Business Case: Ecommerce Company's Operations

Gain valuable insights into ecommerce company's operations in Brazil. The information shed light on various aspects of the business, such as order processing, pricing strategies, payment and shipping efficiency, customer demographics, product characteristics, and customer satisfaction levels.

Table of Content:

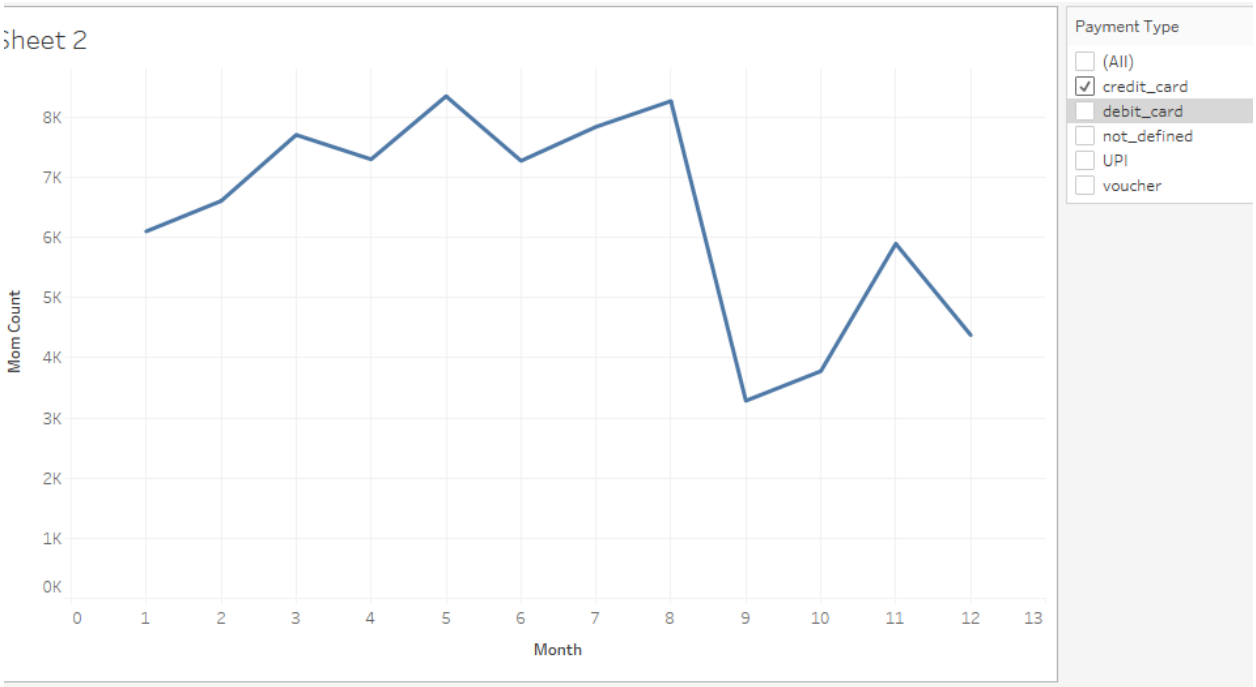
Actionable Insights and Recommendations

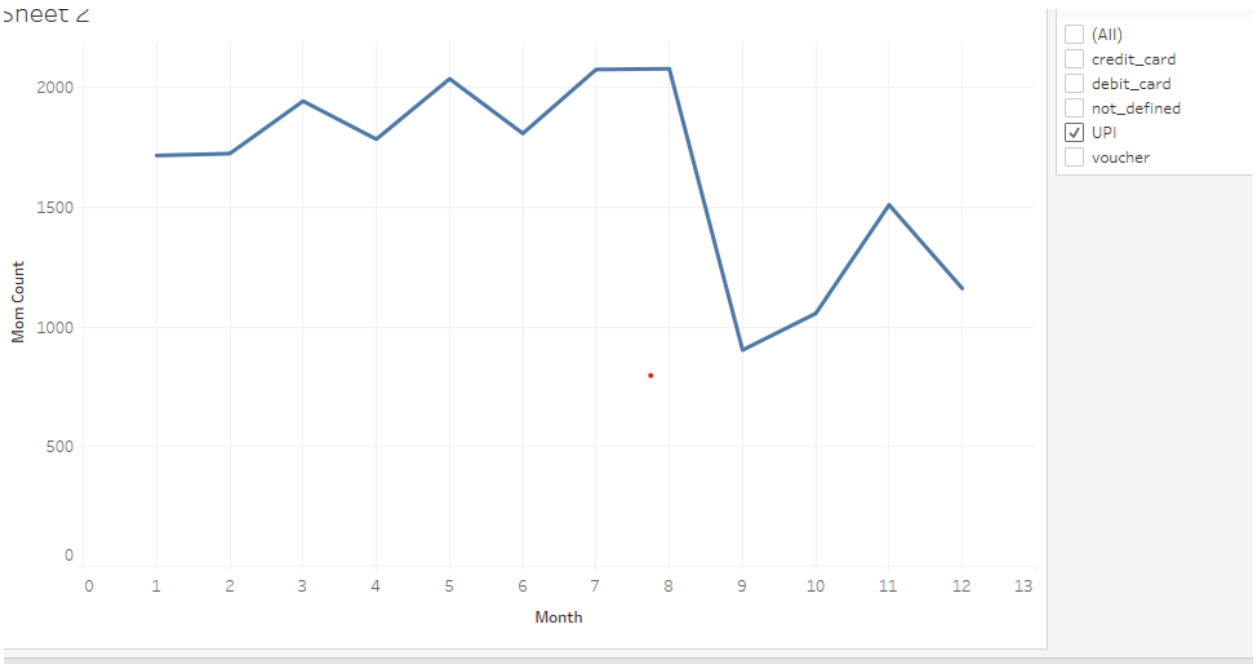
1. **Actionable insight**

```
-----prefered payment_type
SELECT DISTINCT p.payment_type,count(p.order_id) cnt
from `Target_Business_Case.payments` p
group by p.payment_type
order by count desc
```

tow	payment_type	cnt
1	credit_card	76795
2	UPI	19784
3	voucher	5775
4	debit_card	1529
5	not_defined	3

payment type "Credit card" has maximum orders and consistent month on month orders Followed by UPI payment type in consistency


Sheet 2
Mom Count
8K
7K
6K
5K
4K
3K
2K
1K
0K
0
1
2
3
4
5
6
7
8
9
10
11
12
13
Month
Payment Type
(All)
credit_card
debit_card
not_defined
UPI
voucher

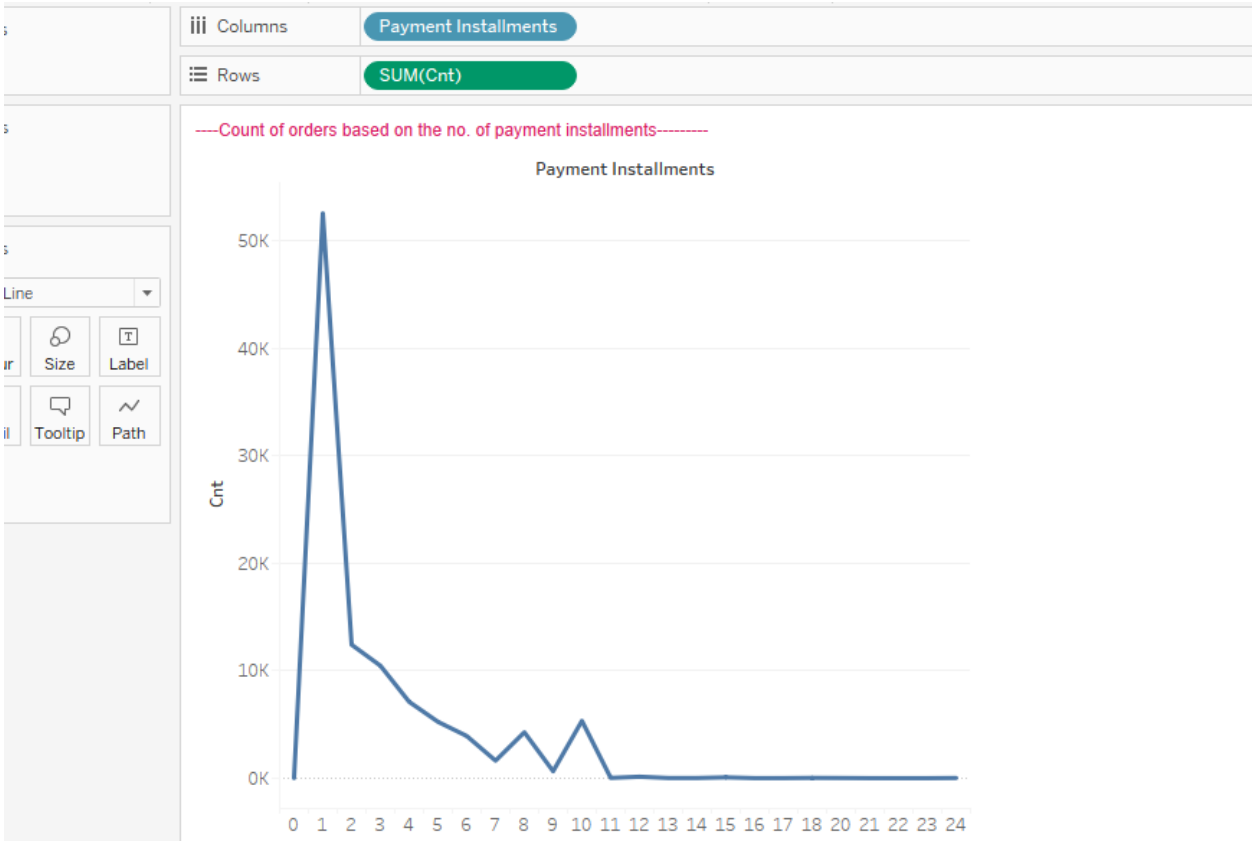

Sheet 2
Mom Count
2000
1500
1000
500
0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
Month
(All)
credit_card
debit_card
not_defined
UPI
voucher

Recommendation:

Target should promote these options to encourage more customers to use them. This can be done through targeted messaging, incentives, or highlighting the benefits of using these preferred payment types during the checkout process.

2. **Actionable Insight**

[Source Query]



maximum orders (count 52546) with payment installments =1
minimum orders (count 1) with payment installments =22 & 23
More the installments lesser is the number of orders

Recommendation:

Promote 1 installment payment type on credit cards by giving discount on one installment payment options or designing effective promotions or strategise the prices to incentivize customers to choose 1 payment installments.

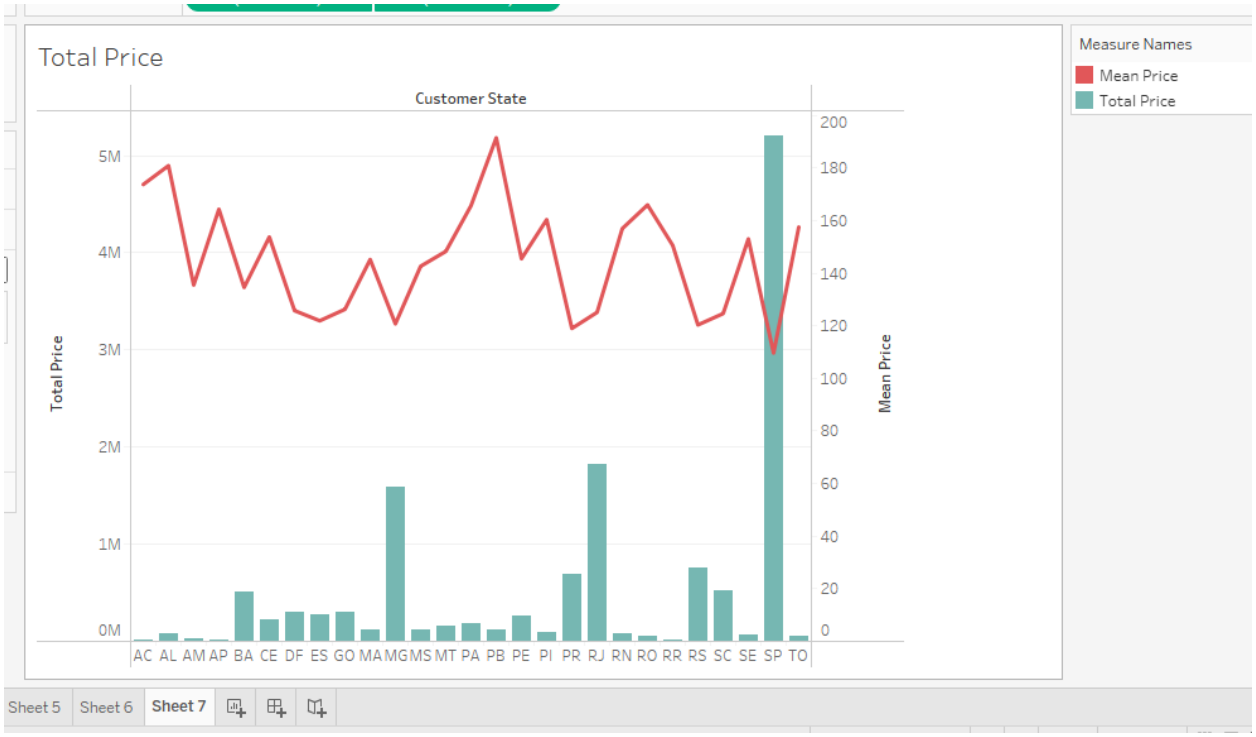
3.Actionable Insights

[Source query]

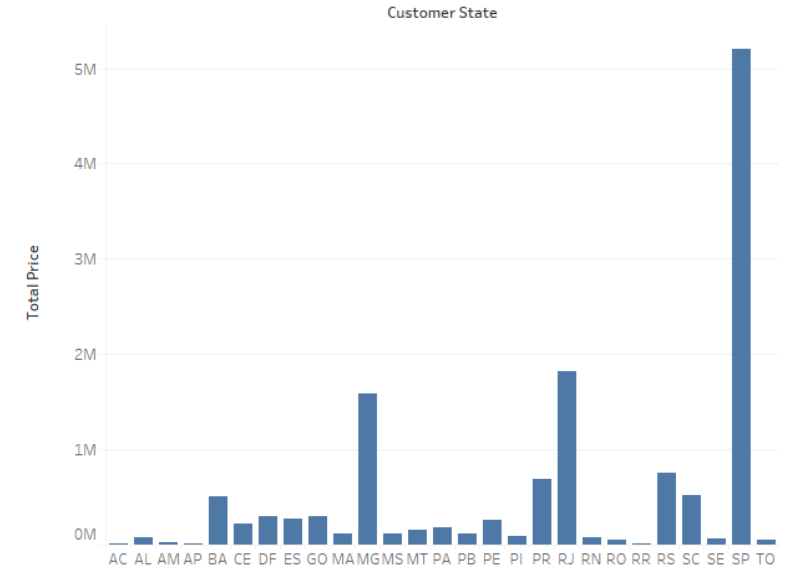
SP state has lowest avg freight_value and RR state has highest avg freight_value

SP state has highest avg time to delivery, MG has least avg time to delivery

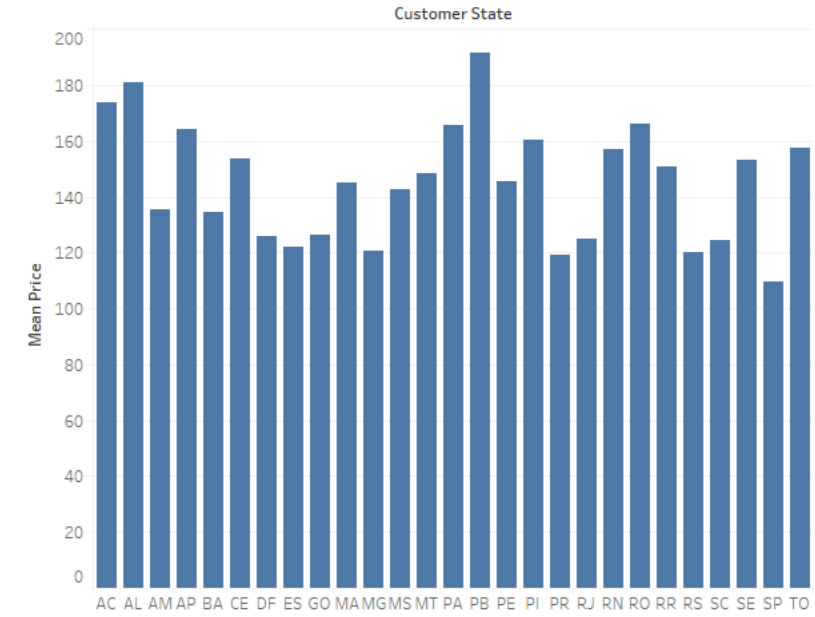
SP highest time to deliver with more estimated delivery time



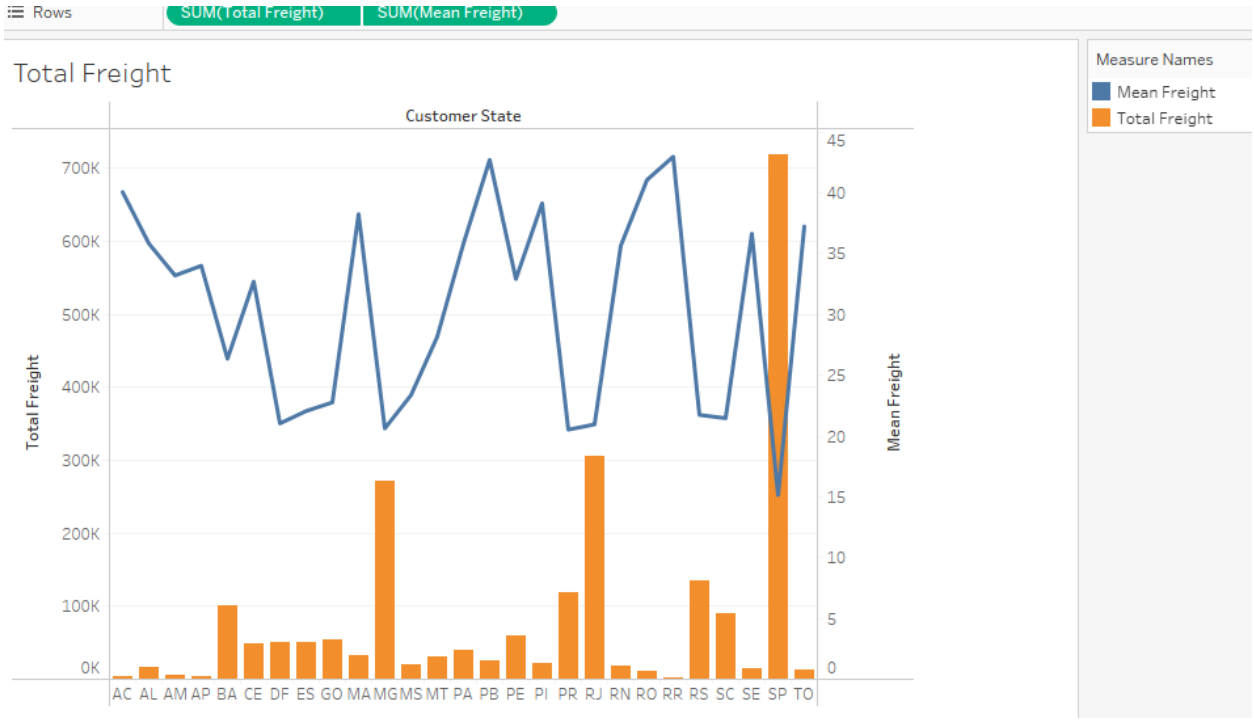
Total Price



Mean Price



Recommendation:
Consider adjusting prices in states with lower mean prices to improve profitability, or explore opportunities to increase prices in states with higher mean prices while monitoring customer response.



Recommendation:
Examine the mean freight value by customer state to identify states where shipping costs are higher or lower compared to the overall average. Use this insight to optimize your freight cost management. Explore options such as negotiating better shipping rates or implementing efficient logistics solutions to reduce costs in states with higher mean freight values.

4. Actionable Insight
Source query
Outcome: 58% increase in the cost of orders from 2017 to 2018

58% increase indicates a significant growth in revenue during this period. This insight highlights the overall positive trend and growth in the business.

5. Actionable Insight
Source query
Maximum customers from customer_state SP and Minimum customers from customer_state RR

Recommendation:
SP has a maximum customer base, allocate resources, marketing efforts, and operational capabilities to cater to the needs and preferences of customers.
RR has the minimum customer base followed by AP, AC, AM, RO.
Evaluate the feasibility of expanding your customer base and market presence in states with a smaller customer population, considering factors such as competition, logistics, and market demand.

6. Actionable Insight:

Source query

24 states had minimum sales in month 1 throughout years 2016,2017,2018
Maximum orders in month 10

Recommendation:
Develop region-specific promotions, discounts, or campaigns to drive sales during historically low-performing months.

Experiment with demand generation strategies during the months with historically low sales. Consider introducing new products or services, collaborating with complementary businesses, or running targeted advertising campaigns to stimulate customer interest and drive sales during these periods. Monitor the impact of these strategies and iterate based on customer response and sales performance.

7. **Actionable Insight:**

Source query

Maximum sales take place during afternoon followed by night, minimum sales are during dawn.

Recommendation:
Shift marketing efforts: Allocate more resources, such as advertising and promotions, during the afternoon and evening periods when sales are at their peak. This can help maximize visibility and engagement with customers during these high-sales periods.

Optimize staffing: Adjust employee schedules and staffing levels to align with the periods of maximum sales. Ensure that you have adequate staff during the afternoon and evening shifts to provide optimal customer service and handle higher transaction volumes.

Pricing strategies: consider adjusting pricing during these periods to optimize revenue generation.

Explore reasons behind low sales during dawn: Investigate the potential reasons behind the lower sales during the dawn period. Are there any operational constraints, customer preferences, or external factors contributing to this trend? By identifying the underlying causes, you can develop strategies to mitigate the impact and potentially increase sales during this period.

Analysis Report

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

Data type of columns in a table

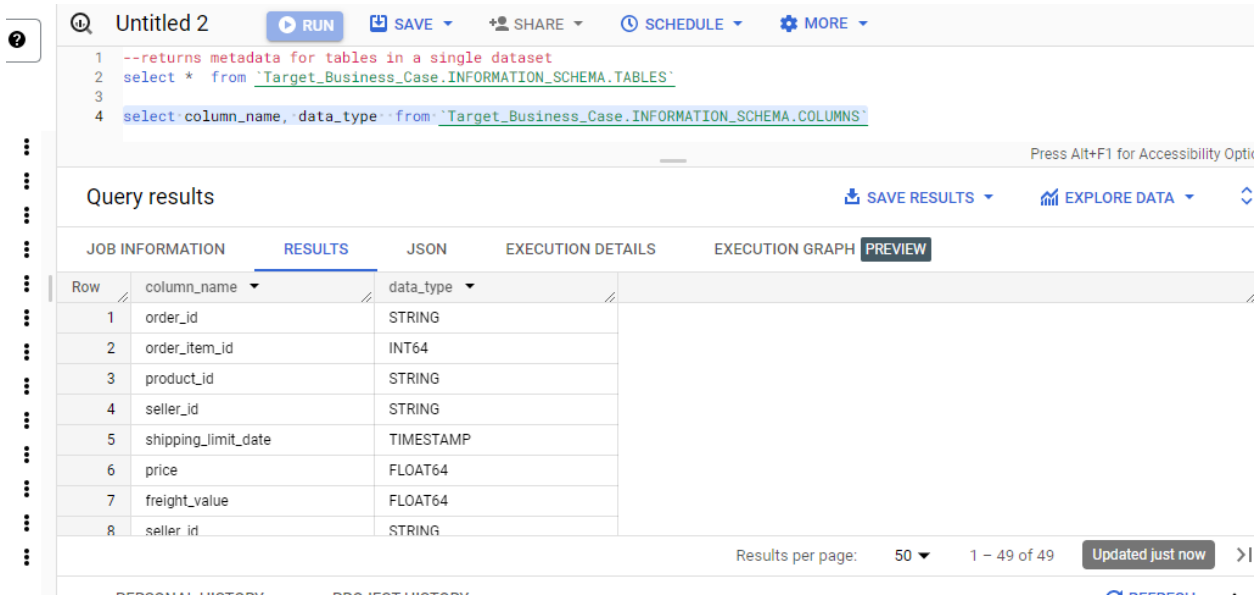
Time period for which the data is given

Cities and States of customers ordered during the given period

Solution:

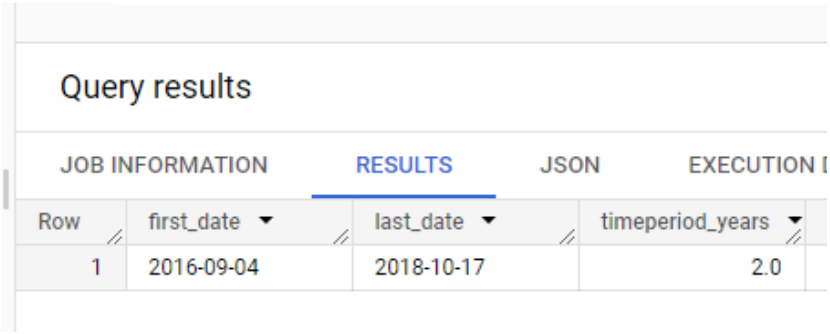
Data type of columns in a table

```
select column_name, data_type  from
`Target_Business_Case.INFORMATION_SCHEMA.COLUMNS`
```



Time period for which the data is given

```
SELECT MIN(date(order_purchase_timestamp)) first_date,
max(date(order_delivered_customer_date)) last_date,
round(timestamp_diff(max(order_delivered_customer_date),MIN(order_purc
hase_timestamp),day)/365) timeperiod_years
from `Target_Business_Case.orders`
```



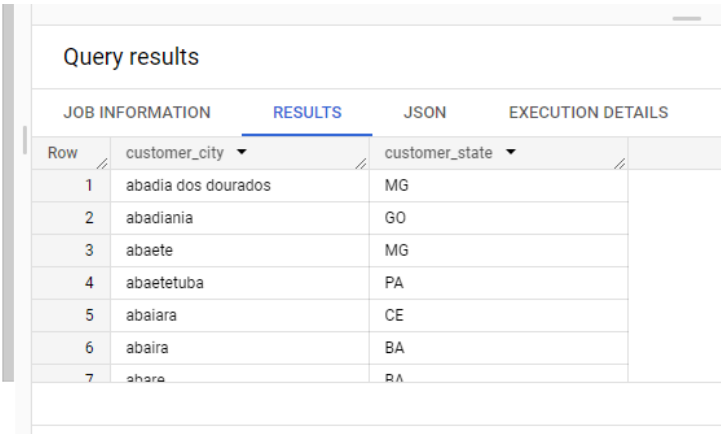
Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION I

Row	first_date	last_date	timeperiod_years
1	2016-09-04	2018-10-17	2.0

Cities and States of customers ordered during the given period

```
select customer_city, customer_state
from `Target_Business_Case.customers`
group by customer_city,customer_state
order by 1,2
```



Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	customer_city	customer_state
1	abadia dos dourados	MG
2	abadiania	GO
3	abaete	MG
4	abaetetuba	PA
5	abaiara	CE
6	abaira	BA
7	abare	BA

Question 2:

In-depth Exploration:

1. Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario? Can we see some seasonality with peaks at specific months?
2. What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

Solution:

Is there a growing trend on e-commerce in Brazil?

Yes, the total number of orders is increasing every year, there was a drastic increase in the orders from 2016 to 2017 , the count increased by factor 137. The major factor behind this drastic change is the sales started in the 10th month of 2016.

While 2017 to 2018 there was a slight increase in the number of orders, the count increased by factor 1.2

```
----------Yearly Trend----------
```

```
  select main.year, main.orders, round(main.orders/lag(orders)
over(order by main.year)) factor
  from
  (
  select temp.year,  count(temp.year) orders
  from
  (
    select order_id, order_purchase_timestamp, extract(year from
order_purchase_timestamp) year,
    extract(month from order_purchase_timestamp) month
    from `Target_Business_Case.orders`
  ) temp
  group by temp.year
  ) main
   order by  main.orders
```

Query results

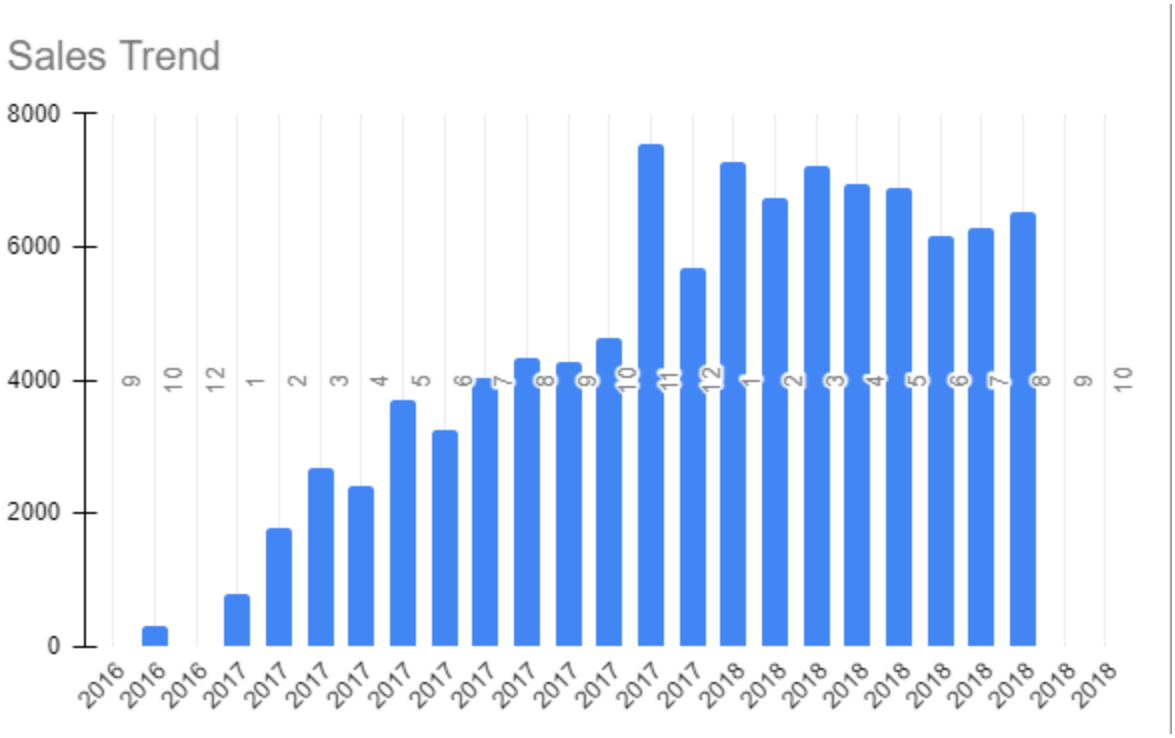
JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	year	orders	factor
1	2016	329	*null*
2	2017	45101	137.09
3	2018	54011	1.2

Can we see some seasonality with peaks at specific months?

Click here for actionable insights

In 2017, sales started in the 9th month and increased in the 10th month. There were no sales in the 11th month and 1 order in the 12th month.
In 2017, sales increased monthly in 2017, it was maximum in 11th month of 2017 then it decreased in 12th month.
In 2018, sales were maximum in 1st month then decreased in 2nd month, again rising in 3rd month followed by decline for next 3 months. It was raised again for the next two months.




```
----------Monthly Trend---------------
  select temp.year, temp.month, count(temp.year) orders,
  dense_rank() over(partition by temp.year order by count(temp.year)
desc ) rnk
  from
  (
    select order_id, order_purchase_timestamp, extract(year from
order_purchase_timestamp) year,
    extract(month from order_purchase_timestamp) month
    from `Target_Business_Case.orders`
  ) temp
  group by temp.year, temp.month
  order by temp.year, rnk
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | EXEC

Row	year	month	orders	rnk
1	2016	10	324	1
2	2016	9	4	2
3	2016	12	1	3
4	2017	11	7544	1
5	2017	12	5673	2
6	2017	10	4631	3
7	2017	8	4331	4
8	2017	9	4285	5
9	2017	7	4026	6
10	2017	5	3700	7
11	2017	6	3245	8
12	2017	3	2682	9
13	2017	4	2404	10

What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

Outcome:
Maximum sales take place during afternoon followed by night, minimum sales are during dawn.
Click here for actionable insights

```
------------Sales trend by hours--------------
select temp.TimeOfDay, count(temp.TimeOfDay) as count_orders
from
(
 SELECT
```

```
  order_id, order_purchase_timestamp,
  CASE
    WHEN EXTRACT(HOUR FROM order_purchase_timestamp) >= 0 AND
EXTRACT(HOUR FROM order_purchase_timestamp) < 6 THEN 'Dawn'
    WHEN EXTRACT(HOUR FROM order_purchase_timestamp) >= 6 AND
EXTRACT(HOUR FROM order_purchase_timestamp) < 12 THEN 'Morning'
    WHEN EXTRACT(HOUR FROM order_purchase_timestamp) >= 12 AND
EXTRACT(HOUR FROM order_purchase_timestamp) < 18 THEN 'Afternoon'
    ELSE 'Night'
  END AS TimeOfDay
FROM
`Target_Business_Case.orders`
) temp
group by temp.TimeOfDay
order by count_orders
```

Query results

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

Row	TimeOfDay	count_orders
1	Dawn	4740
2	Morning	22240
3	Night	34100
4	Afternoon	38361

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

1. **Get month on month orders by states**
2. **Distribution of customers across the states in Brazil**

Solution:

<u>Get month on month orders by states</u>

Outcomes:

1. The 9th month has lowest sales for 19 states out of 27 states
2. **The 5th month has 9 states out of 27 states.**
3. **Maximum orders in 8th month irrespective of states**

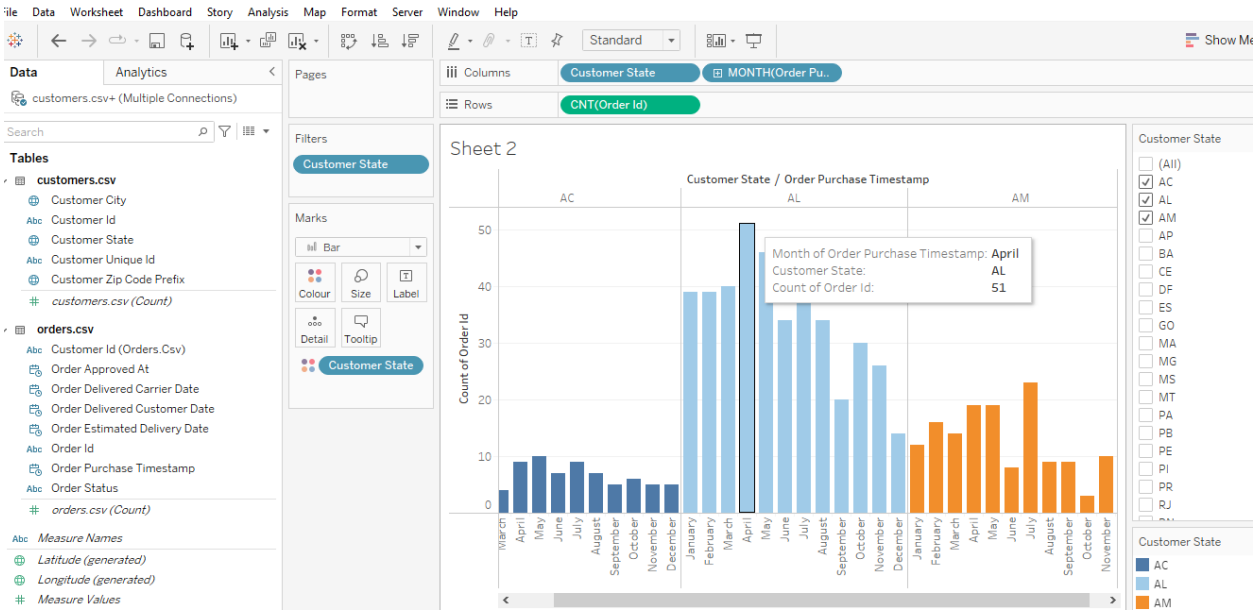
Query 1:

```
select main.*, dense_rank() over(partition by main.customer_state
order by main.month_orders) rnk
    from
    (
      select temp.customer_state,temp.month, count(temp.month)
month_orders
      from
      (


          select  c.customer_state,o.order_id,
o.order_purchase_timestamp,
          extract(month from o.order_purchase_timestamp) month
          from `Target_Business_Case.customers` c
          left join `Target_Business_Case.orders` o on
o.customer_id=c.customer_id
      ) temp


      group by temp.customer_state, temp.month
      order by temp.customer_state, temp.month
    ) main
```

Query results

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

Row	customer_state	month	month_orders	rnk
3	AL	11	26	3
4	AL	10	30	4
5	AL	6	34	5
6	AL	8	34	5
7	AL	2	39	6
8	AL	1	39	6
9	AL	7	40	7
10	AL	3	40	7
11	AL	5	46	8
12	AL	4	51	9



Query 3:

The 9th month has lowest sales for 19 states out of 27 states

```
 -------count number of states with rank=1 i.e minimum number of order
IDs for that month-----------
   select final.month, count(final.rnk) count_States_min_orders
   from
   ( select main.customer_state, main.month, main.order_id_count,
     dense_rank() over(partition by main.customer_state order by
main.order_id_count ) rnk
    from
      (
        select temp.customer_state,temp.month, count(temp.order_id)
order_id_count
        from
        (


            select  c.customer_state,o.order_id,
o.order_purchase_timestamp,
            extract(month from o.order_purchase_timestamp) month
            from `Target_Business_Case.customers` c
            left join `Target_Business_Case.orders` o on
o.customer_id=c.customer_id
        ) temp
        group by temp.customer_state, temp.month

      ) main order by main.customer_state
   )final
    where final.rnk=1
    group by final.month
    order by count_States_min_orders
```

Query results

JOB INFORMATION | RESULTS | JSON | EXEC

Row	month	count_States_min_orders
1	11	1
2	1	1
3	3	1
4	12	4
5	10	5
6	9	19

-------count number of states with rank=1 i.e maximum number of order IDs for that month----------

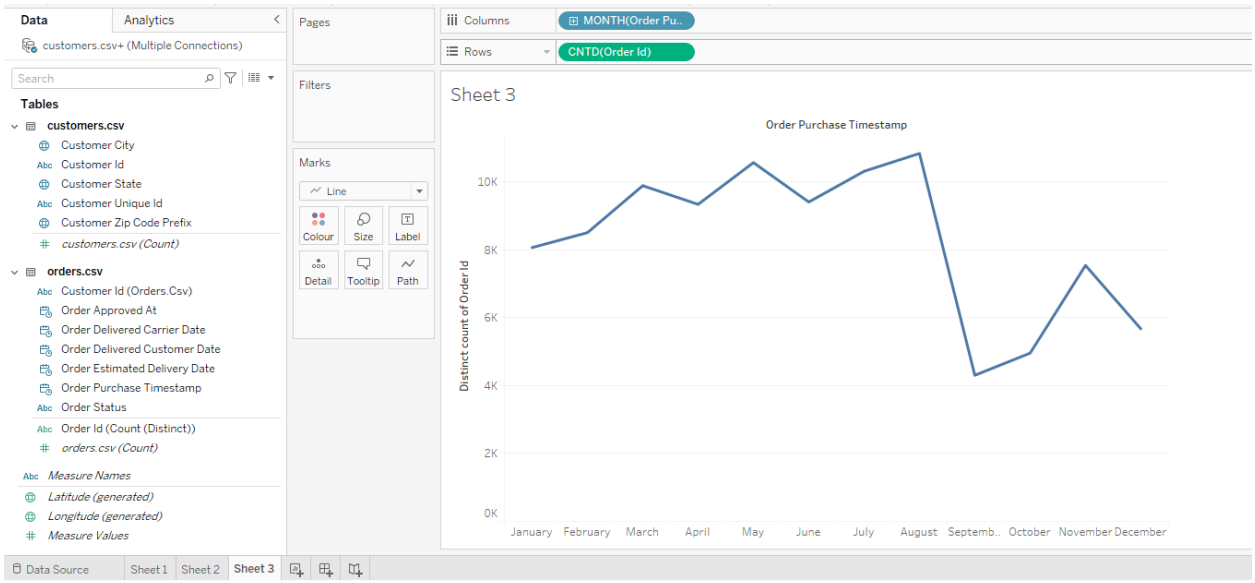
The 5th month has 9 states out of 27 states.

Query results

JOB INFORMATION | RESULTS | JSON | EXEC

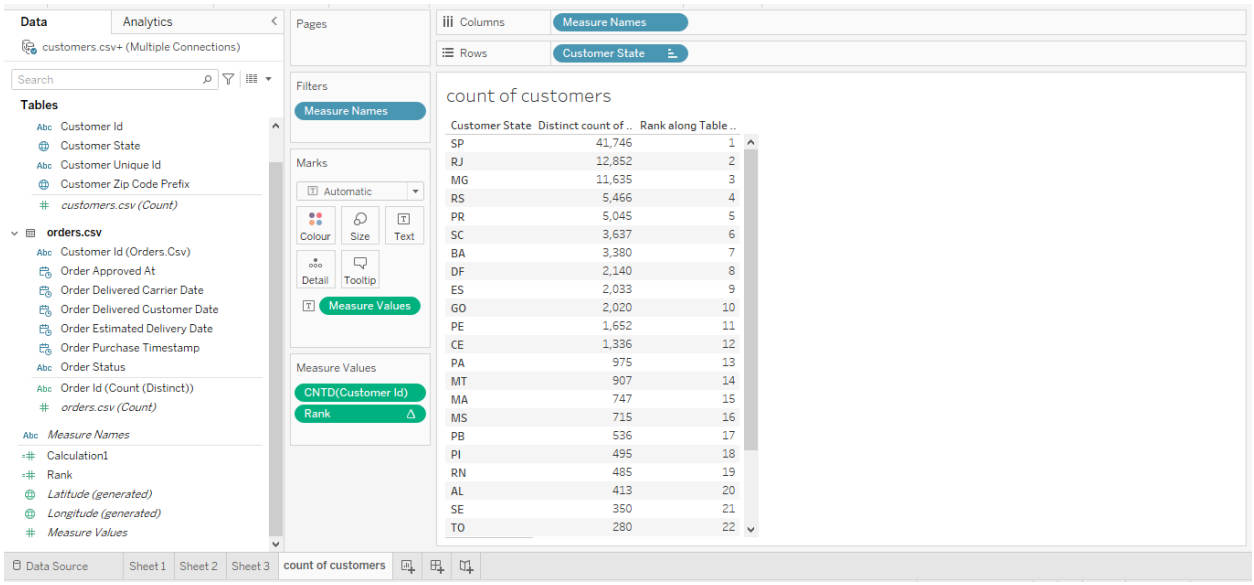
Row	month	count_States_max_orders
1	6	1
2	1	1
3	4	2
4	8	4
5	3	6
6	7	7
7	5	9

Maximum Number of Sales in August/8th month irrespective of the states



Question 4:
Distribution of customers across the states in Brazil
Solution:
Outcome: Maximum customers from customer_state SP and Minimum customers from customer_state RR
Click here for actionable insights

```
----Distribution of customers across states in Brazil----
select customer_state, count(customer_id) number_of_customers
from `Target_Business_Case.customers`
group by customer_state
order by number_of_customers desc
```



Question 4:

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

1. **Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only) - You can use “payment_value” column in payments table**
2. **Mean & Sum of price and freight value by customer state**

Solution:

Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only) - You can use “payment_value” column in payments table

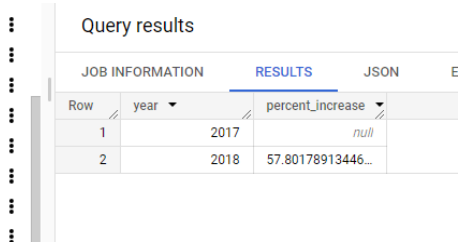
Outcome: 58% increase in the cost of orders from 2017 to 2018

Click here for actionable insights

Querry:

```
select main.year, ((main.cost_of_orders-lag(main.cost_of_orders)
over(order by main.year ))/main.cost_of_orders)*100 percent_increase
from
(
select distinct(temp.year), sum(temp.payment_value) over(partition by
temp.year) cost_of_orders
from
```

```
(
  select o.order_id, extract(year from o.order_purchase_timestamp)
year, extract(month from o.order_purchase_timestamp) month,
  p.payment_value
  from `Target_Business_Case.orders` o
  left join `Target_Business_Case.payments` p on o.order_id=p.order_id
  where extract(month from o.order_purchase_timestamp) between 1 and 8
and extract(year from o.order_purchase_timestamp)
  between  2017 and 2018
)temp
)main
```



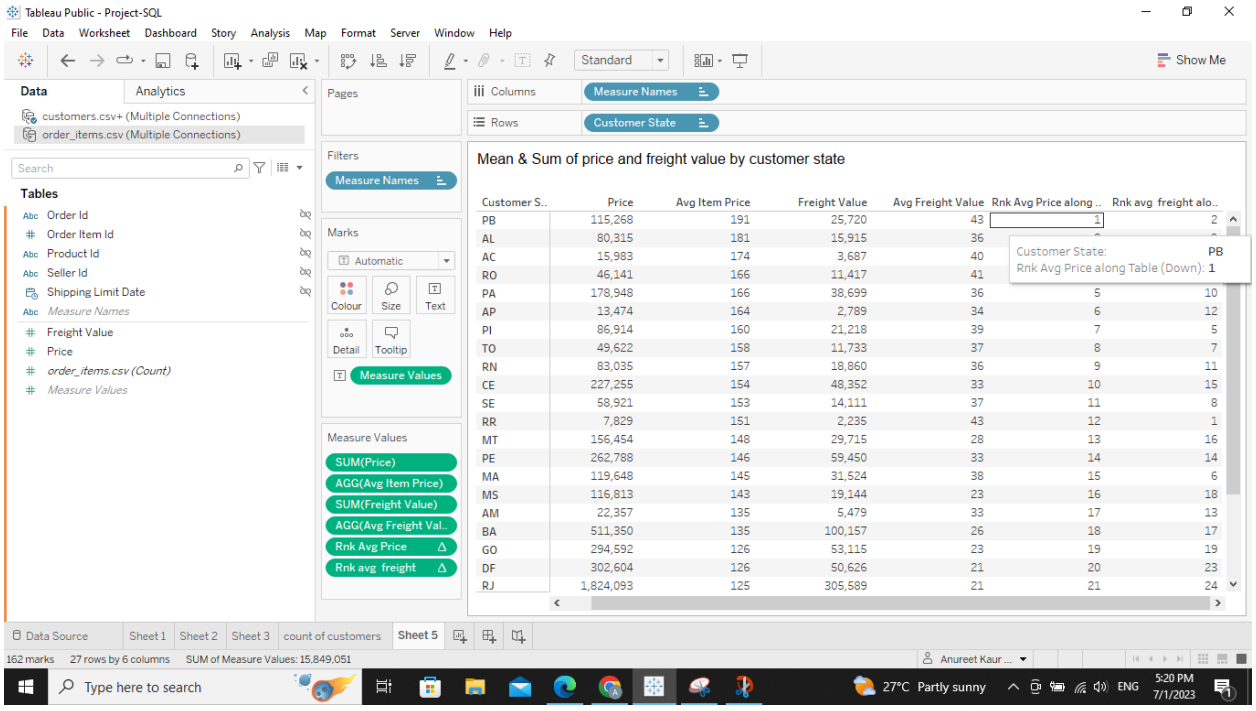
Mean & Sum of price and freight value by customer state

```
---Mean & Sum of price and freight value by customer state
select distinct(c.customer_state),  round(avg(o_items.price),2)
mean_price,round(sum(o_items.price),2) total_price,
round(avg(o_items.freight_value),2)
mean_freight,round(sum(o_items.freight_value),2) total_freight
from `Target_Business_Case.order_items` o_items
left join `Target_Business_Case.orders` o on
o.order_id=o_items.order_id
left join `Target_Business_Case.customers` c on
c.customer_id=o.customer_id
group by c.customer_state
order by mean_price desc
```

Query results

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

Row	customer_state	mean_price	total_price	mean_freight	total_freight
1	PB	191.48	115268.08	42.72	25719.73
2	AL	180.89	80314.81	35.84	15914.59
3	AC	173.73	15982.95	40.07	3686.75
4	RO	165.97	46140.64	41.07	11417.38
5	PA	165.69	178947.81	35.83	38699.3
6	AP	164.32	13474.3	34.01	2788.5
7	PI	160.36	86914.08	39.15	21218.2
8	TO	157.53	49621.74	37.25	11732.68
9	RN	156.97	83034.98	35.65	18860.1



Question 5:

Analysis on sales, freight and delivery time

1. **Calculate days between purchasing, delivering and estimated delivery**
2. **Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:**
    - **time_to_delivery = order_delivered_customer_date-order_purchase_timestamp**
    - **diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date**
3. **Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery**
4. **Sort the data to get the following:**
5. **Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5**
6. **Top 5 states with highest/lowest average time to delivery**
7. **Top 5 states where delivery is really fast/ not so fast compared to estimated date**

Solution:

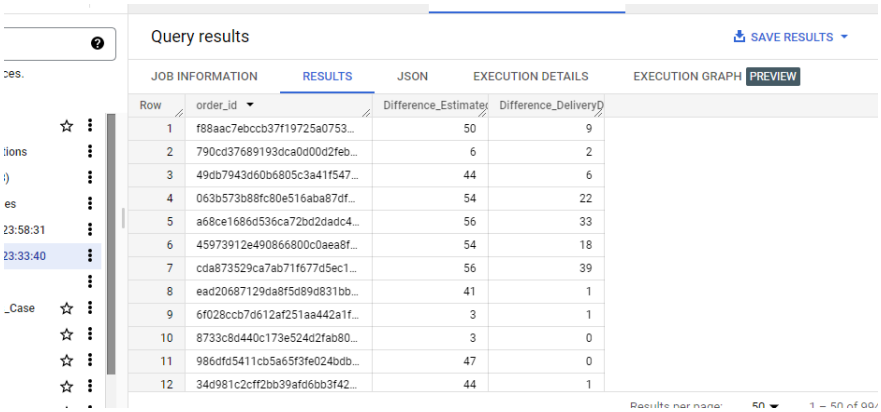
1. Calculate days between purchasing, delivering and estimated delivery

Outcome:

State SP has minimum delivery time and minimum average freight price
State RR maximum avg freight price and shipping time maximum

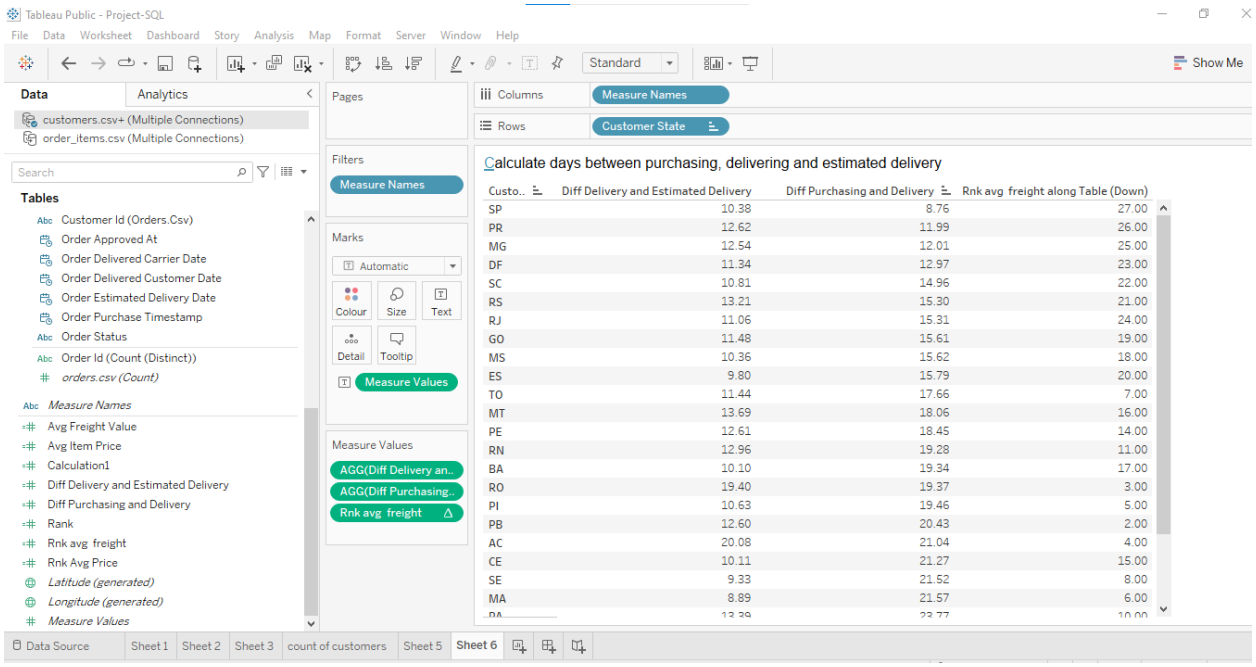
Click here for actionable insight

```
----Calculate days between purchasing, delivering and estimated
delivery
select order_id,
date_diff(order_estimated_delivery_date,order_purchase_timestamp,day)
Difference_EstimatedDeliveryDate_PurchaseDate,
date_diff(order_delivered_carrier_date, order_purchase_timestamp,day)
Difference_DeliveryDate_PurchaseDate
from `Target_Business_Case.orders`
```



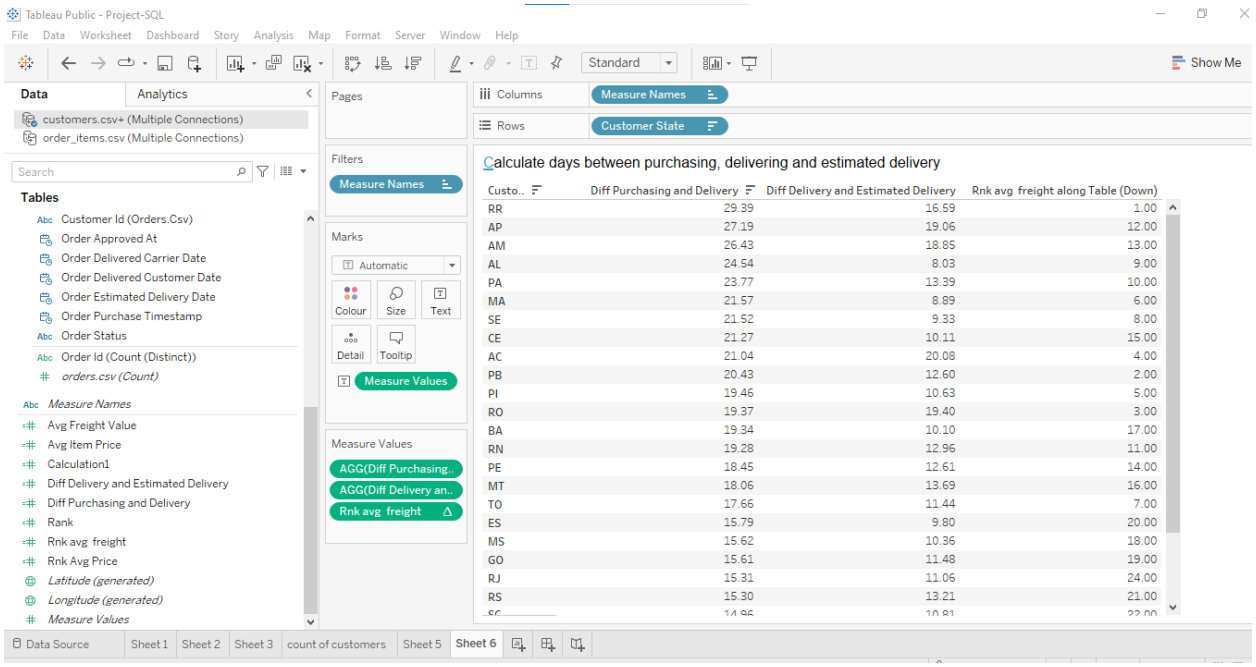
Row	order_id	Difference_Estimate	Difference_Delivery
1	f88aac7ebccb37f19725a0753...	50	9
2	790cd37689193dca0d00d2feb...	6	2
3	49db7943d60b6805c3a41f547...	44	6
4	063b573b88fc80e516aba87df...	54	22
5	a68ce1686d536ca72bd2dadc4...	56	33
6	45973912e490866800c0aea8f...	54	18
7	cda873529ca7ab71f677d5ec1...	56	39
8	ead20687129da8f5d89d831bb...	41	1
9	6f028ccb7d612af251aa442a1f...	3	1
10	8733c8d440c173e524d2fab80...	3	0
11	986dfd5411cb5a65f3fe024bdb...	47	0
12	34d981c2cff2bb39afd6bb3f42...	44	1

State SP has minimum delivery time and minimum avg freight price



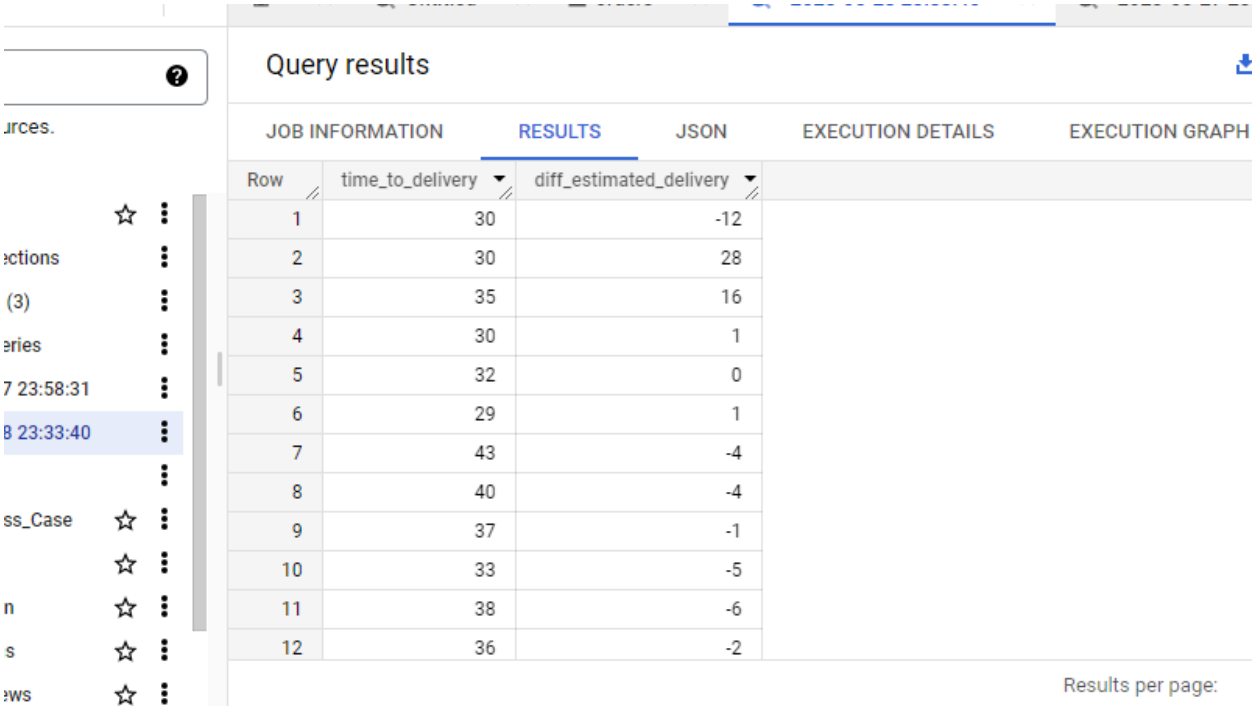
Custo..	Diff Delivery and Estimated Delivery	Diff Purchasing and Delivery	Rnk avg freight along Table (Down)
SP	10.38	8.76	27.00
PR	12.62	11.99	26.00
MG	12.54	12.01	25.00
DF	11.34	12.97	23.00
SC	10.81	14.96	22.00
RS	13.21	15.30	21.00
RJ	11.06	15.31	24.00
GO	11.48	15.61	19.00
MS	10.36	15.62	18.00
ES	9.80	15.79	20.00
TO	11.44	17.66	7.00
MT	13.69	18.06	16.00
PE	12.61	18.45	14.00
RN	12.96	19.28	11.00
BA	10.10	19.34	17.00
RO	19.40	19.37	3.00
PI	10.63	19.46	5.00
PB	12.60	20.43	2.00
AC	20.08	21.04	4.00
CE	10.11	21.27	15.00
SE	9.33	21.52	8.00
MA	8.89	21.57	6.00

State RR maximum avg freight price and shipping time maximum



2. **Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:**
    - **time_to_delivery = order_delivered_customer_date-order_purchase_timestamp**
    - **diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date**

```
----------Find time_to_delivery & diff_estimated_delivery----------
select
timestamp_diff(order_delivered_customer_date,order_purchase_timestamp,
day) time_to_delivery,
timestamp_diff(order_estimated_delivery_date,order_delivered_customer_
date,day) diff_estimated_delivery
from `Target_Business_Case.orders`
```



Query results

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

Row	time_to_delivery	diff_estimated_delivery
1	30	-12
2	30	28
3	35	16
4	30	1
5	32	0
6	29	1
7	43	-4
8	40	-4
9	37	-1
10	33	-5
11	38	-6
12	36	-2

Results per page:

3. **Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery**

```
--------------Group data by state, take mean of freight_value,
time_to_delivery, diff_estimated_delivery-----------
select temp.customer_state, round(avg(temp.freight_value),2)
mean_freight_value,
round(avg(temp.time_to_delivery),2) mean_time_to_delivery,
round(avg(temp.diff_estimated_delivery),2)
mean_diff_estimated_delivery
from
(
select
o.order_id,timestamp_diff(o.order_delivered_customer_date,o.order_purc
hase_timestamp,day) time_to_delivery,
timestamp_diff(o.order_estimated_delivery_date,o.order_delivered_custo
mer_date,day) diff_estimated_delivery ,
c.customer_state, t.freight_value
```

```
from `Target_Business_Case.orders` as o
left join `Target_Business_Case.customers` c on
c.customer_id=o.customer_id
left join `Target_Business_Case.order_items` t on
o.order_id=t.order_id
) temp
group by temp.customer_state
```

4. **Sort the data to get the following:**
    - **Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5**

SP state has lowest avg freight_value and RR state has highest avg freight_value

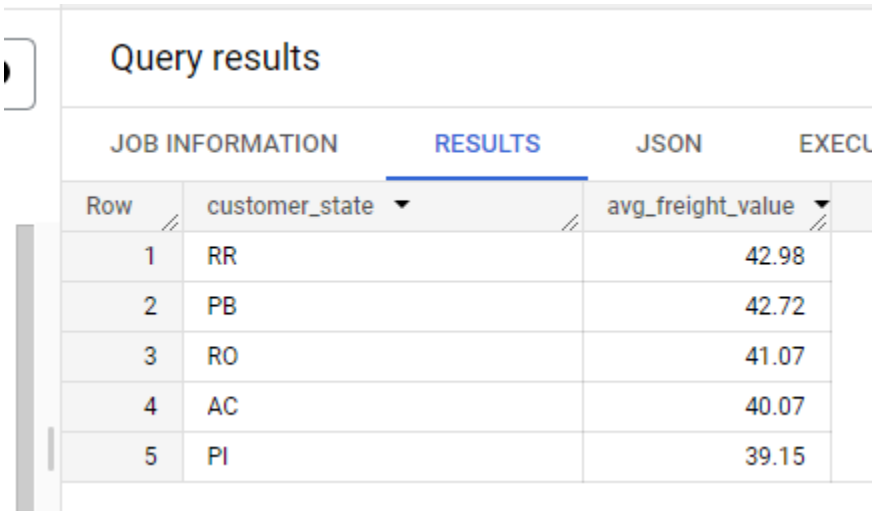
```
select
c.customer_state, round(avg(t.freight_value),2) avg_freight_value
from `Target_Business_Case.orders` as o
left join `Target_Business_Case.customers` c on
c.customer_id=o.customer_id
left join `Target_Business_Case.order_items` t on
o.order_id=t.order_id
group by c.customer_state
order by avg_freight_value
limit 5
```

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

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

```
---Top 5 states with highest average freight value --------
select
c.customer_state, round(avg(t.freight_value),2) avg_freight_value
from `Target_Business_Case.orders` as o
left join `Target_Business_Case.customers` c on
c.customer_id=o.customer_id
```

```
left join `Target_Business_Case.order_items` t on
o.order_id=t.order_id
group by c.customer_state
order by avg_freight_value desc
limit 5
```



Query results

JOB INFORMATION | RESULTS | JSON | EXECU

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

- **Top 5 states with highest/lowest average time to delivery**

SP state has highest avg time to delivery, MG has least avg time to delivery

```
------Top 5 states with highest/lowest average time to delivery----
--highest average time to delivery---
select c.customer_state,
concat(round(avg(timestamp_diff(o.order_delivered_customer_date,o.orde
r_purchase_timestamp,day))),' days') avg_time_to_delivery
from `Target_Business_Case.orders` as o
left join `Target_Business_Case.customers` c on
c.customer_id=o.customer_id
group by c.customer_state
order by avg_time_to_delivery desc
limit 5
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	customer_state	avg_time_to_delivery
1	SP	8 days
2	RR	29 days
3	AP	27 days
4	AM	26 days
5	AL	24 days

```
--lowest average time to delivery---
select c.customer_state,
concat(round(avg(timestamp_diff(o.order_delivered_customer_date,o.orde
r_purchase_timestamp,day))),' days') avg_time_to_delivery
from `Target_Business_Case.orders` as o
left join `Target_Business_Case.customers` c on
c.customer_id=o.customer_id
group by c.customer_state
order by avg_time_to_delivery
limit 5
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	customer_state	avg_time_to_delivery
1	MG	12 days
2	PR	12 days
3	DF	13 days
4	SC	14 days
5	RJ	15 days

5. **Top 5 states where delivery is really fast/ not so fast compared to estimated date**
   SP highest time to deliver with more estimated delivery time

```
select temp.customer_state, temp.time_to_delivery,
temp.diff_estimated_delivery
from
(
select
o.order_id,concat(timestamp_diff(o.order_delivered_customer_date,o.ord
er_purchase_timestamp,day),'days') time_to_delivery,
concat(timestamp_diff(o.order_estimated_delivery_date,o.order_delivere
d_customer_date,day),'days') diff_estimated_delivery ,
c.customer_state
from `Target_Business_Case.orders` as o
left join `Target_Business_Case.customers` c on
c.customer_id=o.customer_id
) temp
where temp.time_to_delivery< temp.diff_estimated_delivery
order by temp.time_to_delivery desc
limit 5
```

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

Row	customer_state	time_to_delivery	diff_estimated_delivery
1	SP	8days	9days
2	SP	8days	9days
3	RJ	8days	9days
4	SP	8days	9days
5	SP	8days	9days

6. **Payment type analysis:**

   [Click here for Actionable Insight]

1. **Month over Month count of orders for different payment types**

```
-----Month over Month count of orders for different payment types
SELECT DISTINCT p.payment_type, EXTRACT(MONTH FROM
o.order_purchase_timestamp) month,
       COUNT(p.payment_type) OVER (PARTITION BY p.payment_type,
EXTRACT(MONTH FROM o.order_purchase_timestamp) ORDER BY EXTRACT(MONTH
FROM o.order_purchase_timestamp)) AS mom_count
FROM `Target_Business_Case.orders` o
LEFT JOIN `Target_Business_Case.payments` p ON p.order_id = o.order_id
order by p.payment_type, month
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | E

Row	payment_type	month	mom_count
1	*null*	9	0
2	UPI	1	1715
3	UPI	2	1723
4	UPI	3	1942
5	UPI	4	1783
6	UPI	5	2035
7	UPI	6	1807
8	UPI	7	2074
9	UPI	8	2077
10	UPI	9	903
11	UPI	10	1056
12	UPI	11	1509

2. **Count of orders based on the no. of payment installments**

Outcome:

maximum orders (count 52546) with payment installments =1

minimum orders (count 1) with payment installments =22 & 23

More the installments lesser is the number of orders

```
----Count of orders based on the no. of payment installments---------
select distinct payment_installments, count(order_id) count
```

```
from `Target_Business_Case.payments`
group by payment_installments
order by payment_installments
```

Query results

JOB INFORMATION | RESULTS | JSON | E

Row	payment_installment	cnt
1	0	2
2	1	52546
3	2	12413
4	3	10461
5	4	7098
6	5	5239
7	6	3920
8	7	1626
9	8	4268
10	9	644
11	10	5328
12	11	23