment_Type</th><th>Order_Count</th></tr><tr><td>1</td><td>2016</td><td>9</td><td>credit_card</td><td>3</td></tr><tr><td>2</td><td>2016</td><td>10</td><td>credit_card</td><td>254</td></tr><tr><td>3</td><td>2016</td><td>10</td><td>UPI</td><td>63</td></tr><tr><td>4</td><td>2016</td><td>10</td><td>voucher</td><td>23</td></tr><tr><td>5</td><td>2016</td><td>10</td><td>debit_card</td><td>2</td></tr><tr><td>6</td><td>2016</td><td>12</td><td>credit_card</td><td>1</td></tr><tr><td>7</td><td>2017</td><td>1</td><td>credit_card</td><td>583</td></tr><tr><td>8</td><td>2017</td><td>1</td><td>UPI</td><td>197</td></tr><tr><td>9</td><td>2017</td><td>1</td><td>voucher</td><td>61</td></tr><tr><td>10</td><td>2017</td><td>1</td><td>debit_card</td><td>9</td></tr></table>					Row	Year	Month	Payment_Type	Order_Count	1	2016	9	credit_card	3	2	2016	10	credit_card	254	3	2016	10	UPI	63	4	2016	10	voucher	23	5	2016	10	debit_card	2	6	2016	12	credit_card	1	7	2017	1	credit_card	583	8	2017	1	UPI	197	9	2017	1	voucher	61	10	2017	1	debit_card	9
Row	Year	Month	Payment_Type	Order_Count																																																								
1	2016	9	credit_card	3																																																								
2	2016	10	credit_card	254																																																								
3	2016	10	UPI	63																																																								
4	2016	10	voucher	23																																																								
5	2016	10	debit_card	2																																																								
6	2016	12	credit_card	1																																																								
7	2017	1	credit_card	583																																																								
8	2017	1	UPI	197																																																								
9	2017	1	voucher	61																																																								
10	2017	1	debit_card	9																																																								
Insights -	<ul style="list-style-type: none"><li>The query provides crucial insights into customer payment preferences by thoroughly analyzing the monthly distribution of payment types. By analyzing the monthly distribution of payment types, businesses can gain invaluable insights into <i>customer payment preferences</i> and identify <i>significant trends</i> in payment methods.</li><li><i>Credit cards</i> are the undisputed preferred payment method due to their unmatched convenience, security, and widespread acceptance. Businesses must prioritize seamless credit card payment processing and maintain strong partnerships with payment service providers to stay competitive. <u>Analyzing payment trends</u> is vital to optimizing the checkout experience, expanding payment options, and ultimately enhancing overall customer satisfaction.</li></ul>																																																											

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

Code -	<pre>SELECT  COUNT(DISTINCT order_id) AS Order_Count,  p.payment_installments AS Payments_Paid  FROM target.orders AS o JOIN target.payments AS p  USING(order_id)  WHERE p.payment_installments IS NOT NULL AND p.payment_installments != 0  GROUP BY 2  ORDER BY 2;</pre>																																	
Output -	<table><tr><th>Row</th><th>Order_Count</th><th>Payments_Paid</th></tr><tr><td>1</td><td>49060</td><td>1</td></tr><tr><td>2</td><td>12389</td><td>2</td></tr><tr><td>3</td><td>10443</td><td>3</td></tr><tr><td>4</td><td>7088</td><td>4</td></tr><tr><td>5</td><td>5234</td><td>5</td></tr><tr><td>6</td><td>3916</td><td>6</td></tr><tr><td>7</td><td>1623</td><td>7</td></tr><tr><td>8</td><td>4253</td><td>8</td></tr><tr><td>9</td><td>644</td><td>9</td></tr><tr><td>10</td><td>5315</td><td>10</td></tr></table>	Row	Order_Count	Payments_Paid	1	49060	1	2	12389	2	3	10443	3	4	7088	4	5	5234	5	6	3916	6	7	1623	7	8	4253	8	9	644	9	10	5315	10
Row	Order_Count	Payments_Paid																																
1	49060	1																																
2	12389	2																																
3	10443	3																																
4	7088	4																																
5	5234	5																																
6	3916	6																																
7	1623	7																																
8	4253	8																																
9	644	9																																
10	5315	10																																
Insights -	<ul style="list-style-type: none"><li>Based on the results of our query, we have determined the count of distinct orders for each payment instalment option. This highlights the distribution of customers who opt for different instalment plans for their payments. The <u>instalment option that customers prefer</u> the most is the one where <i>at least 1 payment has been made</i>. Additionally, the data shows a significant customer preference for <i>instalment plans</i> ranging from 9 to 10 payments.</li><li>To <i>drive sales growth</i>, businesses should offer <u>flexible payment options and instalment plans</u> to customers. This accommodates <i>customer preferences, enhances affordability, and expands the customer base</i>. Incorporating installment plans into payment options is an <u>effective strategy</u> to <i>improve sales and customer satisfaction</i>.</li></ul>																																	

## 7. Recommendations:-

- TARGET must take immediate action to expand its customer base in Brazil. The current presence in only *4119 cities and 27 states* is not sufficient. The company must *focus on increasing its presence* across the country to reach out to more potential customers.
- To combat *slower months*, businesses must *conduct market research, collaborate with complementary businesses, offer irresistible promotions, and employ targeted marketing strategies*. Throughout the day, they must *constantly adjust their tactics* to maximize sales. In states with *higher* customer bases, they must *prioritize customer engagement*, while in states with *lower* customer bases, they must target and exploit *growth opportunities*. It's crucial to identify and leverage competitive advantages to stand out in the market, and businesses must be proactive in doing so.
- To optimize their shipping operations and enhance their businesses, companies must implement effective pricing strategies and allocate resources appropriately. By partnering with local carriers and logistics providers and negotiating costs, businesses can reduce expenses and improve logistics processes, resulting in faster delivery times. The use of technology solutions, like identifying the causes of delays in last-mile delivery processes, can streamline operations and reduce transit times, ultimately leading to improved customer satisfaction.
- It is imperative to maintain proactive communication with customers regarding delivery expectations. We must provide timely updates on order status and potential delays to ensure that our customers receive their orders on time and with no surprises. Our commitment to open communication guarantees that we will always keep our customers in the know. When you choose our services, you can rest assured that we will deliver your orders with utmost transparency and care.
- It is imperative to establish a secure payment infrastructure that supports various payment methods while keeping up with evolving customer preferences and emerging technologies. To achieve this, educating customers about alternative payment methods and offering incentives to encourage their usage is crucial. In addition, promoting the benefits of lower instalment options and creating value through targeted marketing campaigns can effectively drive customer engagement and loyalty.
- Gathering and scrutinizing customer feedback and behaviour is crucial in comprehending their preferences and guaranteeing their satisfaction. This, in turn, is vital for building and strengthening the branding.