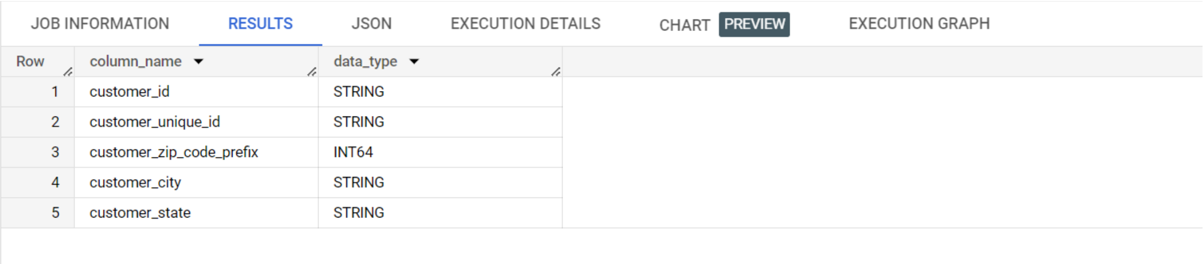
Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset:

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

```
select column_name,data_type from `target_sql`.INFORMATION_SCHEMA.COLUMNS
where table_name='customers'
```

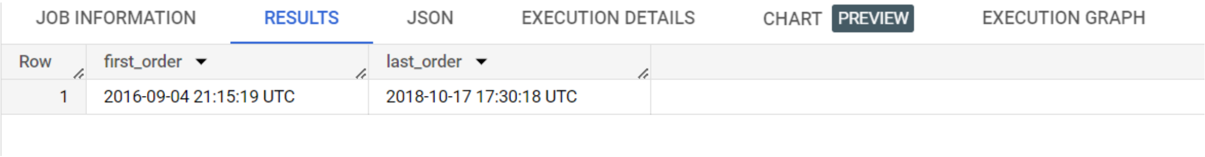


Row	column_name	data_type
1	customer_id	STRING
2	customer_unique_id	STRING
3	customer_zip_code_prefix	INT64
4	customer_city	STRING
5	customer_state	STRING

Inference: Information on data type of the columns in the customers table.

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

```
select min(order_purchase_timestamp) first_order,max(order_purchase_timestamp)
last_order from `target_sql.orders`
```



Row	first_order	last_order
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC

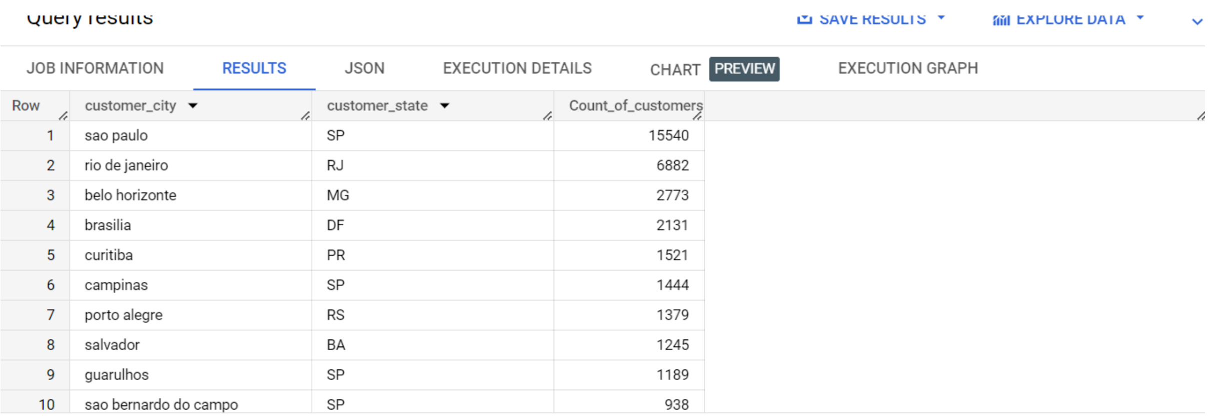
Inference: Date range between which orders were placed i.e. first order was placed on 4th September 2016 and the last order was placed on 17th October 2018.

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

```
select distinct c.customer_city,c.customer_state,count(c.customer_id) Count_of_customers
from `target_sql.customers` c

join `target_sql.orders` o on o.customer_id=c.customer_id

group by c.customer_state,c.customer_city
order by 3 desc
```



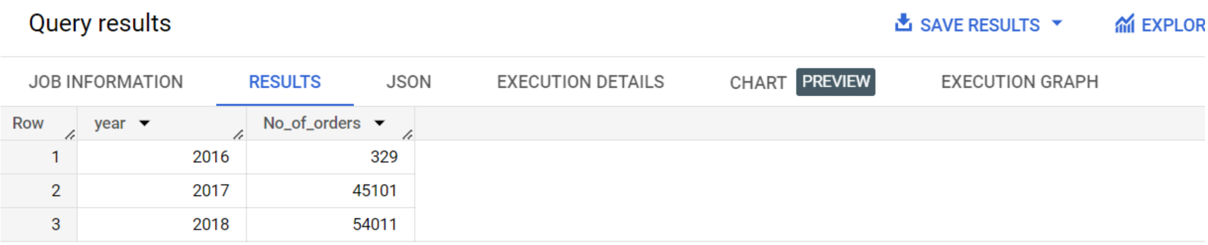
Row	customer_city	customer_state	Count_of_customers
1	sao paulo	SP	15540
2	rio de janeiro	RJ	6882
3	belo horizonte	MG	2773
4	brasilia	DF	2131
5	curitiba	PR	1521
6	campinas	SP	1444
7	porto alegre	RS	1379
8	salvador	BA	1245
9	guarulhos	SP	1189
10	sao bernardo do campo	SP	938

Inference: Sao Paulo city from SP state has the highest number of orders.

In-depth Exploration:

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

```
select extract(year from o.order_purchase_timestamp) as year,
count(distinct o.order_id) as No_of_orders
 from `target_sql.orders` o
join `target_sql.customers` c on c.customer_id=o.customer_id
group by year
order by 1
```



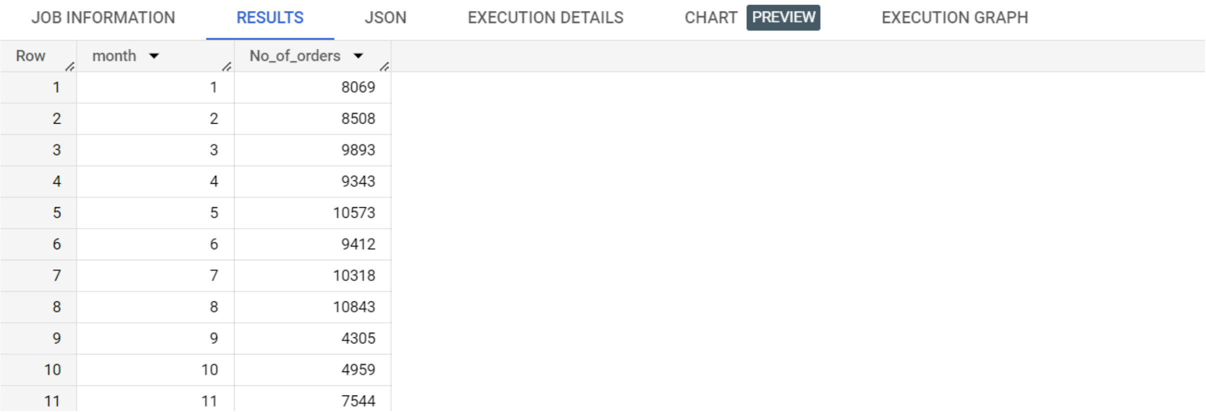
Query results

Row	year	No_of_orders
1	2016	329
2	2017	45101
3	2018	54011

Inference: There has been approximately 137 times increase in orders from 2016-2017.

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

```
select extract(month from o.order_purchase_timestamp) as month,
count(distinct o.order_id) as No_of_orders
from `target_sql.orders` o
join `target_sql.customers` c on c.customer_id=o.customer_id
group by month
order by 1
```



Row	month	No_of_orders
1	1	8069
2	2	8508
3	3	9893
4	4	9343
5	5	10573
6	6	9412
7	7	10318
8	8	10843
9	9	4305
10	10	4959
11	11	7544

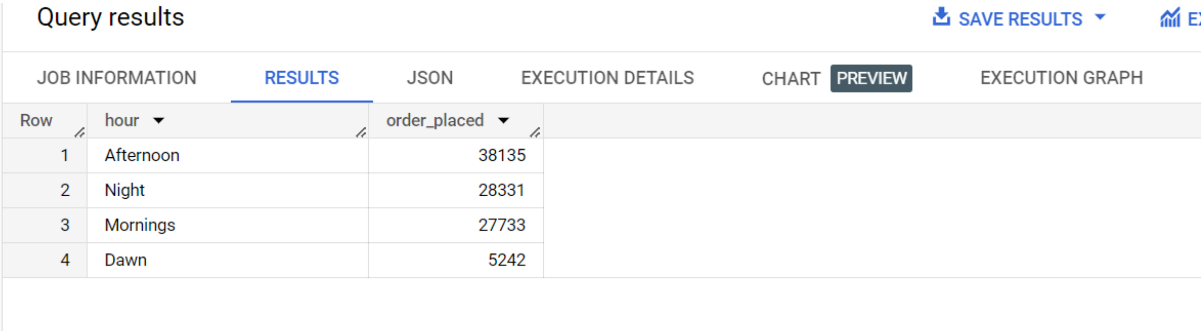
Inference: The count of orders increases from March to August with fluctuations in between.

In September we can see a massive dip in orders count.

3. **During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)**

```
 select

case when extract(hour from o.order_purchase_timestamp)  between 0 and 6 then 'Dawn'
     when extract(hour from o.order_purchase_timestamp)  between 7 and 12 then 'Mornings'
     when extract(hour from o.order_purchase_timestamp)  between 13 and 18 then 'Afternoon'
     when extract(hour from o.order_purchase_timestamp)  between 19 and 23 then 'Night'
end as hour, count(o.order_id) as order_placed
from `target_sql.orders` o
join `target_sql.customers` c on c.customer_id=o.customer_id
group by hour
order by 2 desc
```



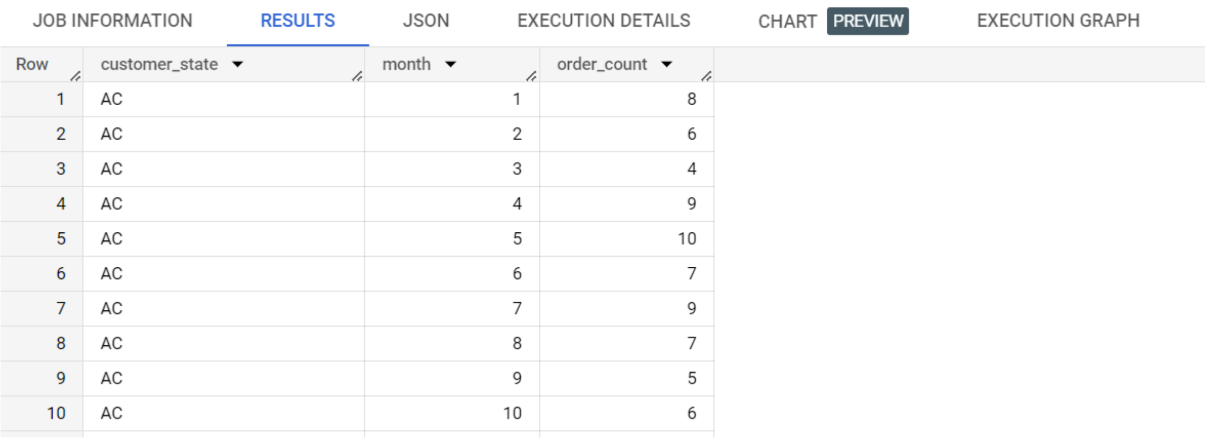
Row	hour	order_placed
1	Afternoon	38135
2	Night	28331
3	Mornings	27733
4	Dawn	5242

Inference: During the afternoon session most of the orders were placed.

Evolution of E-commerce orders in the Brazil region:

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(o.order_id) order_count
from `target_sql.orders` o
join `target_sql.customers` c on c.customer_id=o.customer_id
group by 1,2
order by 1,2
```

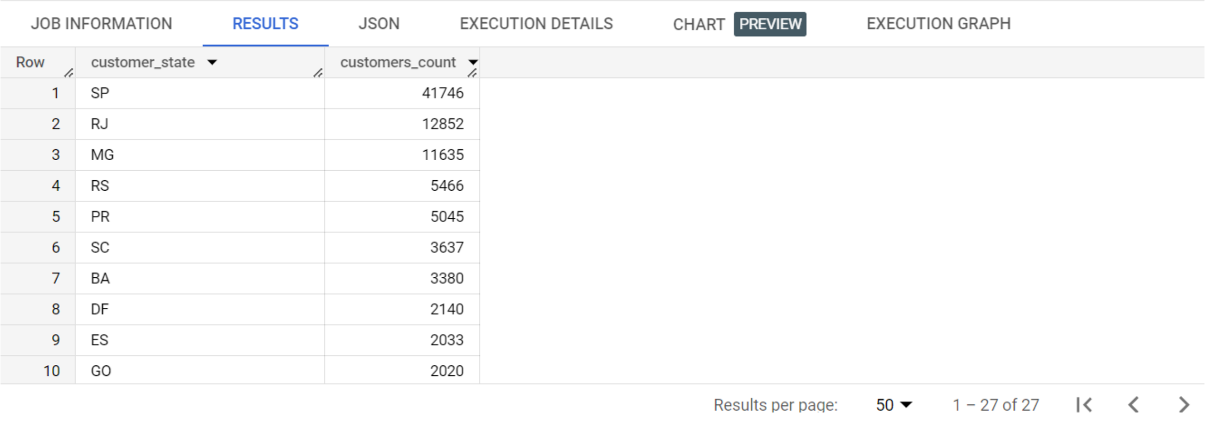


Row	customer_state	month	order_count
1	AC	1	8
2	AC	2	6
3	AC	3	4
4	AC	4	9
5	AC	5	10
6	AC	6	7
7	AC	7	9
8	AC	8	7
9	AC	9	5
10	AC	10	6

Inference: We can infer that SP state has highest number of orders than any other states.

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

```
Select c.customer_state,COUNT(c.customer_id) as customers_count
From `target_sql.customers` c
group by c.customer_state
order by customers_count desc
```



Row	customer_state	customers_count
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

Results per page: 50 1 – 27 of 27

Inference: We can infer that highest distribution of customers is in SP -Sao Paulo state.

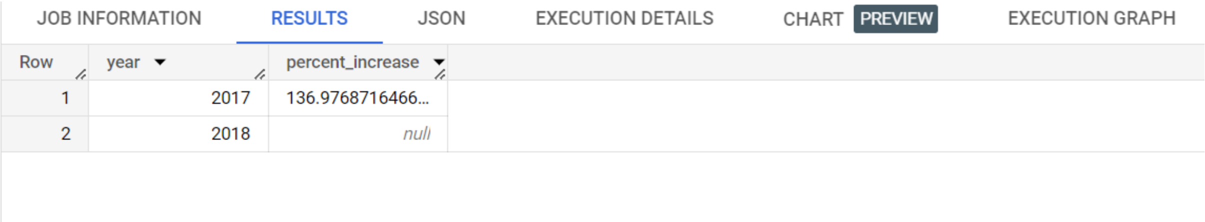
Impact on Economy: Analyse the money movement by e-commerce by looking at order prices, freight and others.

1. **Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only).**

```
with cte1 as
(
select extract(year from order_purchase_timestamp) year,sum(p.payment_value)
curr_month
from `target_sql.orders` o
join `target_sql.payments` p on p.order_id=o.order_id
where  extract(month from order_purchase_timestamp) between 1 and 8
group by 1
order by 1
),

cte2 as
(
select *, lead(curr_month,1) over (order by year) next_month
from cte1
)

select cte2.year,(next_month-curr_month)/curr_month*100 as percent_increase from
cte2
```

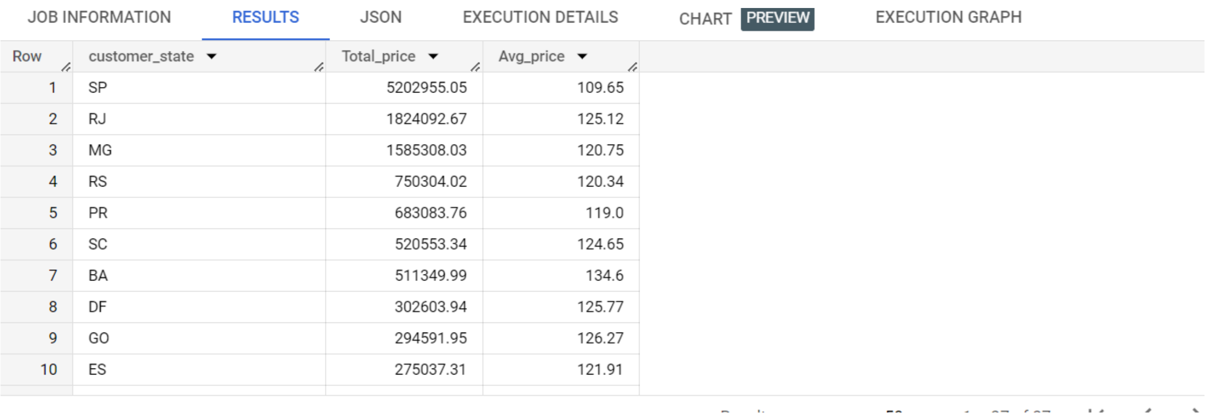


Row	year	percent_increase
1	2017	136.9768716466...
2	2018	null

Inference: We can infer that there was 137 percent increase in payment values from 2017 to 2018(including only months between January-August)

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

```
select c.customer_state,round(sum(oi.price),2) Total_price,round(avg(oi.price),2)
Avg_price
from `target_sql.customers` c
join `target_sql.orders` o on o.customer_id=c.customer_id
join `target_sql.order_items` oi on oi.order_id=o.order_id
group by c.customer_state
order by 2 desc,3 asc
```

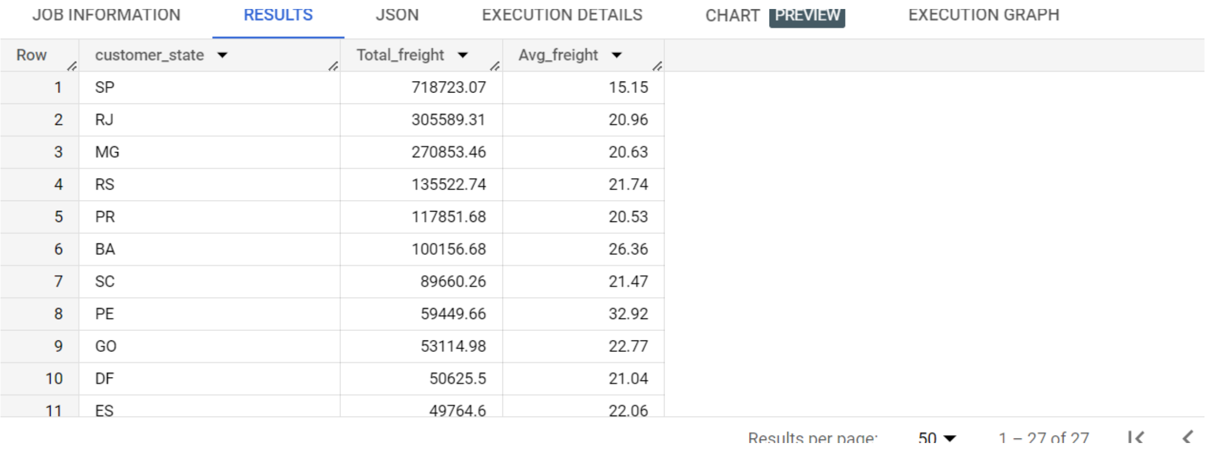


Row	customer_state	Total_price	Avg_price
1	SP	5202955.05	109.65
2	RJ	1824092.67	125.12
3	MG	1585308.03	120.75
4	RS	750304.02	120.34
5	PR	683083.76	119.0
6	SC	520553.34	124.65
7	BA	511349.99	134.6
8	DF	302603.94	125.77
9	GO	294591.95	126.27
10	ES	275037.31	121.91

Inference: SP state has highest total price value of all the states but also has the lowest average price value.

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

```
select c.customer_state,round(sum(oi.freight_value),2)
Total_freight,round(avg(oi.freight_value),2) Avg_freight
from `target_sql.customers` c
join `target_sql.orders` o on o.customer_id=c.customer_id
join `target_sql.order_items` oi on oi.order_id=o.order_id
group by c.customer_state
order by 2 desc,3 asc
```



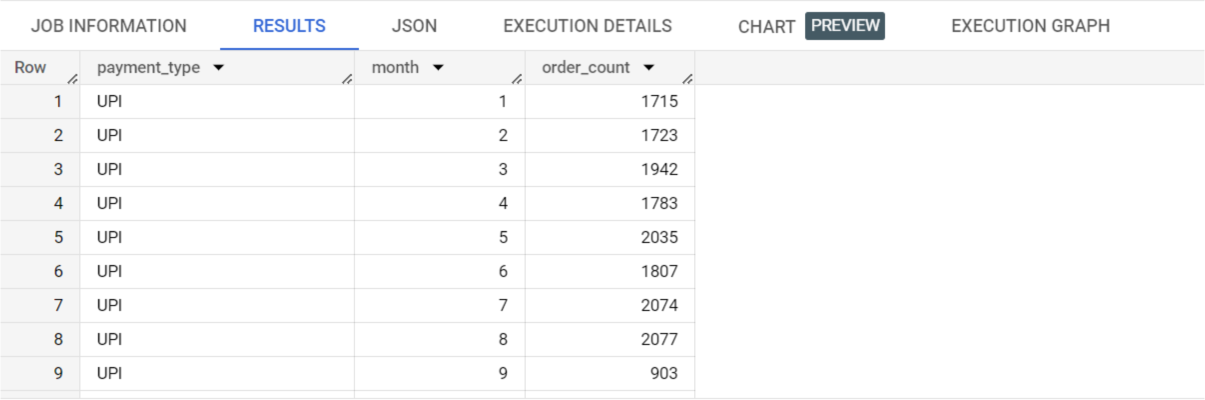
Row	customer_state	Total_freight	Avg_freight
1	SP	718723.07	15.15
2	RJ	305589.31	20.96
3	MG	270853.46	20.63
4	RS	135522.74	21.74
5	PR	117851.68	20.53
6	BA	100156.68	26.36
7	SC	89660.26	21.47
8	PE	59449.66	32.92
9	GO	53114.98	22.77
10	DF	50625.5	21.04
11	ES	49764.6	22.06

Inference: SP state has the highest total freight value and also lowest average freight value.

Analysis based on the payments:

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)
month,COUNT(distinct o.order_id)  order_count
FROM `target_sql.orders` o
JOIN `target_sql.payments` p on o.order_id = p.order_id
GROUP BY 1, 2
ORDER BY 1, 2
```

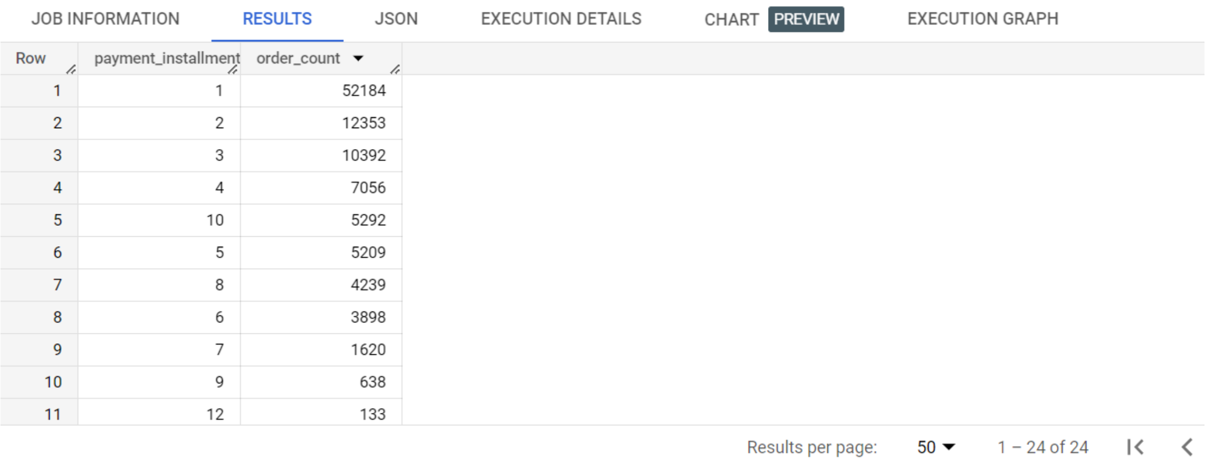


Row	payment_type	month	order_count
1	UPI	1	1715
2	UPI	2	1723
3	UPI	3	1942
4	UPI	4	1783
5	UPI	5	2035
6	UPI	6	1807
7	UPI	7	2074
8	UPI	8	2077
9	UPI	9	903

Inference: We can infer from the data that Credit card payment mode is highly used and popular than others.

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

```
SELECT p.payment_installments,COUNT(o.order_id) order_count
FROM `target_sql.orders` o
join `target_sql.payments` p on o.order_id = p.order_id
where o.order_status != 'canceled'
GROUP BY 1
ORDER BY 2 desc
```



Row	payment_installment	order_count
1	1	52184
2	2	12353
3	3	10392
4	4	7056
5	10	5292
6	5	5209
7	8	4239
8	6	3898
9	7	1620
10	9	638
11	12	133

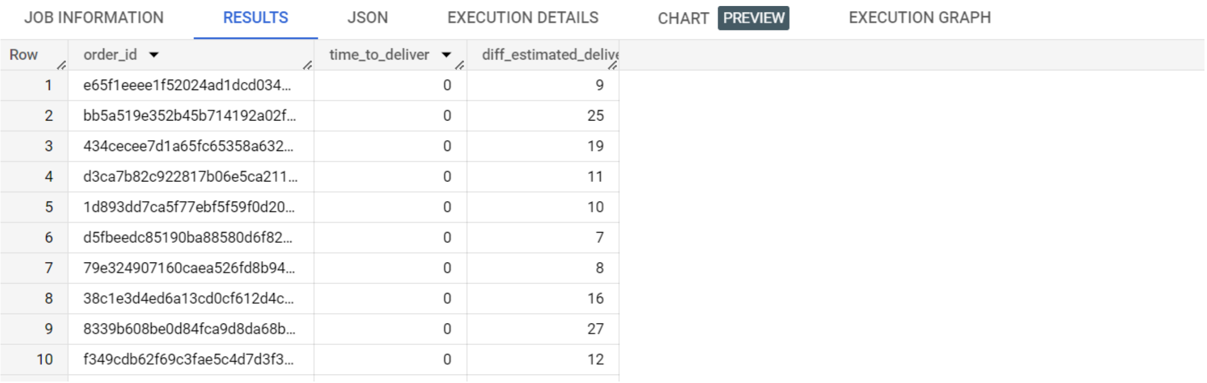
Inference: It can be inferred that most of the orders (52184) like to pay in 1 instalment and a smaller number of orders i.e. 18 pay in 24 instalments

Analysis based on sales, freight and delivery time

1. **Find the no. of days taken to deliver each order from the order's purchase date as delivery time.**
   Also, calculate the difference (in days) between the estimated & actual delivery date of an order.
   Do this in a single query.

   You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:
   - ○ **time_to_deliver = order_delivered_customer_date - order_purchase_timestamp**
   - ○ **diff_estimated_delivery = order_estimated_delivery_date - 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) as
diff_estimated_delivery
from `target_sql.orders`
where date_diff(order_delivered_customer_date, order_purchase_timestamp,day) is not null
order by time_to_deliver
```



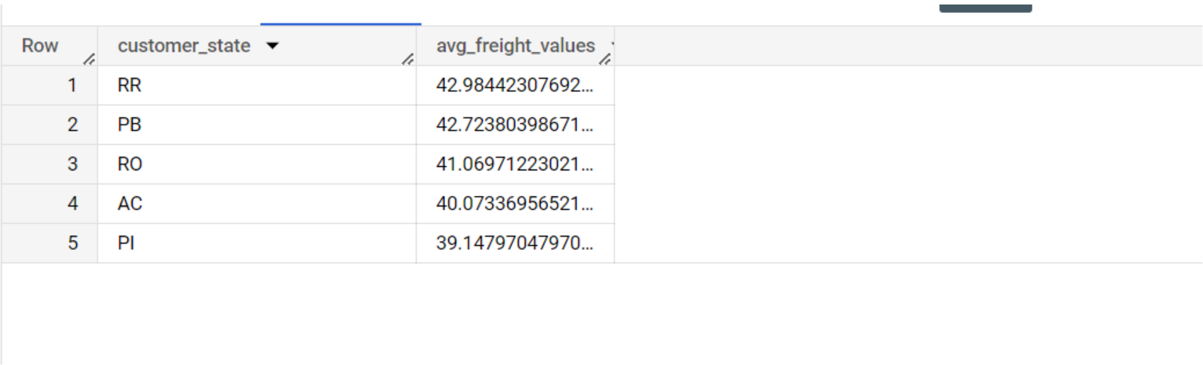
Row	order_id	time_to_deliver	diff_estimated_delive
1	e65f1eeee1f52024ad1dcd034...	0	9
2	bb5a519e352b45b714192a02f...	0	25
3	434cecee7d1a65fc65358a632...	0	19
4	d3ca7b82c922817b06e5ca211...	0	11
5	1d893dd7ca5f77ebf5f59f0d20...	0	10
6	d5fbeedc85190ba88580d6f82...	0	7
7	79e324907160caea526fd8b94...	0	8
8	38c1e3d4ed6a13cd0cf612d4c...	0	16
9	8339b608be0d84fca9d8da68b...	0	27
10	f349cdb62f69c3fae5c4d7d3f3...	0	12

Inference: We can infer that time taken to deliver was less than 1 day for 10 orders .

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

--Top 5 states with highest avg freight values

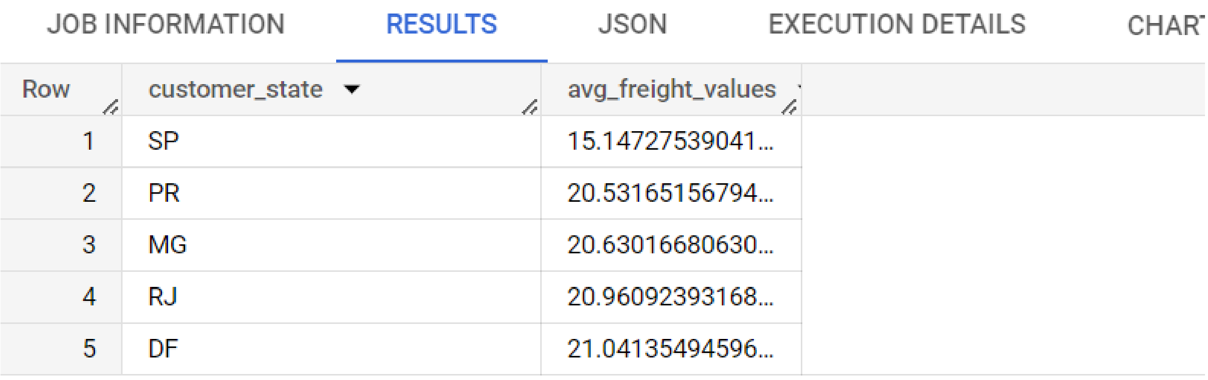
```
select c.customer_state,avg(oi.freight_value) as avg_freight_values from
`target_sql.orders` o
join `target_sql.order_items` oi on oi.order_id=o.order_id
join `target_sql.customers` c on c.customer_id=o.customer_id
group by c.customer_state
order by avg_freight_values desc
limit 5
```



Row	customer_state	avg_freight_values
1	RR	42.98442307692...
2	PB	42.72380398671...
3	RO	41.06971223021...
4	AC	40.07336956521...
5	PI	39.14797047970...

--Bottom 5 states with lowest avg freight values

```
select c.customer_state,avg(oi.freight_value) as avg_freight_values from
`target_sql.orders` o
join `target_sql.order_items` oi on oi.order_id=o.order_id
join `target_sql.customers` c on c.customer_id=o.customer_id
group by c.customer_state
order by avg_freight_values
limit 5
```



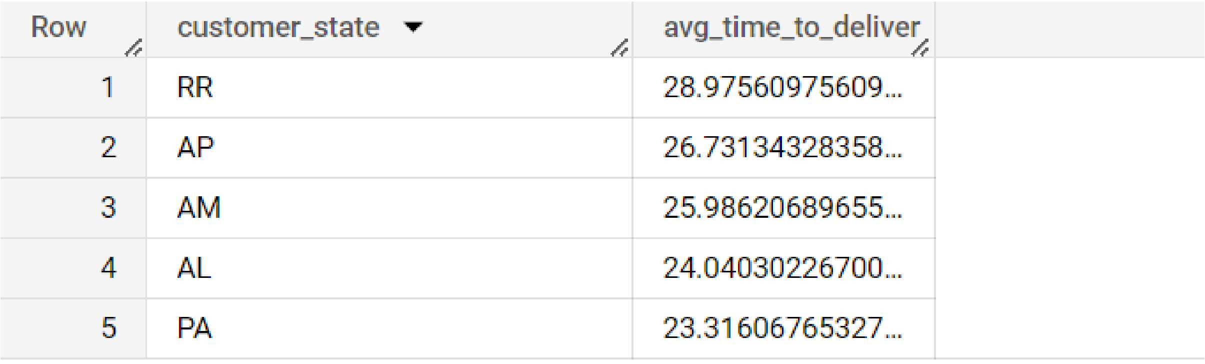
Row	customer_state	avg_freight_values
1	SP	15.14727539041...
2	PR	20.53165156794...
3	MG	20.63016680630...
4	RJ	20.96092393168...
5	DF	21.04135494596...

Inference: We can infer from above data as the top most average freight value is of state- RR and the lowest avg freight value is of state -SP

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

--top 5 states ordered by delivery time

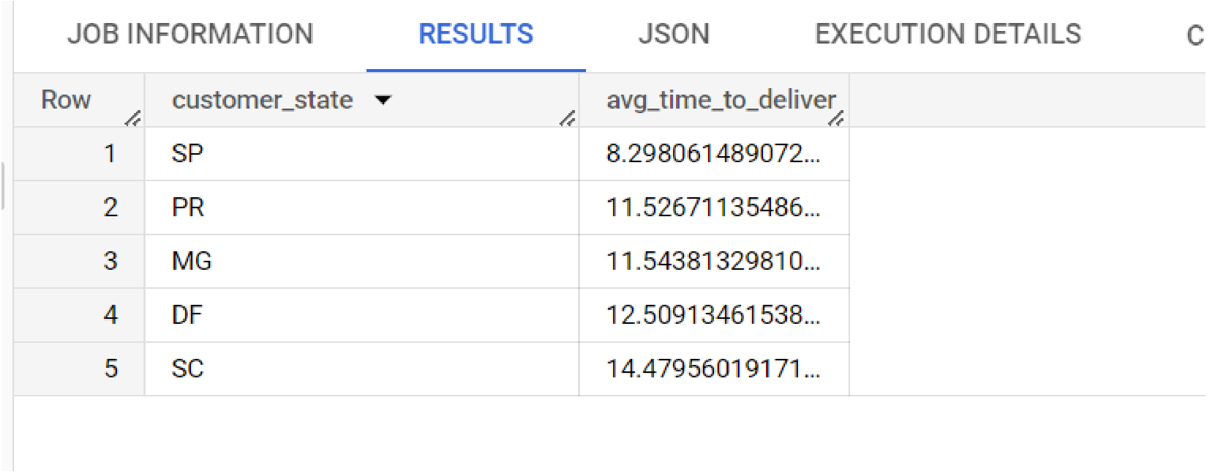
```
select c.customer_state,
avg(date_diff(order_delivered_customer_date, order_purchase_timestamp,day)) as
avg_time_to_deliver
from `target_sql.orders` o
join `target_sql.customers` c on o.customer_id = c.customer_id
group by c.customer_state
order by avg_time_to_deliver desc
limit 5
```



Row	customer_state	avg_time_to_deliver
1	RR	28.97560975609...
2	AP	26.73134328358...
3	AM	25.98620689655...
4	AL	24.04030226700...
5	PA	23.31606765327...

----bottom 5 states ordered by delivery time

```
select c.customer_state,
avg(date_diff(order_delivered_customer_date, order_purchase_timestamp,day)) as
avg_time_to_deliver
from `target_sql.orders` o
join `target_sql.customers` c on o.customer_id = c.customer_id
group by c.customer_state
order by avg_time_to_deliver
limit 5
```

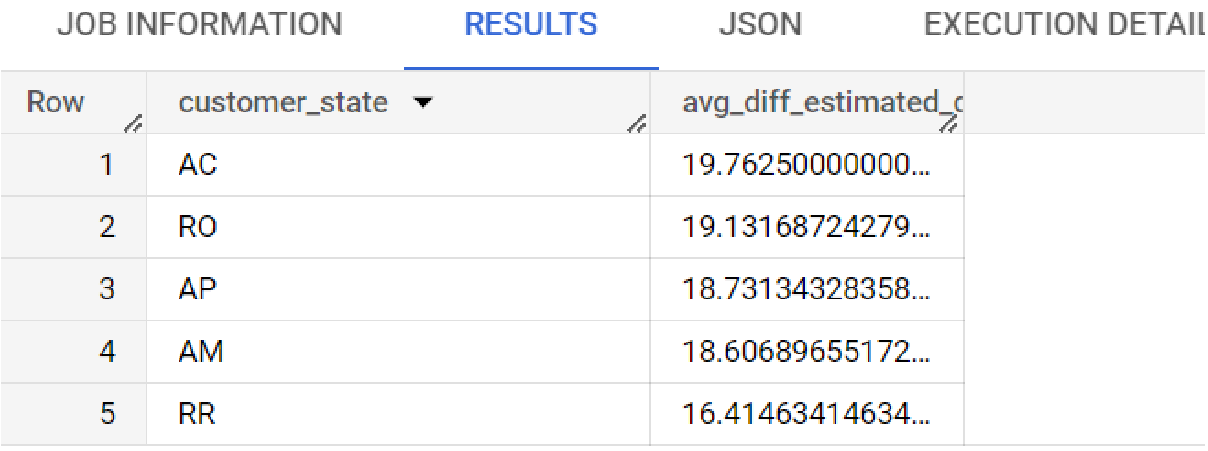


Row	customer_state	avg_time_to_deliver
1	SP	8.298061489072...
2	PR	11.52671135486...
3	MG	11.54381329810...
4	DF	12.50913461538...
5	SC	14.47956019171...

Inference: We can infer that SP state has lowest delivery time and RR state takes lot of time to deliver.

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

```
select c.customer_state,
avg(date_diff(order_estimated_delivery_date, order_delivered_customer_date, day)) as
avg_diff_estimated_delivery
from `target_sql.orders` o
join `target_sql.customers` c on o.customer_id = c.customer_id
group by c.customer_state
order by avg_diff_estimated_delivery desc
limit 5
```



Row	customer_state	avg_diff_estimated_c
1	AC	19.76250000000...
2	RO	19.13168724279...
3	AP	18.73134328358...
4	AM	18.60689655172...
5	RR	16.41463414634...

Inference: We can infer fastest delivery as compared to estimated delivery is in AC state.