



# Business Case Study



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

Q 1.1) Data type of all columns in the "customers" table:

SELECT

column\_name, data\_type

FROM my-scaler-project-sql.guru\_bv\_Ecommerce.INFORMATION\_SCHEMA.COLUMNS

WHERE table\_name = 'customers';

Query results



JOB INFORMATION

RESULTS

JSON

EXECUTION DETAILS

CHART

PREVIEW

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

customers QUERY SHARE COPY SNAPSHOT DELETE EXPORT

SCHEMA

DETAILS

PREVIEW

LINEAGE

DATA PROFILE

DATA QUALITY

Filter Enter property name or value

<input type="checkbox"/>	Field name	Type	Mode	Key	Collation	Default Value	Policy Tags	Description
<input type="checkbox"/>	<a href="#">customer_id</a>	STRING	NULLABLE					
<input type="checkbox"/>	<a href="#">customer_unique_id</a>	STRING	NULLABLE					
<input type="checkbox"/>	<a href="#">customer_zip_code_prefix</a>	INTEGER	NULLABLE					
<input type="checkbox"/>	<a href="#">customer_city</a>	STRING	NULLABLE					
<input type="checkbox"/>	<a href="#">customer_state</a>	STRING	NULLABLE					



All data types are 'string' except for customer\_zip\_code which is 'Integer' and no constraints are imposed on the column as it says 'nullable'.



## Business Case Study



Q 1.2) Get the time range between which the orders were placed.

```
SELECT
    MIN(order_purchase_timestamp) AS order_startdate,
    MAX(order_purchase_timestamp) AS order_enddate
FROM guru_bv_Ecommerce.orders;
```

Query results

[SAV](#)

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	CHART	PREVIEW
Row	order_startdate	order_enddate				
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC				



Market started its order by 4th September 2016 and ended by 17th October 2018.

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

```
SELECT
    DISTINCT c.customer_city,
    c.customer_state,
    COUNT(o.customer_id) AS count_of_customers
FROM guru_bv_Ecommerce.customers c
JOIN guru_bv_Ecommerce.orders o
ON c.customer_id = o.customer_id
GROUP BY 1,2
ORDER BY 3 DESC;
```

Query results

[SAVE RE](#)

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	CHART	PREVIEW
Row	customer_city	customer_state	count_of_customers			
1	sao paulo	SP	15540			
2	rio de janeiro	RJ	6882			
3	belo horizonte	MG	2773			
4	brasilia	DF	2131			
5	curitiba	PR	1521			
6	campinas	SP	1444			



As per the records, max.orders have been placed by customers from Sao Paulo city in SP State (15540 no.of customers) during the market time which is from 4th of September 2016 to 17th of October 2018.



# Business Case Study



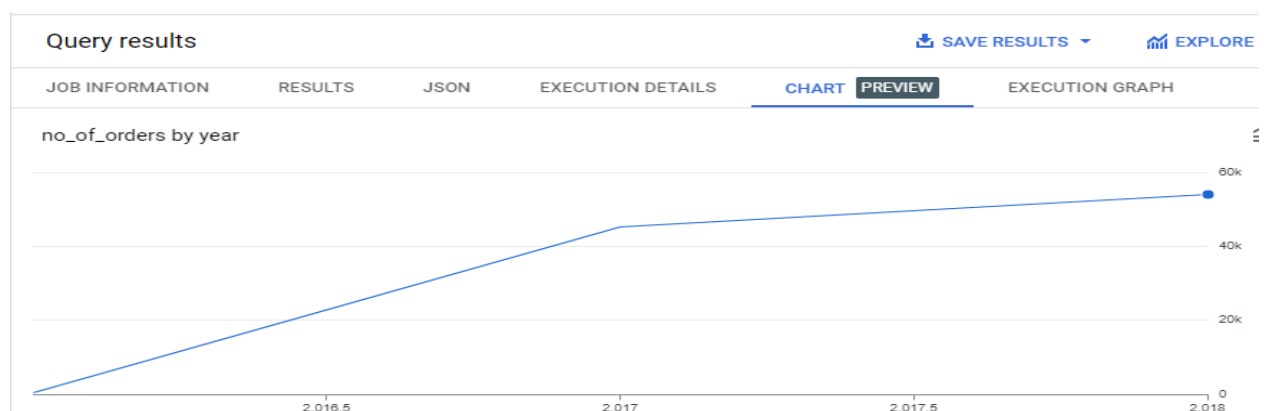
## In-depth Exploration

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

```
SELECT
    EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
    COUNT(DISTINCT order_id) AS no_of_orders
FROM guru_bv_Ecommerce.customers c
JOIN guru_bv_Ecommerce.orders o
ON c.customer_id = o.customer_id
GROUP BY year
ORDER BY year;
```

### Query results

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	CHART
Row	year	no_of_orders			
1	2016	329			
2	2017	45101			
3	2018	54011			



Yes, of course there is a drastic increase in no.of orders placed from 2016(329) to 2017(45101) and subsequently there is good amount of increase in orders when compared to 2017(45101) to 2018(54011), but the huge difference is between 2016 & 2017.



## Business Case Study



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

By monthly only trends:

SELECT

```
EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
```

```
COUNT(DISTINCT order_id) AS no_of_orders
```

```
FROM guru_bv_Ecommerce.orders o
```

```
JOIN guru_bv_Ecommerce.customers c
```

```
ON c.customer_id = o.customer_id
```

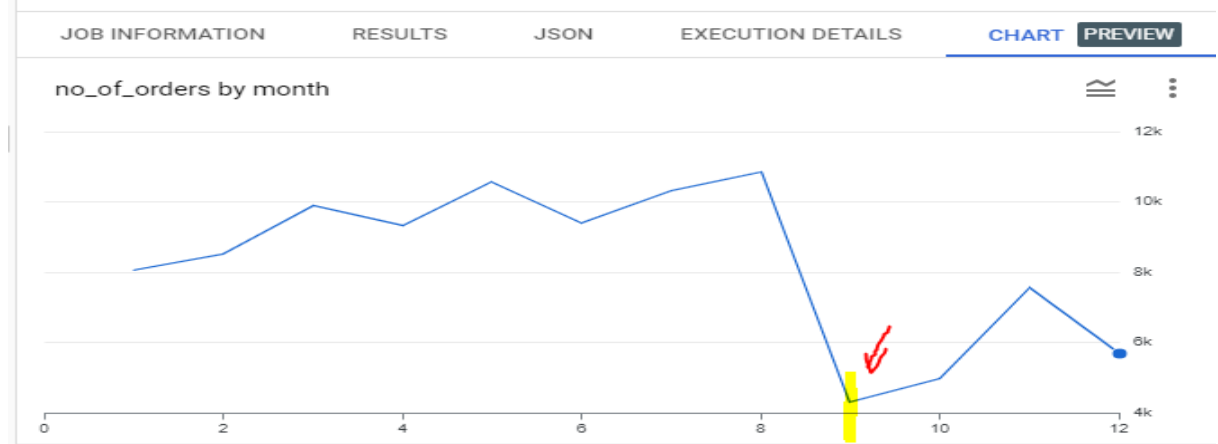
```
GROUP BY 1
```

```
ORDER BY 1;
```

### Query results

JOB INFORMATION		RESULTS		JSON	EXECUTION DETAILS
Row	month		no_of_orders		
1	1	1	8069		
2	2	2	8508		
3	3	3	9893		
4	4	4	9343		
5	5	5	10573		
6	6	6	9412		
7	7	7	10318		
8	8	8	10843		
9	9	9	4305		
10	10	10	4959		

### Query results



**Key insight - exponentially there are good orders being placed during the months of February to August with a maximum order attained by the month of August(10843) and during the month of September had the min order(4305).**



# Business Case Study



**By yearly & monthly trends:**

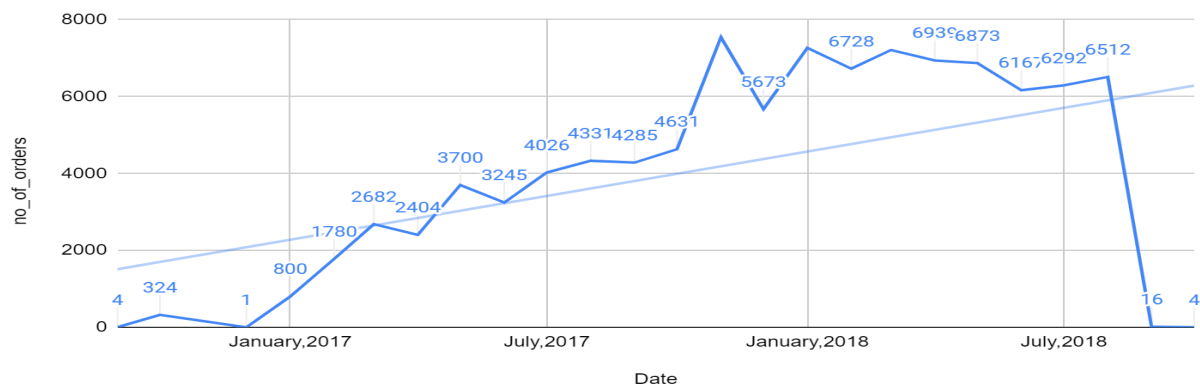
```
SELECT
    EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
    EXTRACT(YEAR FROM o.order_purchase_timestamp) AS year,
    COUNT(DISTINCT order_id) AS no_of_orders
FROM guru_bv_Ecommerce.orders o
JOIN guru_bv_Ecommerce.customers c
ON c.customer_id = o.customer_id
GROUP BY 1,2
ORDER BY 2,1;
```

Query results SAVE RESULTS E

JOB INFORMATION	RESULTS	JSON	EXECUTION DETAILS	CHART	PREVIEW	EXECUTION GRAPH
Row	month	year	no_of_orders			
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			

Results per page: 50 1 - 25 of 25

no\_of\_orders vs. Date



**Key insight - Based on the analysis of order count, it can be observed that there is a growing trend .The count of purchases has shown an overall upward trend, with some fluctuations.**



## Business Case Study



Q 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

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 'Mornings'
WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 13 AND 18 THEN 'Afternoon'
WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 19 AND 23 THEN 'Night'
END AS hour,
COUNT(order_id) AS order_count
FROM guru_bv_Ecommerce.orders o
JOIN guru_bv_Ecommerce.customers c
ON o.customer_id = c.customer_id
GROUP BY 1
ORDER BY 2 DESC;
```

### Query results

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

### Query results

[SAVE RESULTS](#)

[EXPLORE DATA](#)

JOB INFORMATION

RESULTS

JSON

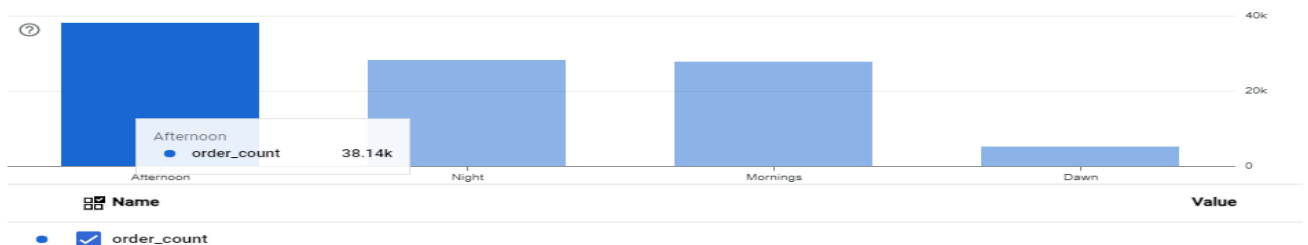
EXECUTION DETAILS

CHART

PREVIEW

EXECUTION GRAPH

order\_count by hour



**Key insight - Based on the analysis, during Afternoon only maximum orders has been placed by the customers(38135).**



# Business Case Study



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

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

SELECT

```
c.customer_state,
EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
COUNT(DISTINCT order_id) AS no_of_orders
```

FROM guru\_bv\_Ecommerce.orders o

JOIN guru\_bv\_Ecommerce.customers c

ON c.customer\_id = o.customer\_id

GROUP BY 1,2

ORDER BY 1,2;

Query results

SAVE RESULTS PREVIEW EXECUTION GRAPH

Row	customer_state	month	no_of_orders
1	AC	1	8
2	AC	2	6
3	AC	3	4
4	AC	4	9
5	AC	5	10
6	AC	6	7
7	AC	7	9
8	AC	8	7
9	AC	9	5
10	AC	10	6
11	AC	11	5
12	AC	12	5
13	AL	1	39
14	AL	2	39
15	AL	3	40
16	AL	4	51

Results per page: 50 1 - 50 of 322

Connected Sheet 1

A	B	C	D	E	F	G	H
SUM of no_of_o month	1	2	3	4	5	6	7
customer_state							
SP	3351	3357	4047	3967	4632	4104	4381
RJ	990	1176	1302	1172	1321	1128	1288
MG	971	1063	1237	1061	1190	1080	1111
RS	427	473	569	488	559	526	565
PR	443	460	504	500	524	478	523
SC	345	316	362	351	379	321	356
BA	264	273	340	318	368	307	405
DF	151	196	207	183	208	220	243
ES	159	186	182	188	228	204	206
GO	164	176	199	177	226	184	192
PE	113	146	153	154	174	140	210
CE	99	101	126	143	136	121	140
PA	82	83	109	107	75	92	96
MT	96	84	71	92	104	83	85
MA	66	67	77	73	65	59	79
MS	71	75	79	58	74	76	74
PB	33	47	55	51	47	51	79
PI	55	46	48	50	56	43	52
RN	51	31	52	42	39	49	56
AL	39	39	40	51	46	34	40
AC	24	27	43	27	19	37	42

Tr customer\_state

Order Descen... Sort by SUM of... Grand...

Number of rows All Show totals

Columns Add

123 month

Order Ascen... Sort by month

Number of columns All Group by None Show totals



Key insights- From the analysis, it is evident that SP state consistently has the highest no.of orders in any given month when compared to other states.



# Business Case Study



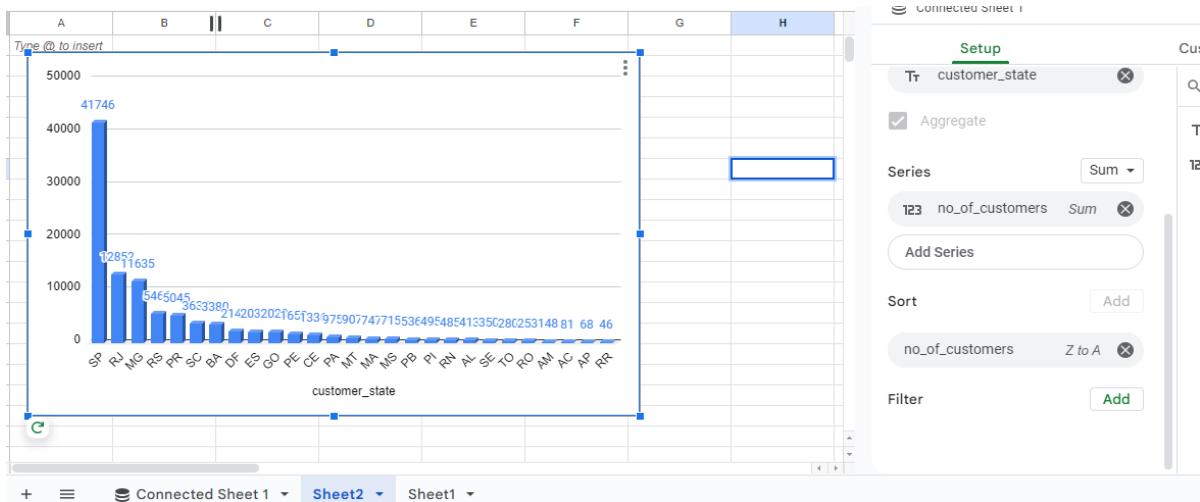
Q 3.2) How are the customers distributed across all the states?

```
SELECT customer_state,
        COUNT(customer_id) AS no_of_customers
FROM    guru_bv_Ecommerce.customers
GROUP BY 1
ORDER BY 2 DESC;
```

Query results SAVE RESULTS EXPLORE DATA

JOB INFORMATION	RESULTS	JSON	EXECUTION DETAILS	CHART	PREVIEW	EXECUTION GRAPH
Row	customer_state	no_of_customers				
1	SP	41746				
2	RJ	12852				
3	MG	11635				
4	RS	5466				
5	PR	5045				
6	SC	3637				
7	BA	3380				
8	DF	2140				
9	ES	2033				
10	GO	2020				
11	PE	1652				
12	CE	1336				
13	PA	975				
14	MT	907				
15	MA	747				
16	MS	715				

Results per page: 50 1 - 27 of 27



**Key insights-** From the analysis, it is evident that SP state has the maximum no. of customers (41746) when compared to other states.





## Business Case Study



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

Q 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.

### OVERALL % increase

```
SELECT
    ROUND(((SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2018
THEN p.payment_value ELSE 0 END )
-
SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017
THEN p.payment_value ELSE 0 END))
/
SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017
THEN p.payment_value END))*100,2) AS percent_increase_in_cost
FROM guru_bv_Ecommerce.orders o
JOIN guru_bv_Ecommerce.payments p ON o.order_id = p.order_id
WHERE EXTRACT(YEAR FROM o.order_purchase_timestamp) IN (2017, 2018) AND
EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8;
```

Query results		SAVE RESULTS	EXPLORE DATA
JOB INFORMATION		RESULTS	JSON
		EXECUTION DETAILS	CHART
		PREVIEW	EXECUTION GRAPH
Row	percent_increase_in_cost		
1	136.98		



**Key insights-** From the analysis, the overall percentage increase in the cost of orders from 2017 to 2018, including only the months from January to August, is 136.98%



# Business Case Study



% increase w.r.t months

SELECT

```
EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
ROUND(((SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2018 THEN
p.payment_value ELSE 0 END )
-
SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017 THEN
p.payment_value ELSE 0 END))
/
SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017 THEN
p.payment_value END))*100,2)
AS percent_increase_in_cost
```

FROM

guru\_bv\_Ecommerce.orders o

JOIN

guru\_bv\_Ecommerce.payments p ON o.order\_id = p.order\_id

WHERE

EXTRACT(YEAR FROM o.order\_purchase\_timestamp) IN (2017, 2018)

AND


EXTRACT(MONTH FROM o.order\_purchase\_timestamp) BETWEEN 1 AND 8

GROUP BY 1

ORDER BY 1;

Query results				
JOB INFORMATION		RESULTS		JSON
Row	month		percent_increase_in	
1		1	705.13	
2		2	239.99	
3		3	157.78	
4		4	177.84	
5		5	94.63	
6		6	100.26	
7		7	80.04	
8		8	51.61	



 **Key insights- On basis of month-wise increase analysis , January shows the highest percentage increase, followed by February and April respectively.**



# Business Case Study



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

SELECT

c.customer\_state,

ROUND(SUM(oi.price),2) AS Total\_Price,

ROUND(AVG(oi.price),2) AS Average\_Price

FROM guru\_bv\_Ecommerce.customers c

LEFT JOIN guru\_bv\_Ecommerce.orders o USING(customer\_id)

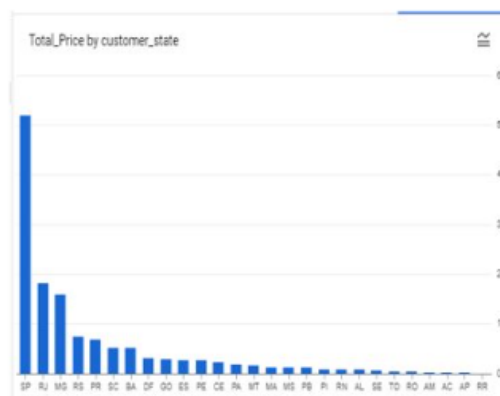
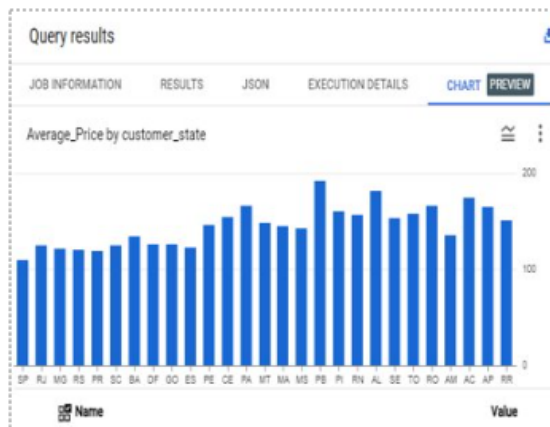
LEFT JOIN guru\_bv\_Ecommerce.order\_items oi USING(order\_id)

GROUP BY 1

ORDER BY 2 DESC;

Query results

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	CHAR
Row	customer_state	Total_Price	Average_Price		
1	SP	5202955.05	109.65		
2	RJ	1824092.67	125.12		
3	MG	1585308.03	120.75		
4	RS	750304.02	120.34		
5	PR	683083.76	119.0		
6	SC	520553.34	124.65		
7	BA	511349.99	134.6		
8	DF	302603.94	125.77		
9	GO	294591.95	126.27		
10	ES	275037.31	121.91		
11	PE	262788.03	145.51		
12	CE	227254.71	153.76		
13	PA	178947.81	165.69		



**Key insights-** State SP has the highest total price value(5202955.05), but has the lowest average price value among all states. On the other hand, the state of PB has the highest average price value(191.48) among all states.



# Business Case Study

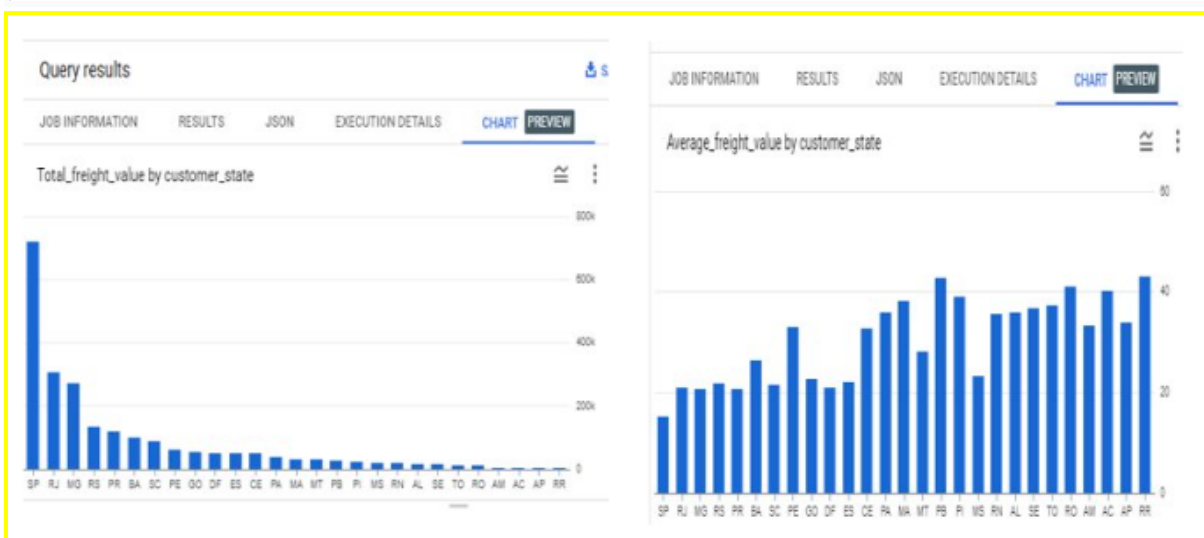


Q 4.3) Calculate the Total & Average value of order freight for each state.

SELECT

```
c.customer_state,
ROUND(SUM(oi.freight_value),2) AS Total_freight_value,
ROUND(AVG(oi.freight_value),2) AS Average_freight_value
FROM guru_bv_Ecommerce.customers c
LEFT JOIN guru_bv_Ecommerce.orders o USING(customer_id)
LEFT JOIN guru_bv_Ecommerce.order_items oi USING(order_id)
GROUP BY 1
ORDER BY 2 DESC;
```

Query results					SAVE RESULTS	
JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	CHART	PREVIEW
Row	customer_state	Total_freight_value	Average_freight_value			
1	SP	718723.07	15.15			
2	RJ	305589.31	20.96			
3	MG	270853.46	20.63			
4	RS	135522.74	21.74			
5	PR	117851.68	20.53			
6	BA	100156.68	26.36			
7	SC	89660.26	21.47			
8	PE	59449.66	32.92			
9	GO	53114.98	22.77			
10	DF	50000.00	21.00			



**Key insights-** State SP has the highest total freight value(718723.07), but has the lowest average freight value among all states. On the other hand, the state of RR has the highest average freight value(42.98) followed by PB (42.72)among all states.



## Business Case Study



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

Q 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.

You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:

- $\text{time\_to\_deliver} = \text{order\_delivered\_customer\_date} - \text{order\_purchase\_timestamp}$
- $\text{diff\_estimated\_delivery} = \text{order\_estimated\_delivery\_date} - \text{order\_delivered\_customer\_date}$

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)

diff\_estimated\_delivery

FROM guru\_bv\_Ecommerce.orders

WHERE DATE\_DIFF(order\_delivered\_customer\_date,order\_purchase\_timestamp, DAY)IS NOT NULL

ORDER BY 3 DESC;

Query results					SAVE RESULTS	E
JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	CHART	PREVIEW
Row	order_id	time_to_deliver	diff_estimated_delivery			
1	0607f0feea4b566f1eb8f7d3c2...	3	146			
2	c72727d29cde4cf870d569bf6...	6	139			
3	eec7f369423b033e549c02f3c...	20	134			
4	c2bb89b5c1dd978d507284be...	16	123			
5	40dc2ba6f322a17626aac6244...	7	108			
6	1a695d543b7302aa9446c8d5f...	12	83			
7	39e0115911bf404857e14baa7...	11	82			
8	38930f76efb00b138f4d632e4d...	11	77			
9	c5132855100a12d63ed4e8ae0...	12	77			
10	559eea5a72341a4c82dbce988...	13	77			
11	cfe2129968372224ea392ad47...	14	76			

Results per page: 50 1 – 50 of 96476



**Key insights-** As per the analysis, maximum order's diff\_estimated\_delivery is 146 days and maximum order's time\_to\_deliver is 209 days.



# Business Case Study



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

WITH gbv4 AS

(SELECT

```
c.customer_state AS state,
ROUND(AVG(oi.freight_value),2) AS Average_freight_value,
DENSE_RANK()OVER(ORDER BY AVG(oi.freight_value)DESC) AS highest_rnk,
DENSE_RANK()OVER(ORDER BY AVG(oi.freight_value)) AS lowest_rnk
```

FROM guru\_bv\_Ecommerce.customers c

LEFT JOIN guru\_bv\_Ecommerce.orders o USING(customer\_id)

LEFT JOIN guru\_bv\_Ecommerce.order\_items oi USING(order\_id)

GROUP BY 1

)

SELECT

```
state,
Average_freight_value,
CASE
WHEN highest_rnk <=5 THEN 'Highest Average Freight Value State'
WHEN lowest_rnk<=5 THEN 'Lowest Average Freight Value State'
END AS rank
```

FROM gbv4

WHERE highest\_rnk <=5 or lowest\_rnk<=5

ORDER BY 2 DESC;

Query results				
JOB INFORMATION RESULTS JSON EXECUTION DETAILS CHART PREVIEW				
Row	state	Average_freight_valu	rank	
1	RR	42.98	Highest Average Freight Value ...	
2	PB	42.72	Highest Average Freight Value ...	
3	RO	41.07	Highest Average Freight Value ...	
4	AC	40.07	Highest Average Freight Value ...	
5	PI	39.15	Highest Average Freight Value ...	
6	DF	21.04	Lowest Average Freight Value ...	
7	RJ	20.96	Lowest Average Freight Value ...	
8	MG	20.63	Lowest Average Freight Value ...	
9	PR	20.53	Lowest Average Freight Value ...	
10	SP	15.15	Lowest Average Freight Value ...	



**Key insights-** As per the analysis, 'RR' State has the Highest Average Freight value(42.98) and 'SP' State has the lowest Average Freight value(15.15) in top 5 states.



# Business Case Study



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

```
WITH gbv5 AS
(
    SELECT
        c.customer_state AS state,
        ROUND(AVG(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
DAY)),2) AS average_delivery_time,
        DENSE_RANK()OVER(ORDER BY AVG(DATE_DIFF(order_delivered_customer_date,
order_purchase_timestamp, DAY))DESC) AS highest_rnk,
        DENSE_RANK()OVER(ORDER BY AVG(DATE_DIFF(order_delivered_customer_date,
order_purchase_timestamp, DAY))) AS lowest_rnk
    FROM guru_bv_Ecommerce.customers c
    LEFT JOIN guru_bv_Ecommerce.orders o USING(customer_id)
    LEFT JOIN guru_bv_Ecommerce.order_items oi USING(order_id)
    GROUP BY 1 )
SELECT
    state,average_delivery_time,
    CASE
        WHEN highest_rnk <=5 THEN 'Highest Average Delivery Time State'
        WHEN lowest_rnk<=5 THEN 'Lowest Average Delivery Time State'
    END AS rank
FROM gbv5
WHERE highest_rnk <=5 or lowest_rnk<=5
ORDER BY 2 DESC;
```

Query results			
JOB INFORMATION			
Row	state	average_delivery_time	rank
1	RR	27.83	Highest Average Delivery Time ..
2	AP	27.75	Highest Average Delivery Time ..
3	AM	25.96	Highest Average Delivery Time ..
4	AL	23.99	Highest Average Delivery Time ..
5	PA	23.3	Highest Average Delivery Time ..
6	SC	14.52	Lowest Average Delivery Time ..
7	DP	12.5	Lowest Average Delivery Time ..
8	MD	11.52	Lowest Average Delivery Time ..
9	PR	11.48	Lowest Average Delivery Time ..
10	SP	8.26	Lowest Average Delivery Time ..



**Key insights-As per the analysis, 'RR' State has the Highest Average Delivery Time(27.83) and 'SP' State has the Lowest Average Delivery Time(8.26) in top 5 states.**





## Business Case Study



Q 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.

WITH gbv6 AS

```
(
SELECT
    c.customer_state AS state,
    ROUND(AVG(DATE_DIFF(order_delivered_customer_date,
order_purchase_timestamp, DAY)), 2) AS average_delivery_time,
    ROUND(AVG(DATE_DIFF(order_estimated_delivery_date ,
order_delivered_customer_date, DAY)), 2) AS average_estimated_delivery_time
    FROM guru_bv_Ecommerce.customers c
    LEFT JOIN guru_bv_Ecommerce.orders o USING(customer_id)
    LEFT JOIN guru_bv_Ecommerce.order_items oi USING(order_id)
    GROUP BY 1
),
```

gbv7 AS

```
(
SELECT
    State,
    average_delivery_time,
    average_estimated_delivery_time,
    ROUND((average_estimated_delivery_time-average_delivery_time), 2) AS
diff_estimated_delivery_time,
    DENSE_RANK()OVER(ORDER BY
(average_estimated_delivery_time-average_delivery_time) DESC) as highest_rank
    FROM gbv6 )
SELECT state,
    average_delivery_time,
    average_estimated_delivery_time,
    diff_estimated_delivery_time
    FROM gbv7
    WHERE highest_rank<=5
    ORDER BY 3;
```





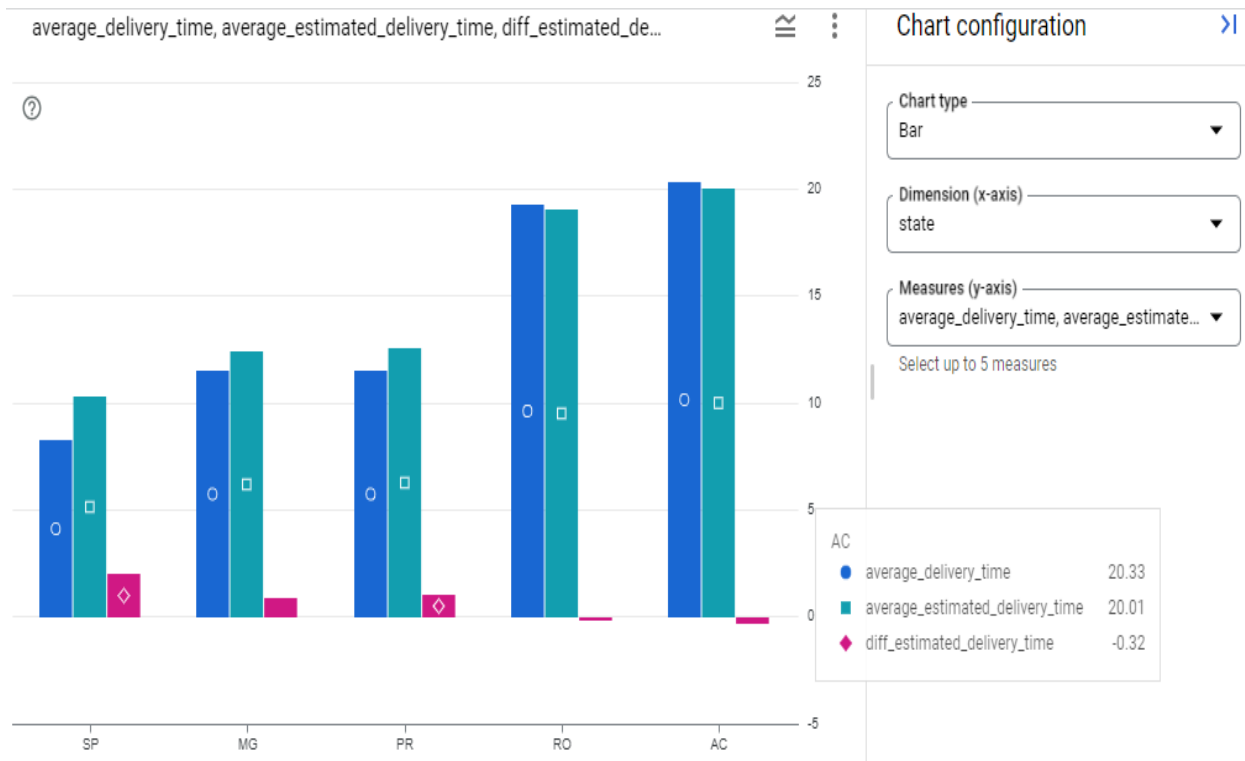
# Business Case Study



## Query results

SAVE

JOB INFORMATION	RESULTS	JSON	EXECUTION DETAILS	CHART	PREVIEW	EXI
Row	state	average_delivery_time	average_estimated_delivery_time	diff_estimated_delivery_time		
1	SP	8.26	10.27	2.01		
2	MG	11.52	12.4	0.88		
3	PR	11.48	12.53	1.05		
4	RO	19.28	19.08	-0.2		
5	AC	20.33	20.01	-0.32		



**Key insights-**As per the analysis, 'SP' State is the fastest Delivery Time when compared to average of actual delivery and estimated delivery ( 2.01 days) with average delivery time as 8.26 days(least delivery time among all states) and average estimated delivery as 10.27 days.



## Business Case Study



### Analysis based on the payments:

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

SELECT

p.payment\_type,

EXTRACT(MONTH FROM o.order\_purchase\_timestamp) AS month,

COUNT(DISTINCT o.order\_id) AS count\_of\_order

FROM guru\_bv\_Ecommerce.payments p

JOIN guru\_bv\_Ecommerce.orders o

USING(order\_id)

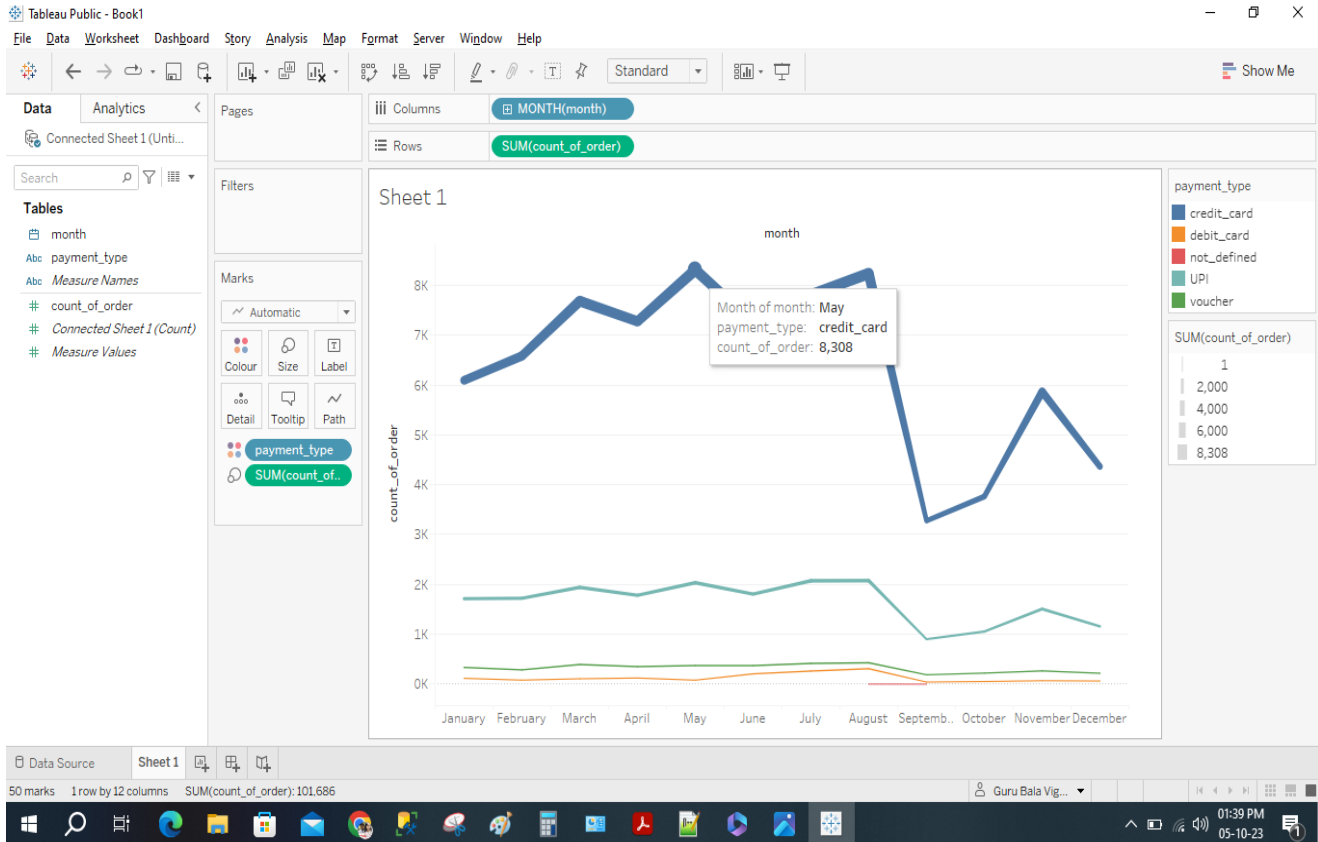
GROUP BY 1,2

ORDER BY 1,2;

Query results					CHART	PREVIEW
JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS		
Row	payment_type	month	count_of_order			
1	UPI	1	1715			
2	UPI	2	1723			
3	UPI	3	1942			
4	UPI	4	1783			
5	UPI	5	2035			
6	UPI	6	1807			
7	UPI	7	2074			
8	UPI	8	2077			
9	UPI	9	903			
10	UPI	10	1056			
11	UPI	11	1509			
12	UPI	12	1160			
13	credit_card	1	6093			
14	credit_card	2	6582			
15	credit_card	3	7682			
16	credit_card	4	7276			
17	credit_card	5	8308			



# Business Case Study



Columns: MONTH(month)

Rows: payment\_type

Sheet 2

payment_t..	January	February	March	April	May	June	July	August	September	October	November	December
credit_card	6,093	6,582	7,682	7,276	8,308	7,248	7,810	8,235	3,277	3,763	5,867	4,364
debit_card	118	82	109	124	81	208	264	311	43	54	70	64
not_defined								2	1			
UPI	1,715	1,723	1,942	1,783	2,035	1,807	2,074	2,077	903	1,056	1,509	1,160
voucher	337	288	395	353	374	373	417	430	189	223	267	220



**Key insights-**The analysis shows an overall uptrend from January to August and another uptrend from September to November. Credit card transactions are the most popular payment method with its peak attained by May with 8308, followed by UPI. Debit card transactions are the least preferred option. Notably, credit card transactions are rapidly increasing compared to other payment methods, possibly due to benefits like "buy now, pay later" options or cashback or conversion of EMI received using credit cards.



# Business Case Study



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

SELECT

p.payment\_installments,  
COUNT(DISTINCT o.order\_id) AS count\_of\_orders

FROM guru\_bv\_Ecommerce.payments p

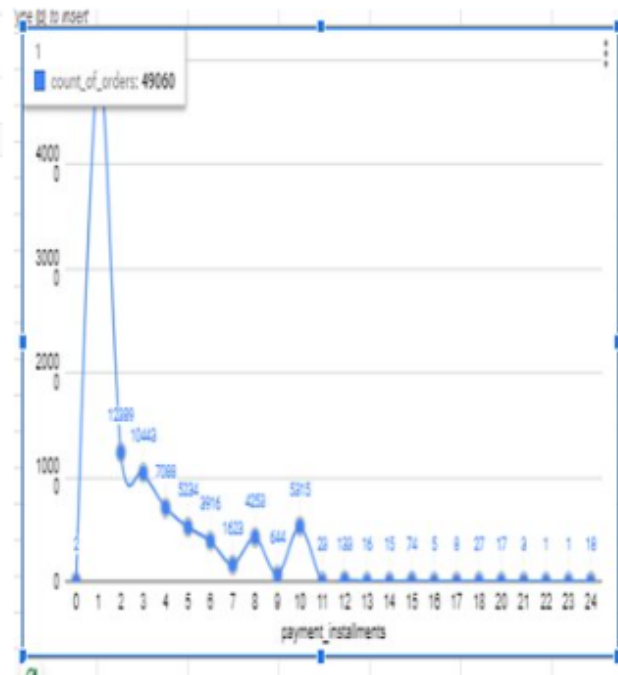
JOIN guru\_bv\_Ecommerce.orders o

USING(order\_id)

GROUP BY 1

ORDER BY 2 DESC;

Query results			
JOB INFORMATION		RESULTS	JSON
Row	payment_installment	count_of_orders	
1	1	49060	
2	2	12389	
3	3	10443	
4	4	7088	
5	10	5315	
6	5	5234	
7	8	4253	
8	6	3916	
9	7	1623	
10	9	644	
11	12	133	
12	15	74	
13	18	27	
14	11	23	
15	24	18	
16	20	17	



**Key insights-**As per the analysis, it reveals that the majority of orders (maximum count-49060) have only one payment installment. The highest number of installments is 24, which is associated with 18 orders. Understanding payment types and installment preferences is essential for businesses to optimize their payment processes and cater to customer preferences. By analyzing payment trends, companies can make informed decisions to improve payment options, streamline processes, and enhance the overall customer experience. It highlights the popularity of credit card transactions, the increasing trend of credit card usage, and the prevalence of single-payment installment orders. These insights can help businesses align their payment strategies and improve customer satisfaction.



## Business Case Study



### **Actionable Insights & Recommendations**

#### **Actionable Insights**

- The data reveals that the state of SP has significantly more orders than the other states. This indicates an opportunity for improvement in the other states. Focusing on these states can help increase the number of orders and expand the customer base.
- Improving delivery period in areas with longer delivery durations can have a positive impact on customer satisfaction and encourage repeat purchases. Streamlining logistics and implementing efficient shipping processes are key to achieving this.
- States like SP and RJ already have high order counts, it is recommended to focus on customer retention strategies, such as personalized marketing campaigns, loyalty programs, and exceptional customer service experiences.
- Data indicates a decline in orders during September to October. Offering discounts or promotions during off-peak seasons can incentivize customers to make purchases during these periods, thus boosting sales.
- Analysis provides valuable insights into payment types and installment preferences. It highlights the popularity of credit card transactions, the increasing trend of credit card usage, and the prevalence of single-payment installment orders. These insights can help businesses align their payment strategies and improve customer satisfaction.
- Analysis provides valuable insights on the freight charges for the delivery.
- Based on the analysis, we found that Brazilian customers tend to place most orders during the daytime, specifically in the afternoon and night. This analysis helps e-commerce businesses optimize their operations. By identifying peak buying times, companies can allocate resources, such as customer service representatives and inventory, more effectively to meet customer demands and provide a seamless shopping experience.



## Business Case Study



### Recommendations:

- Improve logistics and shipping processes to reduce delivery times and enhance customer satisfaction. This includes optimizing warehouse operations, refining shipping routes, and partnering with reliable courier services.
- Implementing customer retention strategies to encourage repeat purchases and foster loyalty. This can be achieved through loyalty programs, referral rewards, and personalized offers and also to extend the customers.
- Evaluating pricing and freight fees to ensure competitiveness in the market while maximizing revenue and profitability. Consider increasing prices or adjusting the freight fees to appropriate levels.
- Collaborate with sellers and negotiate on the price value of the products and adjusting the cost of the orders to appropriate levels.

### Key Takeaways

- The state of SP dominates the e-commerce market in Brazil, indicating the need to focus on other states for potential growth opportunities.
- Analyzing customer demographics can help tailor products and marketing strategies to specific target audiences, leading to increased sales.
- Offering discounts during off-peak seasons can incentivize customers and boost sales during slower periods.