BUSINESS CASE STUDY
SQL

Made by **Suryansh**
LinkedIn
E-mail - **suryansh_ce21b18_67@dtu.ac.in**

Context:

Target is one of the world's most recognized brands and one of America's leading retailers. Target makes itself a preferred shopping destination by offering outstanding value, inspiration, innovation and an exceptional guest experience that no other retailer can deliver.
This business case has information of 100k orders from 2016 to 2018 made at Target in Brazil. Its features allows viewing an order from multiple dimensions: from order status, price, payment and freight performance to customer location, product attributes and finally reviews written by customers.

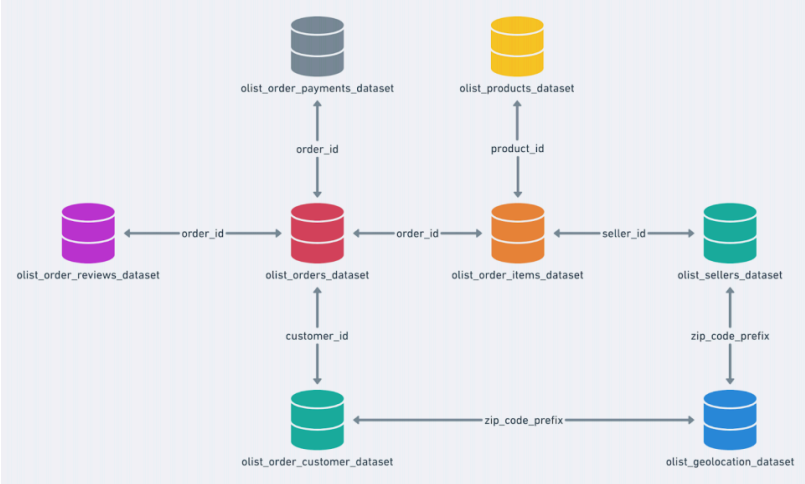
Dataset:

https://drive.google.com/drive/folders/1TGEc66YKbD443nslRi1bWgVd238gJCnb

Data is available in **8 csv files**:

1. Customers.csv
2. geolocation.csv
3. order_items.csv
4. payments.csv
5. reviews.csv
6. orders.csv
7. products.csv
8. sellers.csv

Each feature or columns of different CSV files are described below:

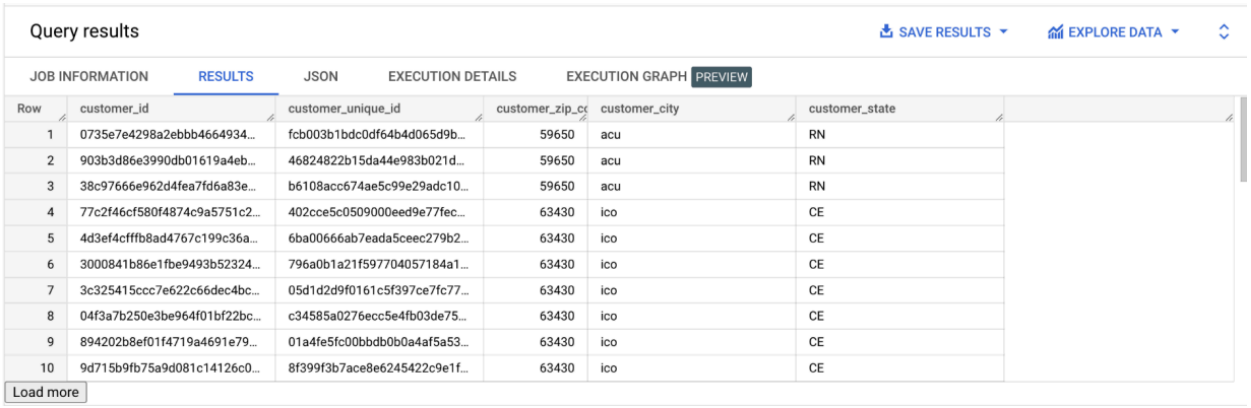


Q1) Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset :

- 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

Customers Table:

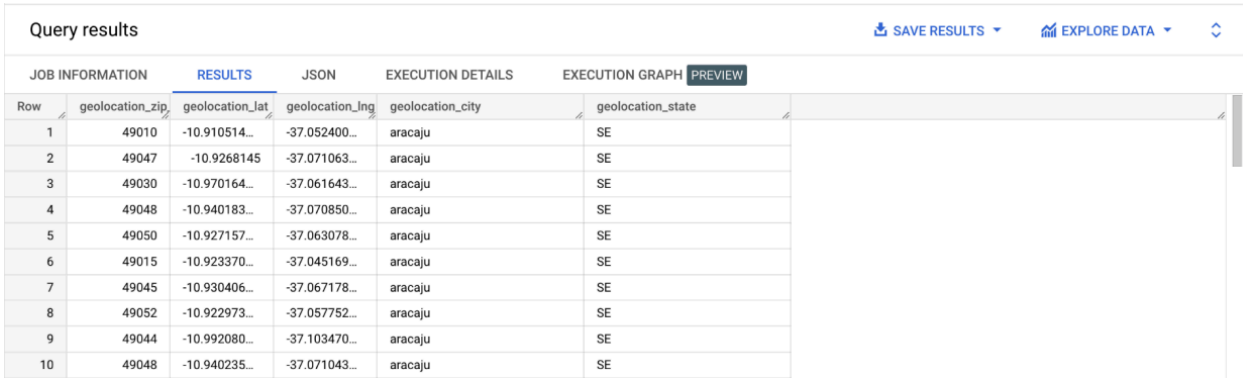
SELECT * FROM `business-case-study-sql.Target_dataset.customers` LIMIT 1000;



Row	customer_id	customer_unique_id	customer_zip_c	customer_city	customer_state
1	0735e7e4298a2ebbb4664934...	fcb003b1bdc0df64b4d065d9b...	59650	acu	RN
2	903b3d86e3990db01619a4eb...	46824822b15da44e983b021d...	59650	acu	RN
3	38c97666e962d4fea7fd6a83e...	b6108acc674ae5c99e29adc10...	59650	acu	RN
4	77c2f46cf580f4874c9a5751c2...	402cce5c0509000eed9e77fec...	63430	ico	CE
5	4d3ef4cfffb8ad4767c199c36a...	6ba00666ab7eada5ceec279b2...	63430	ico	CE
6	3000841b86e1fbe9493b52324...	796a0b1a21f597704057184a1...	63430	ico	CE
7	3c325415ccc7e622c66dec4bc...	05d1d2d9f0161c5f397ce7fc77...	63430	ico	CE
8	04f3a7b250e3be964f01bf22bc...	c34585a0276ecc5e4fb03de75...	63430	ico	CE
9	894202b8ef01f4719a4691e79...	01a4fe5fc00bbdb0b0a4af5a53...	63430	ico	CE
10	9d715b9fb75a9d081c14126c0...	8f399f3b7ace8e6245422c9e1f...	63430	ico	CE

GeoLocation Table:

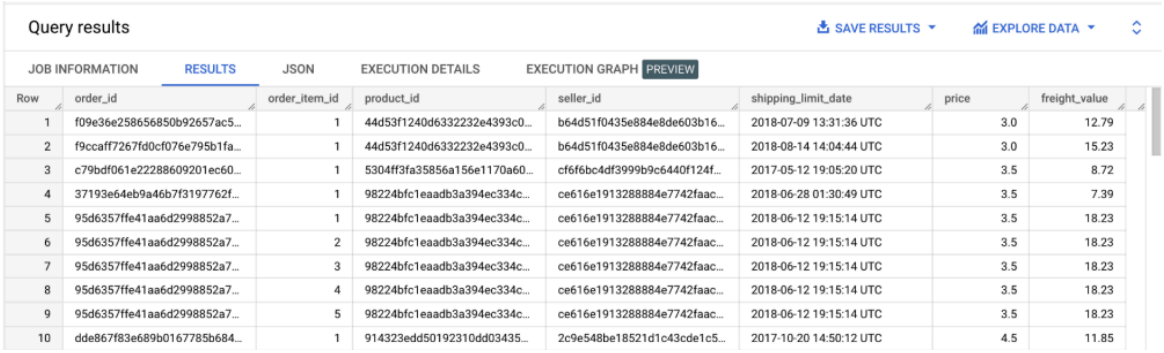
SELECT * FROM `business-case-study-sql.Target_dataset.geolocation` LIMIT 1000;



Row	geolocation_zip	geolocation_lat	geolocation_lng	geolocation_city	geolocation_state
1	49010	-10.910514...	-37.052400...	aracaju	SE
2	49047	-10.9268145	-37.071063...	aracaju	SE
3	49030	-10.970164...	-37.061643...	aracaju	SE
4	49048	-10.940183...	-37.070850...	aracaju	SE
5	49050	-10.927157...	-37.063078...	aracaju	SE
6	49015	-10.923370...	-37.045169...	aracaju	SE
7	49045	-10.930406...	-37.067178...	aracaju	SE
8	49052	-10.922973...	-37.057752...	aracaju	SE
9	49044	-10.992080...	-37.103470...	aracaju	SE
10	49048	-10.940235...	-37.071043...	aracaju	SE

Order Items Table:

SELECT * FROM `business-case-study-sql.Target_dataset.order_items` LIMIT 1000



Query results

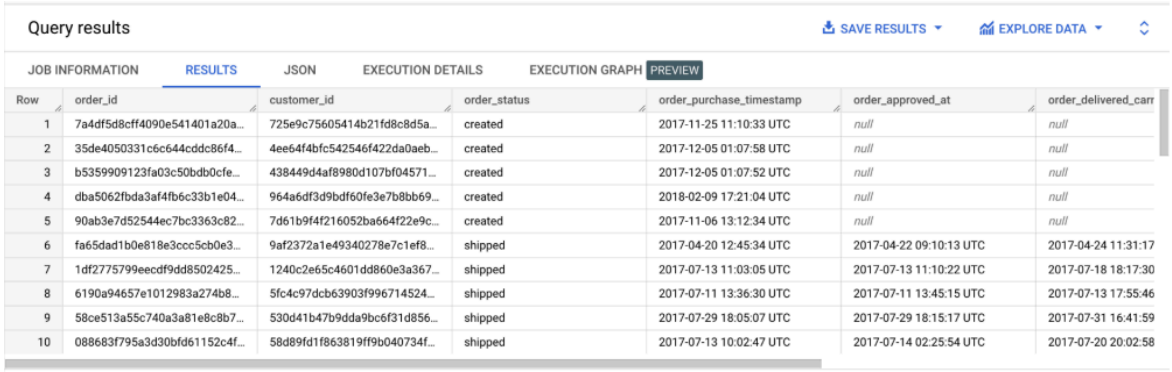
SAVE RESULTS | EXPLORE DATA

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

Row	order_id	order_item_id	product_id	seller_id	shipping_limit_date	price	freight_value
1	f09e36e258656850b92657ac5...	1	44d53f1240d6332232e4393c0...	b64d51f0435e884e8de603b16...	2018-07-09 13:31:36 UTC	3.0	12.79
2	f9ccaff7267fd0cf076e795b1fa...	1	44d53f1240d6332232e4393c0...	b64d51f0435e884e8de603b16...	2018-08-14 14:04:44 UTC	3.0	15.23
3	c79bdf061e22288609201ec60...	1	5304ff3fa35856a156e1170a60...	cf6f6bc4df3999b9c6440f124f...	2017-05-12 19:05:20 UTC	3.5	8.72
4	37193e64eb9a46b7f3197762f...	1	98224bfc1eaadb3a394ec334c...	ce616e1913288884e7742faac...	2018-06-28 01:30:49 UTC	3.5	7.39
5	95d6357ffe41aa6d2998852a7...	1	98224bfc1eaadb3a394ec334c...	ce616e1913288884e7742faac...	2018-06-12 19:15:14 UTC	3.5	18.23
6	95d6357ffe41aa6d2998852a7...	2	98224bfc1eaadb3a394ec334c...	ce616e1913288884e7742faac...	2018-06-12 19:15:14 UTC	3.5	18.23
7	95d6357ffe41aa6d2998852a7...	3	98224bfc1eaadb3a394ec334c...	ce616e1913288884e7742faac...	2018-06-12 19:15:14 UTC	3.5	18.23
8	95d6357ffe41aa6d2998852a7...	4	98224bfc1eaadb3a394ec334c...	ce616e1913288884e7742faac...	2018-06-12 19:15:14 UTC	3.5	18.23
9	95d6357ffe41aa6d2998852a7...	5	98224bfc1eaadb3a394ec334c...	ce616e1913288884e7742faac...	2018-06-12 19:15:14 UTC	3.5	18.23
10	dde867f83e689b0167785b684...	1	914323edd50192310dd03435...	2c9e548be18521d1c43cde1c5...	2017-10-20 14:50:12 UTC	4.5	11.85

OrdersTable:

SELECT * FROM `business-case-study-sql.Target_dataset.orders` LIMIT1000



Query results

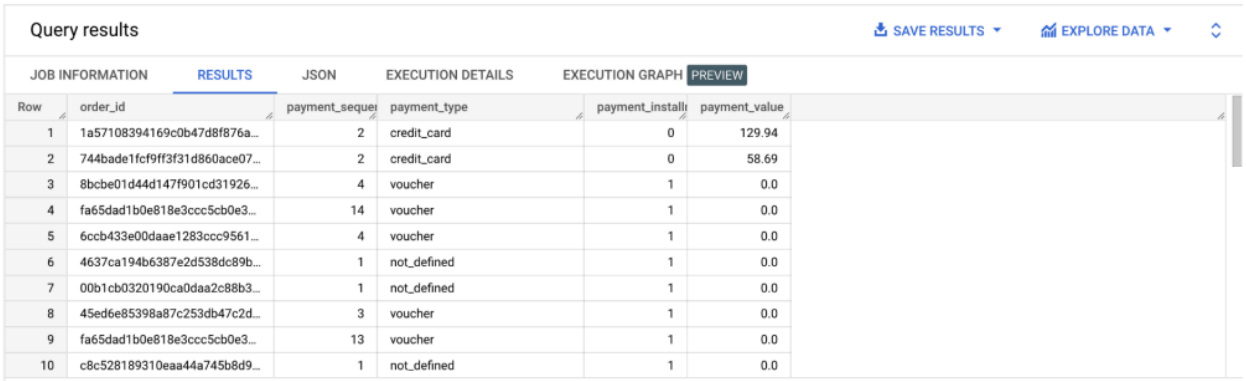
SAVE RESULTS | EXPLORE DATA

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

Row	order_id	customer_id	order_status	order_purchase_timestamp	order_approved_at	order_delivered_carr
1	7a4df5d8cff4090e541401a20a...	725e9c75605414b21fd8c8d5a...	created	2017-11-25 11:10:33 UTC	null	null
2	35de4050331c6c644cddc86f4...	4ee64f4bfc542546f422da0aeb...	created	2017-12-05 01:07:58 UTC	null	null
3	b5359909123fa03c50bdb0cfe...	438449d4af8980d107bf04571...	created	2017-12-05 01:07:52 UTC	null	null
4	dba5062fbda3af4fb6c33b1e04...	964a6df3d9bdf60fe3e7b8bb69...	created	2018-02-09 17:21:04 UTC	null	null
5	90ab3e7d52544ec7bc3363c82...	7d61b9f4f216052ba664f22e9c...	created	2017-11-06 13:12:34 UTC	null	null
6	fa65dad1b0e818e3ccc5cb0e3...	9af2372a1e49340278e7c1ef8...	shipped	2017-04-20 12:45:34 UTC	2017-04-22 09:10:13 UTC	2017-04-24 11:31:17
7	1df2775799eecdf9dd8502425...	1240c2e65c4601dd860e3a367...	shipped	2017-07-13 11:03:05 UTC	2017-07-13 11:10:22 UTC	2017-07-18 18:17:30
8	6190a94657e1012983a274b8...	5fc4c97dcb63903f996714524...	shipped	2017-07-11 13:36:30 UTC	2017-07-11 13:45:15 UTC	2017-07-13 17:55:46
9	58ce513a55c740a3a81e8c8b7...	530d41b47b9dda9bc6f31d856...	shipped	2017-07-29 18:05:07 UTC	2017-07-29 18:15:17 UTC	2017-07-31 16:41:59
10	088683f795a3d30bfd61152c4f...	58d89fd1f863819ff9b040734f...	shipped	2017-07-13 10:02:47 UTC	2017-07-14 02:25:54 UTC	2017-07-20 20:02:58

Payments:

SELECT * FROM `business-case-study-sql.Target_dataset.payments` LIMIT1000



Query results

SAVE RESULTS | EXPLORE DATA

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

Row	order_id	payment_sequei	payment_type	payment_installi	payment_value
1	1a57108394169c0b47d8f876a...	2	credit_card	0	129.94
2	744bade1fcf9ff3f31d860ace07...	2	credit_card	0	58.69
3	8bcbe01d44d147f901cd31926...	4	voucher	1	0.0
4	fa65dad1b0e818e3ccc5cb0e3...	14	voucher	1	0.0
5	6ccb433e00daae1283ccc9561...	4	voucher	1	0.0
6	4637ca194b6387e2d538dc89b...	1	not_defined	1	0.0
7	00b1cb0320190ca0daa2c88b3...	1	not_defined	1	0.0
8	45ed6e85398a87c253db47c2d...	3	voucher	1	0.0
9	fa65dad1b0e818e3ccc5cb0e3...	13	voucher	1	0.0
10	c8c528189310eaa44a745b8d9...	1	not_defined	1	0.0

Products:

SELECT * FROM `business-case-study-sql.Target_dataset.products` LIMIT 1000

Query results

SAVE RESULTS | EXPLORE DATA

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

Row	product_id	product_category	product_name_l	product_descrip	product_photos	product_weight	product_length_	product_height_	product_width_c
1	5eb564652db742ff8f28759cd8...	null	null	null	null	null	null	null	null
2	09ff539a621711667c43eba6a...	babies	60	865	3	null	null	null	null
3	2f763ba79d9cd987b2034aac7...	electronics	45	1198	2	595	8	6	6
4	a69f15dfb803d485e8933e80b...	Watches present	53	506	6	150	11	16	6
5	e1cfc87f543782b8a78b59fc85...	Garden tools	39	524	4	369	26	7	7
6	106392145fca363410d287a81...	bed table bath	58	309	1	2083	12	2	7
7	7e33f4a1c59f89da30a335b2d...	electronics	51	381	3	1075	22	5	7
8	bc9cc914f974963c07be697fc...	HEALTH BEAUTY	55	435	1	75	14	9	7
9	5ae533eac9c0e93b3f89bc9ae...	computer accessories	58	1340	1	83	12	8	8
10	67d1a56495104e195338ec900...	pet Shop	20	2153	1	275	8	13	8

Sellers:

SELECT * FROM `business-case-study-sql.Target_dataset.sellers` LIMIT 1000

Query results

SAVE RESULTS | EXPLORE DATA

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

Row	seller_id	seller_zip_code	seller_city	seller_state
1	4be2e7f96b4fd749d52dff41f8...	69900	rio branco	AC
2	327b89b872c14d1c0be7235ef...	69005	manaus	AM
3	4221a7df464f1fe2955934e30f...	48602	bahia	BA
4	651530bf5c607240ccdd89a30...	44600	ipira	BA
5	2b402d5dc42554061f8ea98d1...	44900	irece	BA
6	d03698c2efd04a549382afa66...	45658	ilheus	BA
7	c72de06d72748d1a0dfb2125b...	46430	guanambi	BA
8	fc59392d66ef99377e50356ee...	40243	salvador	BA
9	b00af24704019bd2e1b335e70...	40130	salvador	BA
10	eb4a59a06b3948e851a7d7a83...	41820	salvador	BA

Q2) Time period for which the data is given:

```
SELECT
MIN(order_purchase_timestamp) AS start_date,
MAX(order_purchase_timestamp) AS end_date
FROM `business-case-study-sql.Target_dataset.orders`
```

Query results

SAVE RESULTS | EXPLORE DATA

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

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

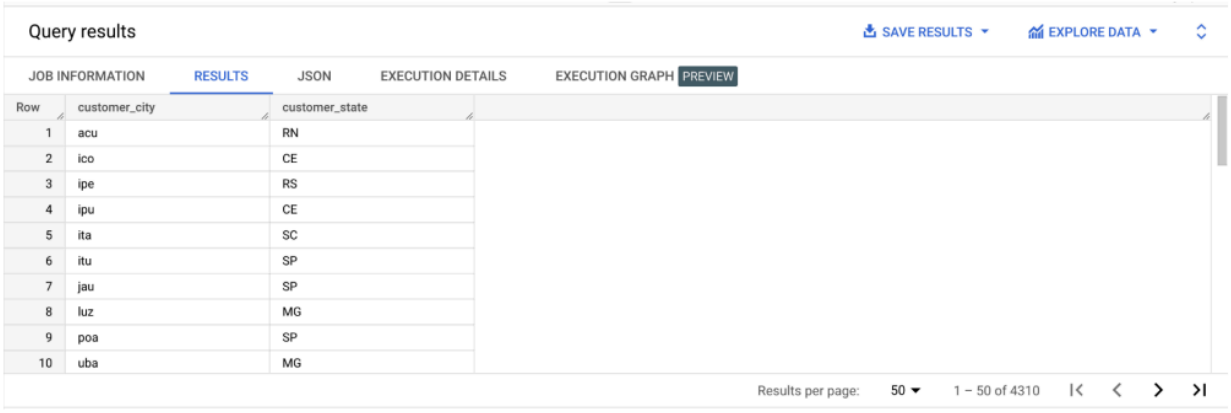
Insights:

- Start date = 04-09-2016
- End date = 17-10-2018

- Time period = 2 years, 7 months, 14 days including the end date.

Q3) Cities and States of customers ordered during the given period :

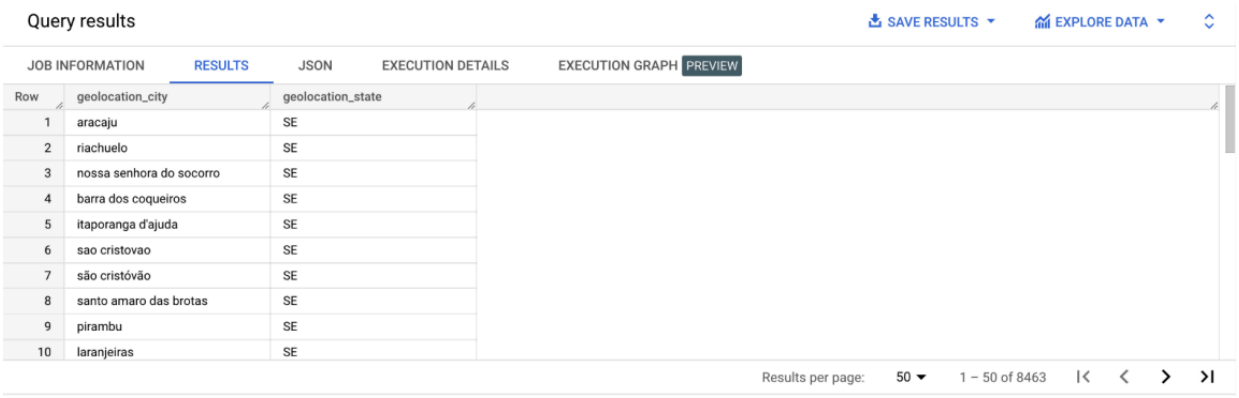
```
SELECT DISTINCT customer_city, customer_state
FROM `business-case-study-sql.Target_dataset.customers` AS c
JOIN `business-case-study-sql.Target_dataset.orders` AS o
ON c.customer_id = o.customer_id
```



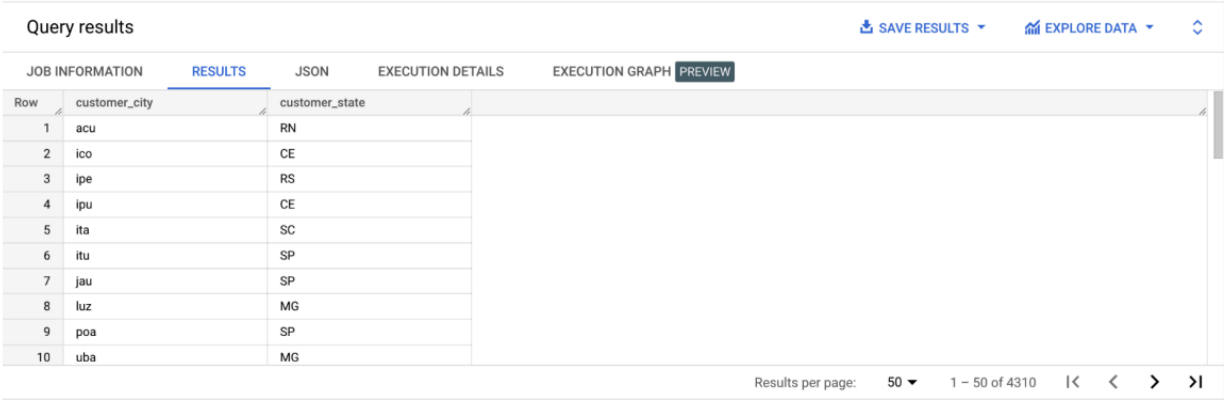
Insights:

- There are a total of 4311 rows.
- Count of unique cities = 4119
- Count of unique states = 27

```
SELECT DISTINCT geolocation_city, geolocation_state
FROM `business-case-study-sql.Target_dataset.geolocation`
```



```
SELECT DISTINCT customer_city, customer_state
FROM `business-case-study-sql.Target_dataset.customers`
```



Query results

SAVE RESULTS | EXPLORE DATA

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

Row	customer_city	customer_state
1	acu	RN
2	ico	CE
3	ipe	RS
4	ipu	CE
5	ita	SC
6	itu	SP
7	jau	SP
8	luz	MG
9	poa	SP
10	uba	MG

Results per page: 50 | 1 – 50 of 4310

Insights:

There are a total of 8464 rows.
Count of unique cities = 4119
Count of unique states = 27

Q4) In-depth Exploration:

- Is there a growing trend of e-commerce in Brazil?
- How can we describe a complete scenario?
- Can we see some seasonality with peaks at specific months?

```
SELECT
EXTRACT(MONTH FROM order_purchase_timestamp) AS order_month,
EXTRACT(YEAR FROM order_purchase_timestamp) AS order_year,
COUNT(DISTINCT order_id) AS order_count,
SUM(DISTINCT(order_id)) AS total_sales
FROM `business-case-study-sql.Target_dataset.orders`
GROUP BY order_month, order_year
ORDER BY order_year, order_month
```

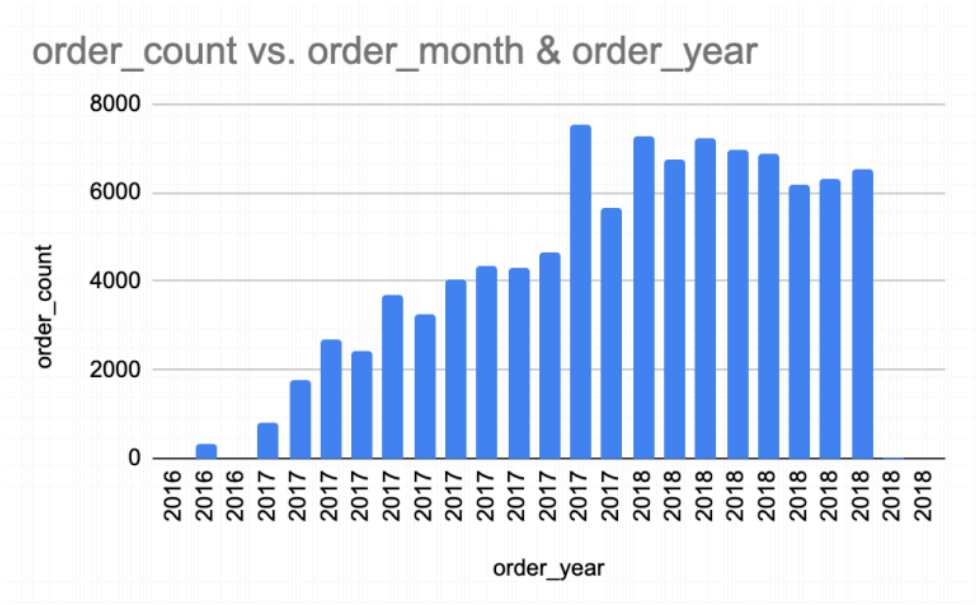
Query results

SAVE RESULTS | EXPLORE DATA

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

Row	order_month	order_year	order_count
1	9	2016	4
2	10	2016	324
3	12	2016	1
4	1	2017	800
5	2	2017	1780
6	3	2017	2682
7	4	2017	2404
8	5	2017	3700
9	6	2017	3245
10	7	2017	4026

Results per page: 50 1 – 25 of 25



Insights:

- We can clearly see that there is a growing trend on e-commerce in Brazil from 2016 to 2018.
- Nov 2017 is the month when most of the orders were placed.
- Outlier data points: (09 &12)in 2016 and (09 & 10) in 2018. Here, the number of order_count is very less than the median data. It seems the data was not captured correctly and looks corrupted. If these data points are not corrected then it may negatively impact the analysis of the entire dataset

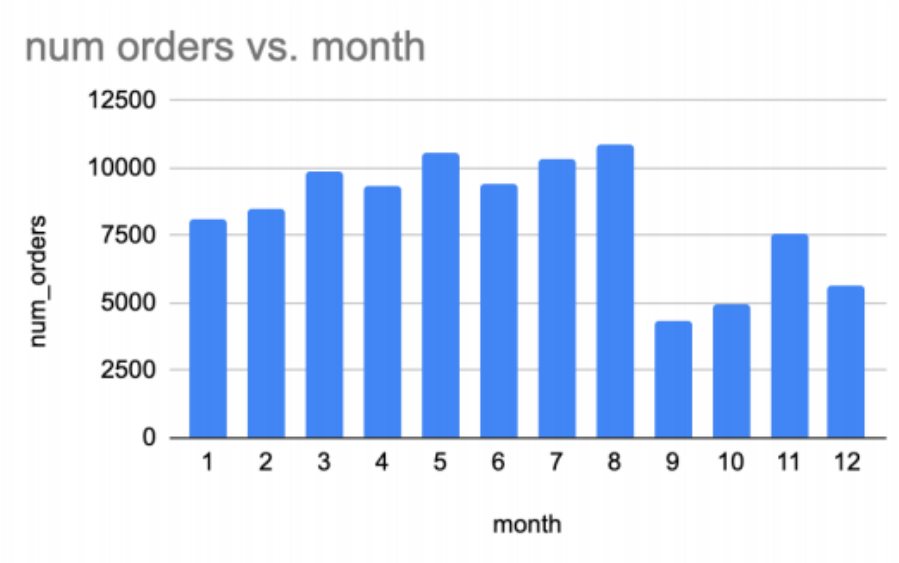
```
SELECT
EXTRACT(MONTH FROM order_purchase_timestamp) AS order_month,
EXTRACT(YEAR FROM order_purchase_timestamp) AS order_year,
COUNT(DISTINCT c.order_id) AS order_count,
SUM(p.payment_value) AS total_sales
FROM `business-case-study-sql.Target_dataset.orders` AS c
JOIN `business-case-study-sql.Target_dataset.payments` AS p
ON c.order_id = p.order_id
GROUP BY
order_month, order_year
ORDER BY
order_year, order_month
```

Query results

SAVE RESULTS | EXPLORE DATA

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

Row	order_month	order_year	order_count	total_sales
1	9	2016	3	252.24
2	10	2016	324	59090.4800...
3	12	2016	1	19.62
4	1	2017	800	138488.039...
5	2	2017	1780	291908.009...
6	3	2017	2682	449863.600...
7	4	2017	2404	417788.030...
8	5	2017	3700	592918.820...
9	6	2017	3245	511276.380...
10	7	2017	4026	592382.920...



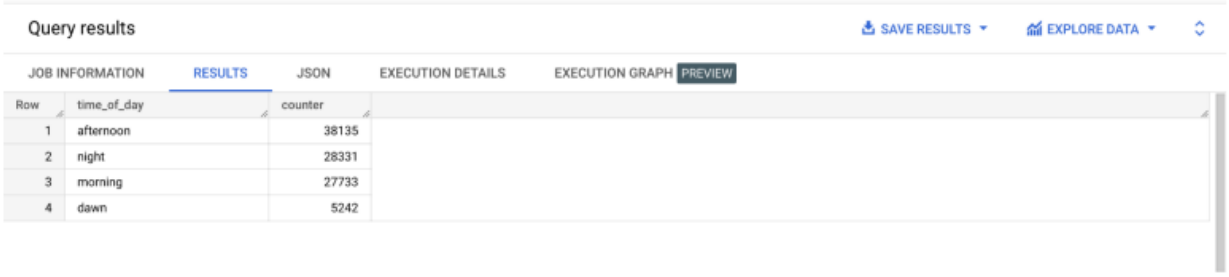
Insights:

- Seasonality chart (Month wise date added from Sept 2016 to Oct 2018)
- August has the highest number of orders. (Maybe there is Big Sale Offer/Stock Clearance offer)
- September has the lowest number of orders.

- The average number of orders for the first 8 months is higher than the last 4 months.

Q5) What time do Brazilian customers tend to buy? (Dawn - 12am-6am, Morning 6am-12pm, Afternoon - 12 noon to 6pm,or Night 6-pm - 12am)?

```
SELECT
CASE
WHEN EXTRACT(hour FROM timestamp(order_purchase_timestamp)) BETWEEN 0 AND 6
THEN 'dawn'
WHEN EXTRACT(hour FROM timestamp(order_purchase_timestamp)) BETWEEN 7 AND 12
THEN 'morning'
WHEN
EXTRACT(hour FROM timestamp(order_purchase_timestamp)) BETWEEN 13 AND 18
THEN 'afternoon'
WHEN
EXTRACT(hour FROM timestamp(order_purchase_timestamp)) BETWEEN 19 AND 23
THEN 'night'
END AS time_of_day,
COUNT(DISTINCT order_id) AS counter
FROM `business-case-study-sql.Target_dataset.orders`
GROUP BY 1
ORDER BY 2 DESC;
```

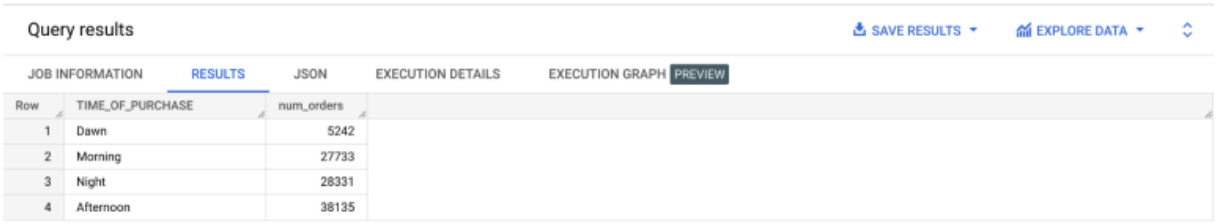


Row	time_of_day	counter
1	afternoon	38135
2	night	28331
3	morning	27733
4	dawn	5242

Insights:

- Here the time slots are: Dawn:- 12am-6am, Morning:- 7am-12pm, Afternoon:- 1pm to 6pm, Night:- 7pm - 11pm.

```
SELECT
CASE
WHEN TIME(order_purchase_timestamp) BETWEEN '00:00:00' AND '06:59:59'
THEN 'Dawn'
WHEN TIME(order_purchase_timestamp) BETWEEN '07:00:00' AND '12:59:59'
THEN 'Morning'
WHEN TIME(order_purchase_timestamp) BETWEEN '13:00:00' AND '18:59:59'
THEN 'Afternoon'
ELSE 'Night'
END AS TIME_OF_PURCHASE,
COUNT(DISTINCT order_id) AS num_orders
FROM `business-case-study-sql.Target_dataset.orders`
GROUP BY TIME_OF_PURCHASE
order by num_orders
```



Row	TIME_OF_PURCHASE	num_orders
1	Dawn	5242
2	Morning	27733
3	Night	28331
4	Afternoon	38135

Insights:

- During the afternoon time (12pm-6pm), most of the customers place orders. Reason: People are awake during this time period.So, most advertisements can be shown during this time period.
- During morning and night, almost equal amounts of orders are placed.
- During dawn timing (12am-6am), very few people order.Reason: People are mostly asleep during this time.

Q6) Evolution of E-commerce orders in the Brazil region:

- **Get month on month orders by states**

```
SELECT
EXTRACT(month FROM timestamp(order_purchase_timestamp)) AS month,
g.geolocation_state,
COUNT(1) AS num_orders
FROM `business-case-study-sql.Target_dataset.orders` o
INNER JOIN `business-case-study-sql.Target_dataset.customers` c
```

```
ON o.customer_id = c.customer_id
INNER JOIN `business-case-study-sql.Target_dataset.geolocation` g
ON c.customer_zip_code_prefix = g.geolocation_zip_code_prefix
GROUP BY g.geolocation_state, month
ORDER BY geolocation_state DESC, month ASC
```

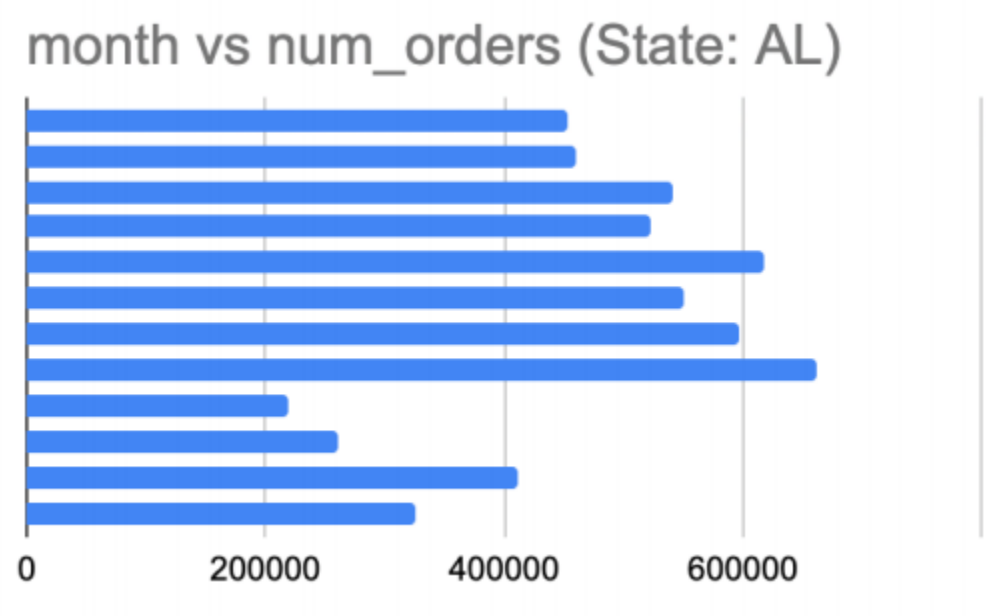
Query results

SAVE RESULTS | EXPLORE DATA

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

Row	month	geolocation_state	num_orders
1	1	TO	1544
2	2	TO	1287
3	3	TO	1626
4	4	TO	2379
5	5	TO	2691
6	6	TO	1577
7	7	TO	743
8	8	TO	1603
9	9	TO	1236
10	10	TO	1010

Results per page: 50 | 1 – 50 of 322



- **Distribution of customers across the states in Brazil**

```
SELECT g.geolocation_state, COUNT(DISTINCT (c.customer_unique_id)) AS num_customers
FROM `business-case-study-sql.Target_dataset.customers` c
INNER JOIN `business-case-study-sql.Target_dataset.geolocation` g
ON c.customer_zip_code_prefix = g.geolocation_zip_code_prefix
GROUP BY g.geolocation_state
ORDER BY num_customers DESC;
```

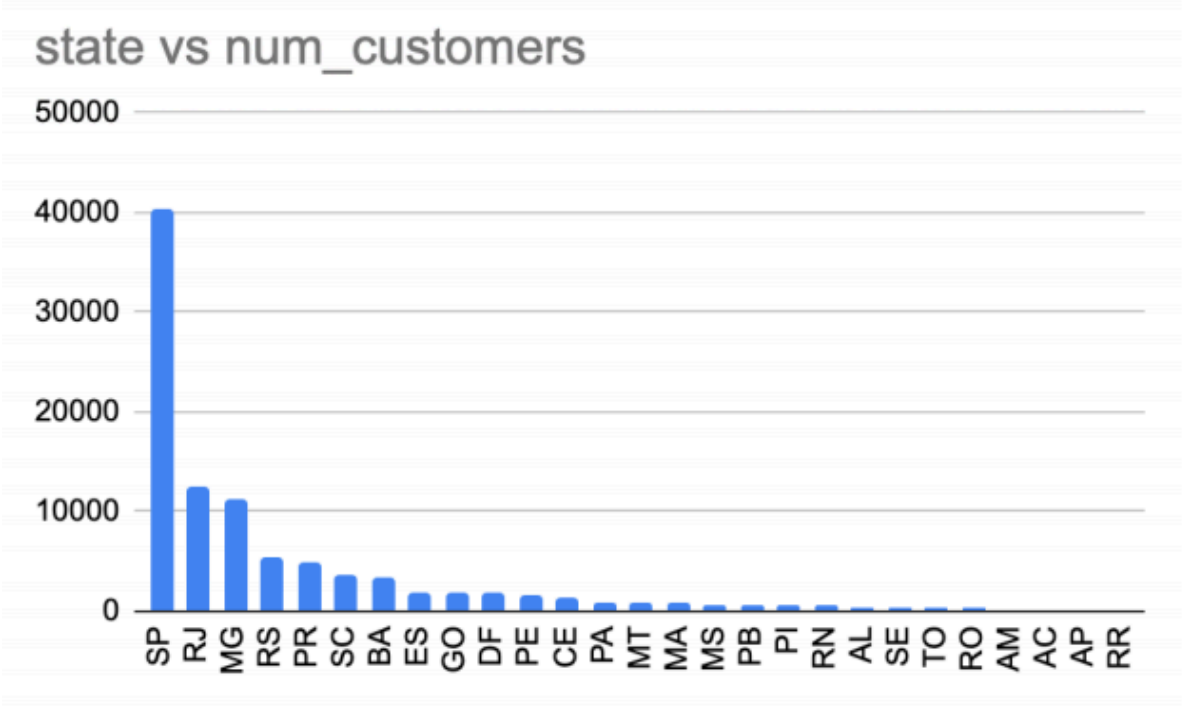
Query results

SAVE RESULTS | EXPLORE DATA

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

Row	geolocation_state	num_customers
1	SP	40287
2	RJ	12372
3	MG	11248
4	RS	5284
5	PR	4871
6	SC	3547
7	BA	3268
8	ES	1959
9	GO	1944
10	DF	1913

Results per page: 50 1 – 27 of 27



Q7) Payment type analysis:

- **Month over Month count of orders for different payment types**

```
WITH
cte_table AS
 (
SELECT
EXTRACT(month FROM timestamp(o.order_purchase_timestamp)) AS month,
EXTRACT(year FROM timestamp(o.order_purchase_timestamp)) AS year,
(sum(price) / COUNT( distinct o.order_id)) AS price_per_order,
(sum(freight_value) / COUNT(distinct o.order_id)) AS freight_per_order
FROM `business-case-study-sql.Target_dataset.orders` o
```

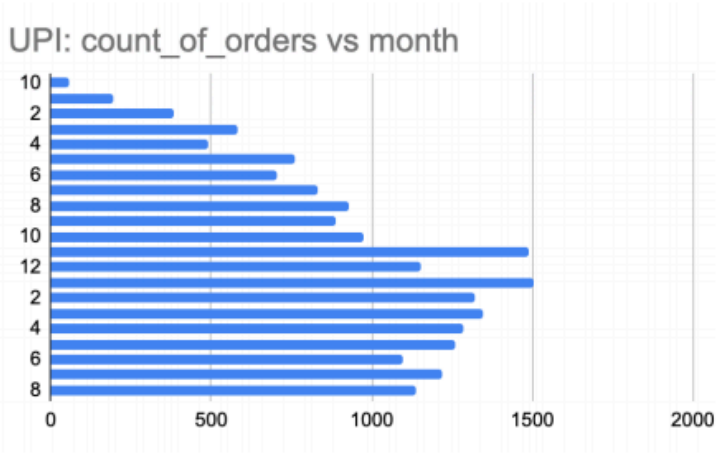
```
INNER JOIN `business-case-study-sql.Target_dataset.order_items` i
ON o.order_id = i.order_id
GROUP BY year, month
)
SELECT (price_per_order), (freight_per_order), month, year
FROM cte_table
order by payment_type, year asc, month asc ;
```

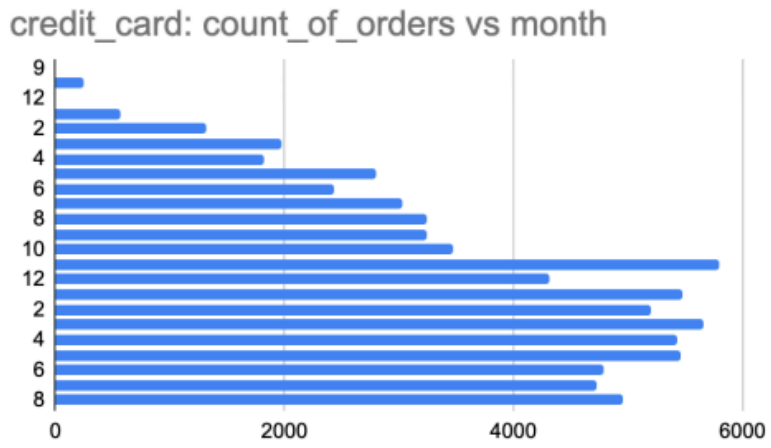
Query results

SAVE RESULTS | EXPLORE DATA

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

Row	payment_type	count_of_orders	month	year
1	UPI	58	10	2016
2	UPI	193	1	2017
3	UPI	385	2	2017
4	UPI	584	3	2017
5	UPI	491	4	2017
6	UPI	761	5	2017
7	UPI	703	6	2017
8	UPI	833	7	2017
9	UPI	928	8	2017
10	UPI	888	9	2017



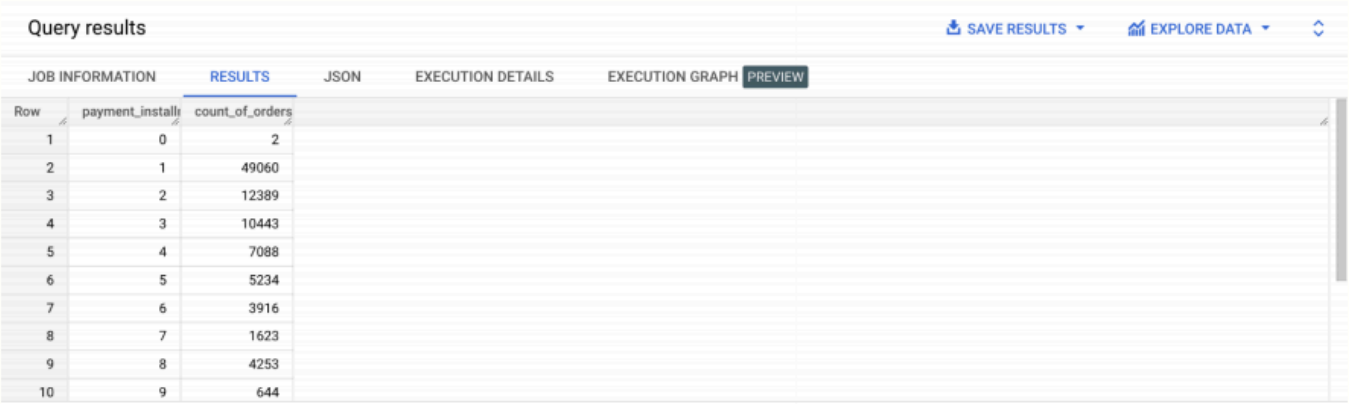


Insights:

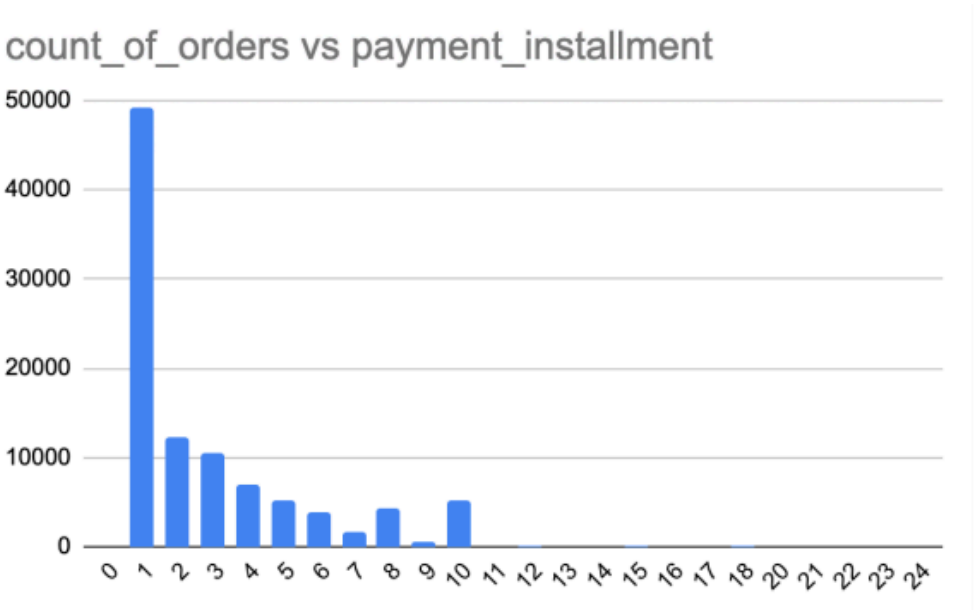
- We observe that most of the payments are done using Credit Cards.
- The second most popular method of payment is UPI.
- Then vouchers are used for payments & Debit cards are used the least for payments.
- Credit card companies can add some offers/coupons for purchasing anything to boost the usage and sales.

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

```
SELECT
payment_installments,
COUNT(DISTINCT order_id) AS count_of_orders
FROM
`business-case-study-sql.Target_dataset.payments`
GROUP BY
Payment_installments;
```



Row	payment_installments	count_of_orders
1	0	2
2	1	49060
3	2	12389
4	3	10443
5	4	7088
6	5	5234
7	6	3916
8	7	1623
9	8	4253
10	9	644



Insights:

- We observe that most of the payments are done at once.
- Then 2 to 10 installments are used for payments.
- And 11-24 installments are very less used for payments.

THANKYOU

Made by **Suryansh**
LinkedIn
E-mail - suryansh_ce21b18_67@dtu.ac.in