1. **Time period for which the data is given**

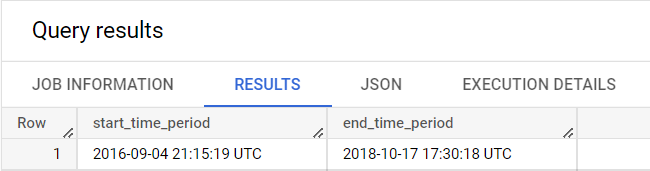
**Query:**

SELECT

MIN(order\_purchase\_timestamp) AS start\_time\_period,

MAX(order\_purchase\_timestamp) AS end\_time\_period

FROM `target-scaler-360317.customers.orders`



1. **Cities and States covered in the dataset**

**Query:**

SELECT DISTINCT geolocation\_city,geolocation\_state

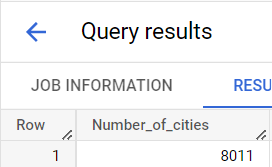
FROM  geolocation



**Query:**

SELECT COUNT(DISTINCT geolocation\_city) AS Number\_of\_cities

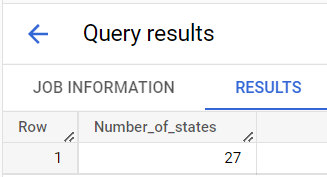
FROM geolocation;



**Query:**

SELECT COUNT(DISTINCT geolocation\_state) AS Number\_of\_states

FROM .geolocation



**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?**

**Month- Wise**

**Query:**

WITH month\_purchase AS

(

SELECT EXTRACT(MONTH FROM order\_purchase\_timestamp) AS Month\_wise

FROM orders

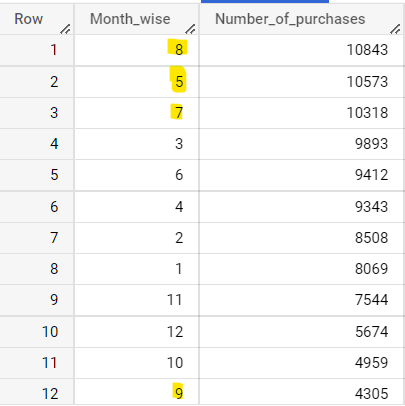
)

SELECT Month\_wise,COUNT(Month\_wise) AS Number\_of\_purchases

from month\_purchase

GROUP BY Month\_wise

ORDER BY COUNT(Month\_wise) DESC;



* **As we see in the Screenshot 8,5,7 month have more number of purchases compares to other months and 9th month have lowest purchases.**
* **if we observer it seems like there is pattern according to different quarters in the year.**

**Quarter – wise**

**Query:**

WITH month\_purchase AS

(

SELECT EXTRACT(MONTH FROM order\_purchase\_timestamp) AS Month\_wise

FROM orders

),

quarter\_purchse AS

(

SELECT CASE

WHEN Month\_wise IN (1,2,3,4) THEN "1st Quarter"

WHEN Month\_wise IN (5,6,7,8) THEN "2nd Quarter"

ELSE "3RD Quarter"

END AS Quarters

FROM month\_purchase

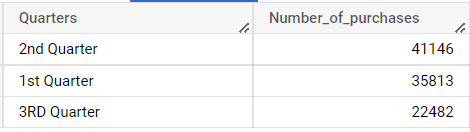
)

SELECT Quarters,COUNT(Quarters) AS Number\_of\_purchases

from quarter\_purchse

GROUP BY Quarters

ORDER BY COUNT(Quarters) DESC;



**3rd Quarter purchases are very low compares to other quarters ,2nd quarter have high purchases counts.**

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

**Query:**

WITH time\_purchase AS

(

SELECT EXTRACT(TIME FROM order\_purchase\_timestamp) AS time\_wise

FROM orders

),

category AS (

SELECT

CASE

WHEN time\_wise BETWEEN "02:01:00" AND "06:00:00"

THEN "DAWN"

WHEN time\_wise BETWEEN "06:01:00" AND "12:00:00"

THEN "MORNING"

WHEN time\_wise BETWEEN "12:01:00" AND "15:00:00"

THEN "AFTERNOON"

WHEN time\_wise BETWEEN "15:01:00" AND "19:00:00"

THEN "EVENING"

ELSE "NIGHT"

END AS Time\_category

FROM time\_purchase

)

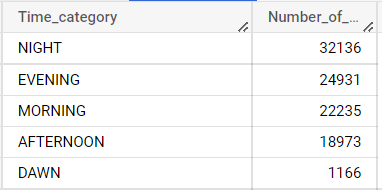
SELECT Time\_category,

COUNT(Time\_category) AS Number\_of\_purchases

FROM category

GROUP BY Time\_category

ORDER BY COUNT(Time\_category) DESC;



**It clearly makes sense as most of the people will be free at evening and night times that’s why at that time we see increase in number of purchases.**

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

1. **Get month on month orders by states**

**Query:**

WITH monthadd AS(

  SELECT \*,EXTRACT(MONTH FROM order\_purchase\_timestamp) AS month

  FROM  orders

)

SELECT c.customer\_state,o.month,COUNT(o.order\_id) AS Number\_of\_purchases

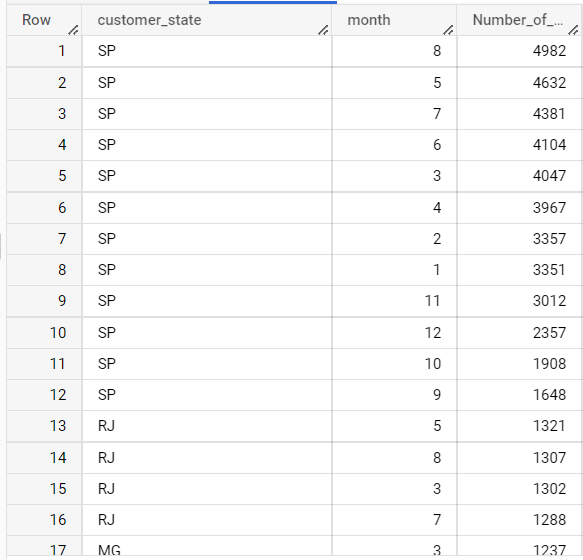
FROM monthadd AS o

JOIN customers AS c

ON o.customer\_id = c.customer\_id

GROUP BY c.customer\_state,o.month

ORDER BY COUNT(o.order\_id) DESC ,c.customer\_state;



**SP – state have clearly very high number of purchases in all months**

**City Wise – Highest and Lowest number of purchases , 5 Cities in each case.**

**Query:**

**TOP 5:**

SELECT c.customer\_city,COUNT(o.order\_id) AS Number\_of\_purchases

FROM orders AS o

JOIN  customers AS c

ON o.customer\_id = c.customer\_id

GROUP BY c.customer\_city

ORDER BY COUNT(o.order\_id) DESC

LIMIT 5;



**Least 5**

SELECT c.customer\_city,COUNT(o.order\_id) AS Number\_of\_purchases

FROM orders AS o

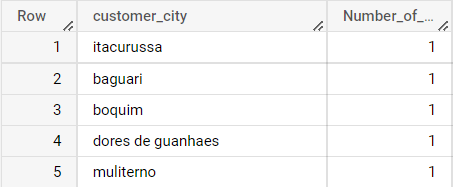
JOIN customers AS c

ON o.customer\_id = c.customer\_id

GROUP BY c.customer\_city

ORDER BY COUNT(o.order\_id)

LIMIT 5;

****

**Sao Paulo have the highest number of purchases by far – This also makes sense as Sao Paulo is populous city in Brazil**

1. How are customers distributed in Brazil

**Query:**

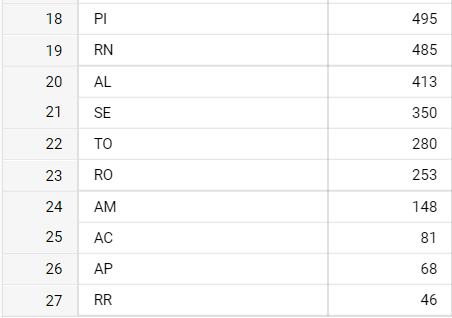
SELECT customer\_state, COUNT(customer\_unique\_id)

FROM customers

GROUP BY customer\_state

ORDER BY COUNT(customer\_unique\_id) DESC;





**As we saw above SP had more purchases it links well here as more number of customers, more number of purchases.**

**Impact on Economy: Analyze the money movemented 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)**

**QUERY:**

WITH monthyearadd AS(

  SELECT \*,EXTRACT(MONTH FROM order\_purchase\_timestamp) AS month,

  EXTRACT(YEAR FROM order\_purchase\_timestamp) AS year

  FROM orders

),

year\_wise AS(

SELECT m.year,SUM(p.payment\_value) AS total\_payment

FROM monthyearadd AS m

JOIN payments AS p

ON m.order\_id = p.order\_id

WHERE m.year  in  (2017,2018) AND m.month BETWEEN 1 AND 8

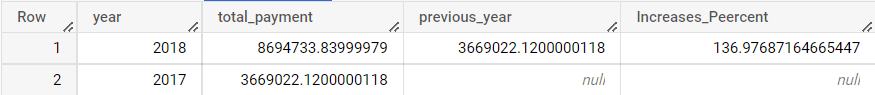
GROUP BY m.year

)

SELECT \*,LAG(total\_payment) OVER(order by year) AS previous\_year,

(total\_payment-(LAG(total\_payment) OVER(order by year)))/(LAG(total\_payment) OVER(order by year))\*100 AS Increased\_Percentage

FROM year\_wise;



**We have 136% Increase in payments from 2017 to 2018 (include months between Jan to Aug only)**

1. **Mean & Sum of price and freight value by customer state**

**QUERY:**

SELECT c.customer\_state,

SUM(ot.price) AS Total\_Price ,

AVG(ot.price) AS Mean\_Price ,

SUM(ot.freight\_value) AS Total\_Freight\_Value ,

AVG(ot.freight\_value) AS Mean\_Freight\_Value

FROM order\_items AS ot

JOIN orders AS o

ON ot.order\_id = o.order\_id

JOIN customers AS c

ON o.customer\_id = c.customer\_id

GROUP BY c.customer\_state



**Analysis on sales, freight and delivery time**

1. **Calculate days between purchasing, delivering and estimated delivery**

**Query:**

WITH diff AS(

SELECT  \*,order\_delivered\_customer\_date-order\_purchase\_timestamp AS time\_to\_delivery,

order\_estimated\_delivery\_date-order\_delivered\_customer\_date AS diff\_estimated\_delivery

FROM orders

WHERE (order\_delivered\_customer\_date IS NOT NULL AND order\_purchase\_timestamp IS NOT NULL) AND order\_estimated\_delivery\_date IS NOT NULL

)

SELECT c.customer\_state,AVG(ot.freight\_value) AS AVG\_freight\_value,

AVG(o.time\_to\_delivery) AS AVG\_time\_to\_delivery, AVG(o.diff\_estimated\_delivery) AS AVG\_diff\_estimated\_delivery

FROM customers AS c

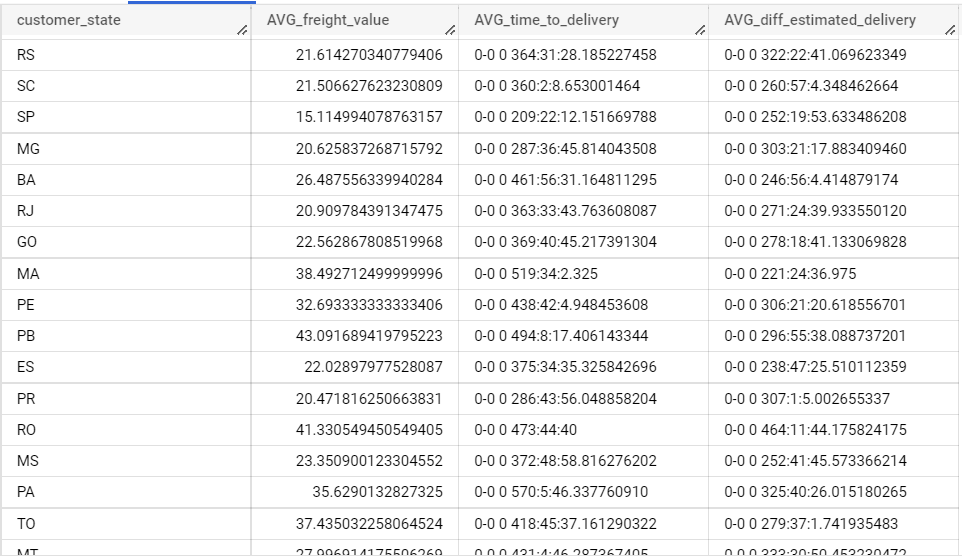
JOIN diff AS o

ON c.customer\_id = o.customer\_id

JOIN order\_items AS ot

ON o.order\_id = ot.order\_id

GROUP BY c.customer\_state



**Top 5 states with highest/lowest average freight value:**

**Highest:**

WITH diff AS(

SELECT  \*,order\_delivered\_customer\_date-order\_purchase\_timestamp AS time\_to\_delivery,

order\_estimated\_delivery\_date-order\_delivered\_customer\_date AS diff\_estimated\_delivery

FROM orders

WHERE (order\_delivered\_customer\_date IS NOT NULL AND order\_purchase\_timestamp IS NOT NULL) AND order\_estimated\_delivery\_date IS NOT NULL

)

SELECT c.customer\_state,AVG(ot.freight\_value) AS AVG\_freight\_value,

AVG(o.time\_to\_delivery) AS AVG\_time\_to\_delivery, AVG(o.diff\_estimated\_delivery) AS AVG\_diff\_estimated\_delivery

FROM customers AS c

JOIN diff AS o

ON c.customer\_id = o.customer\_id

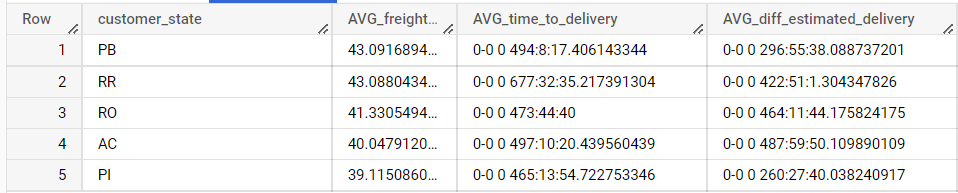
JOIN order\_items AS ot

ON o.order\_id = ot.order\_id

GROUP BY c.customer\_state

ORDER BY AVG(ot.freight\_value) DESC

LIMIT 5



**Lowest:**

WITH diff AS(

SELECT  \*,order\_delivered\_customer\_date-order\_purchase\_timestamp AS time\_to\_delivery,

order\_estimated\_delivery\_date-order\_delivered\_customer\_date AS diff\_estimated\_delivery

FROM orders

WHERE (order\_delivered\_customer\_date IS NOT NULL AND order\_purchase\_timestamp IS NOT NULL) AND order\_estimated\_delivery\_date IS NOT NULL

)

SELECT c.customer\_state,AVG(ot.freight\_value) AS AVG\_freight\_value,

AVG(o.time\_to\_delivery) AS AVG\_time\_to\_delivery, AVG(o.diff\_estimated\_delivery) AS AVG\_diff\_estimated\_delivery

FROM customers AS c

JOIN diff AS o

ON c.customer\_id = o.customer\_id

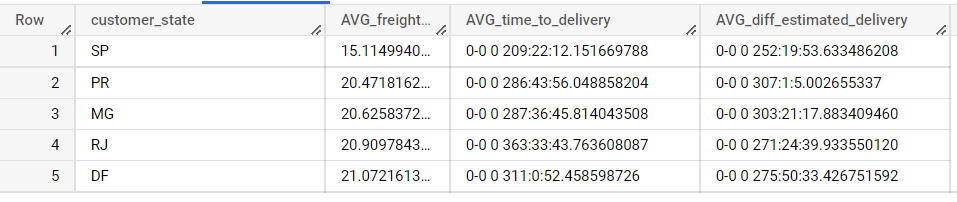
JOIN order\_items AS ot

ON o.order\_id = ot.order\_id

GROUP BY c.customer\_state

ORDER BY AVG(ot.freight\_value)

LIMIT 5



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

**Highest:**

WITH diff AS(

SELECT  \*,order\_delivered\_customer\_date-order\_purchase\_timestamp AS time\_to\_delivery,

order\_estimated\_delivery\_date-order\_delivered\_customer\_date AS diff\_estimated\_delivery

FROM orders

WHERE (order\_delivered\_customer\_date IS NOT NULL AND order\_purchase\_timestamp IS NOT NULL) AND order\_estimated\_delivery\_date IS NOT NULL

)

SELECT c.customer\_state,

AVG(o.time\_to\_delivery) AS AVG\_time\_to\_delivery, AVG(o.diff\_estimated\_delivery) AS AVG\_diff\_estimated\_delivery

FROM customers AS c

JOIN diff AS o

ON c.customer\_id = o.customer\_id

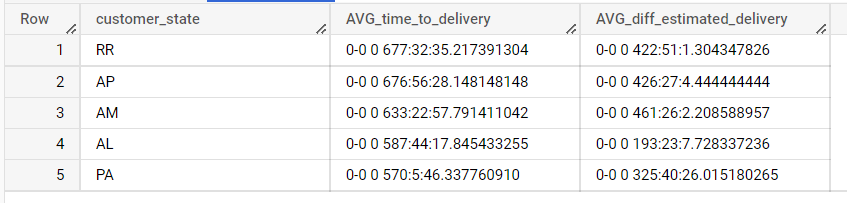
JOIN order\_items AS ot

ON o.order\_id = ot.order\_id

GROUP BY c.customer\_state

ORDER BY  AVG(o.time\_to\_delivery) DESC

LIMIT 5

****

**Lowest:**

WITH diff AS(

SELECT  \*,order\_delivered\_customer\_date-order\_purchase\_timestamp AS time\_to\_delivery,

order\_estimated\_delivery\_date-order\_delivered\_customer\_date AS diff\_estimated\_delivery

FROM orders

WHERE (order\_delivered\_customer\_date IS NOT NULL AND order\_purchase\_timestamp IS NOT NULL) AND order\_estimated\_delivery\_date IS NOT NULL

)

SELECT c.customer\_state,

AVG(o.time\_to\_delivery) AS AVG\_time\_to\_delivery, AVG(o.diff\_estimated\_delivery) AS AVG\_diff\_estimated\_delivery

FROM customers AS c

JOIN diff AS o

ON c.customer\_id = o.customer\_id

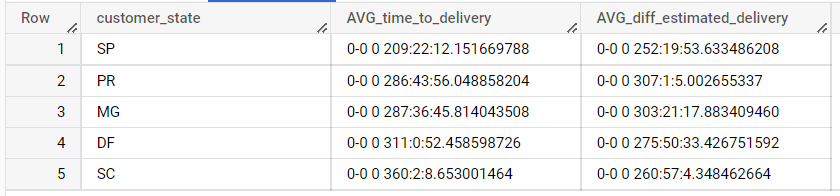
JOIN order\_items AS ot

ON o.order\_id = ot.order\_id

GROUP BY c.customer\_state

ORDER BY  AVG(o.time\_to\_delivery)

LIMIT 5

****

**Top 5 states where delivery is really fast/ not so fast compared to estimated date**

**Really Fast:**

WITH diff AS(

SELECT  \*,order\_delivered\_customer\_date-order\_purchase\_timestamp AS time\_to\_delivery,

order\_estimated\_delivery\_date-order\_delivered\_customer\_date AS diff\_estimated\_delivery

FROM orders

WHERE (order\_delivered\_customer\_date IS NOT NULL AND order\_purchase\_timestamp IS NOT NULL) AND order\_estimated\_delivery\_date IS NOT NULL

)

SELECT c.customer\_state,

AVG(o.diff\_estimated\_delivery) AS AVG\_diff\_estimated\_delivery

FROM customers AS c

JOIN diff AS o

ON c.customer\_id = o.customer\_id

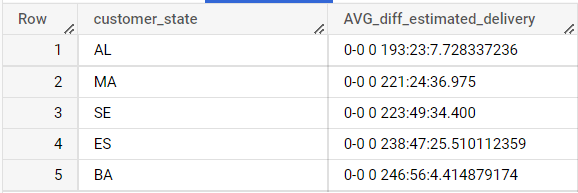
JOIN order\_items AS ot

ON o.order\_id = ot.order\_id

GROUP BY c.customer\_state

ORDER BY  AVG(o.diff\_estimated\_delivery)

LIMIT 5

****

**Not So Fast:**

WITH diff AS(

SELECT  \*,order\_delivered\_customer\_date-order\_purchase\_timestamp AS time\_to\_delivery,

order\_estimated\_delivery\_date-order\_delivered\_customer\_date AS diff\_estimated\_delivery

FROM orders

WHERE (order\_delivered\_customer\_date IS NOT NULL AND order\_purchase\_timestamp IS NOT NULL) AND order\_estimated\_delivery\_date IS NOT NULL

)

SELECT c.customer\_state,

AVG(o.diff\_estimated\_delivery) AS AVG\_diff\_estimated\_delivery

FROM customers AS c

JOIN diff AS o

ON c.customer\_id = o.customer\_id

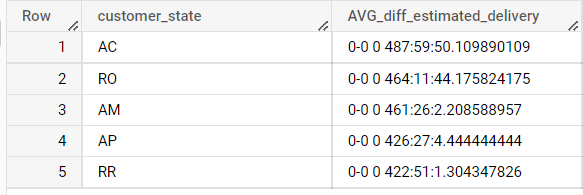
JOIN order\_items AS ot

ON o.order\_id = ot.order\_id

GROUP BY c.customer\_state

ORDER BY  AVG(o.diff\_estimated\_delivery) DESC

LIMIT 5

****

**Payment type analysis:**

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

**Over Payment types**

**QUERY:**

WITH monthadd AS(

  SELECT \*,EXTRACT(MONTH FROM order\_purchase\_timestamp) AS month

  FROM orders

)

SELECT p.payment\_type,COUNT(m.order\_id) AS Number\_of\_orders

FROM monthadd AS m

JOIN payments AS p

ON m.order\_id = p.order\_id

GROUP BY p.payment\_type

ORDER BY p.payment\_type,COUNT(m.order\_id);



**Over payments and months:**

WITH monthadd AS(

  SELECT \*,EXTRACT(MONTH FROM order\_purchase\_timestamp) AS month

  FROM orders

)

SELECT p.payment\_type,m.month,COUNT(m.order\_id) AS Number\_of\_orders

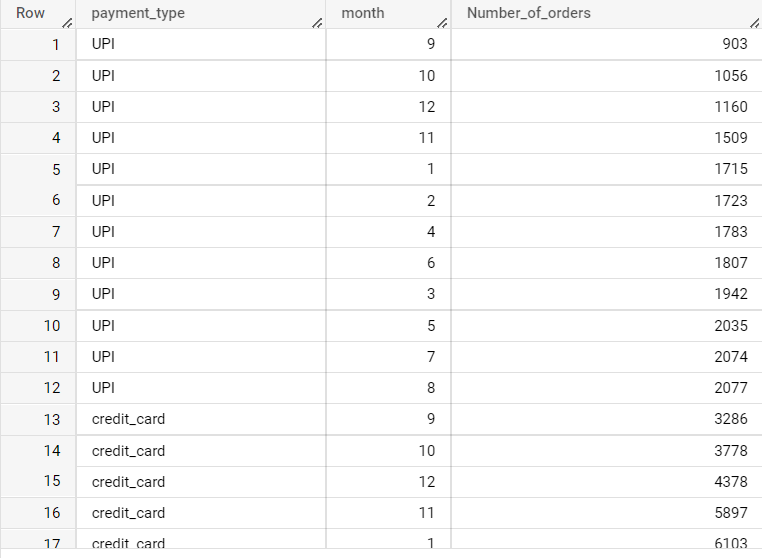
FROM monthadd AS m

JOIN payments AS p

ON m.order\_id = p.order\_id

GROUP BY p.payment\_type,m.month

ORDER BY p.payment\_type,COUNT(m.order\_id);



1. **Distribution of payment installments and count of orders**

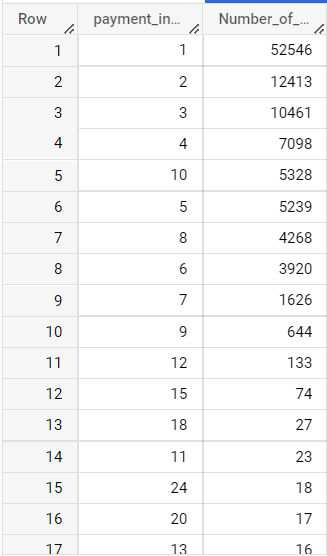
**Query:**

SELECT payment\_installments,COUNT(order\_id) AS Number\_of\_orders

FROM payments

GROUP BY payment\_installments

ORDER BY COUNT(order\_id) DESC



**Distribution – Most of our payment installments are between 1 to 10.**

**Actionable Insights and Recommendations:**

* Comparing 2017- 2018 growth rate is above 100 percent in sales which implies we need to increase the physical workforce and branches in the cities where most sales happened.
* AS 2nd quarter have more sales basically in a year hiring the intern or temporary workers for that month will be good.
* Sao Paulo have the highest number of purchases by far -- Increase the workforce in the stores open new branch in the city.
* SP state have lowest freight value (lowest cost to transport goods) and also have most number of sales which means here we can derive maximum profit should set up more resource towards this state.
* AL,MA,SE are the states where the delivery are done fast (here fast means how much less difference in delivered time and estimated time) , Call a meeting with this state managers and analyse to find out how they deliver fast and try to replicate in other states as well.
* Most of the payments are done using Credit cards, if able to get credit data find out which bank card is used most and try to have a partnership with the bank might help in increase revenue.
* Most used payment installments by users are 1,2,3,4,10. Increasing these installments to be valid for many products as possible will increase the interest in customers.
* Evening and Night times we see increase in purchases, Alerting the employees in the store to have better focus and have best employee at sight in this period of time.