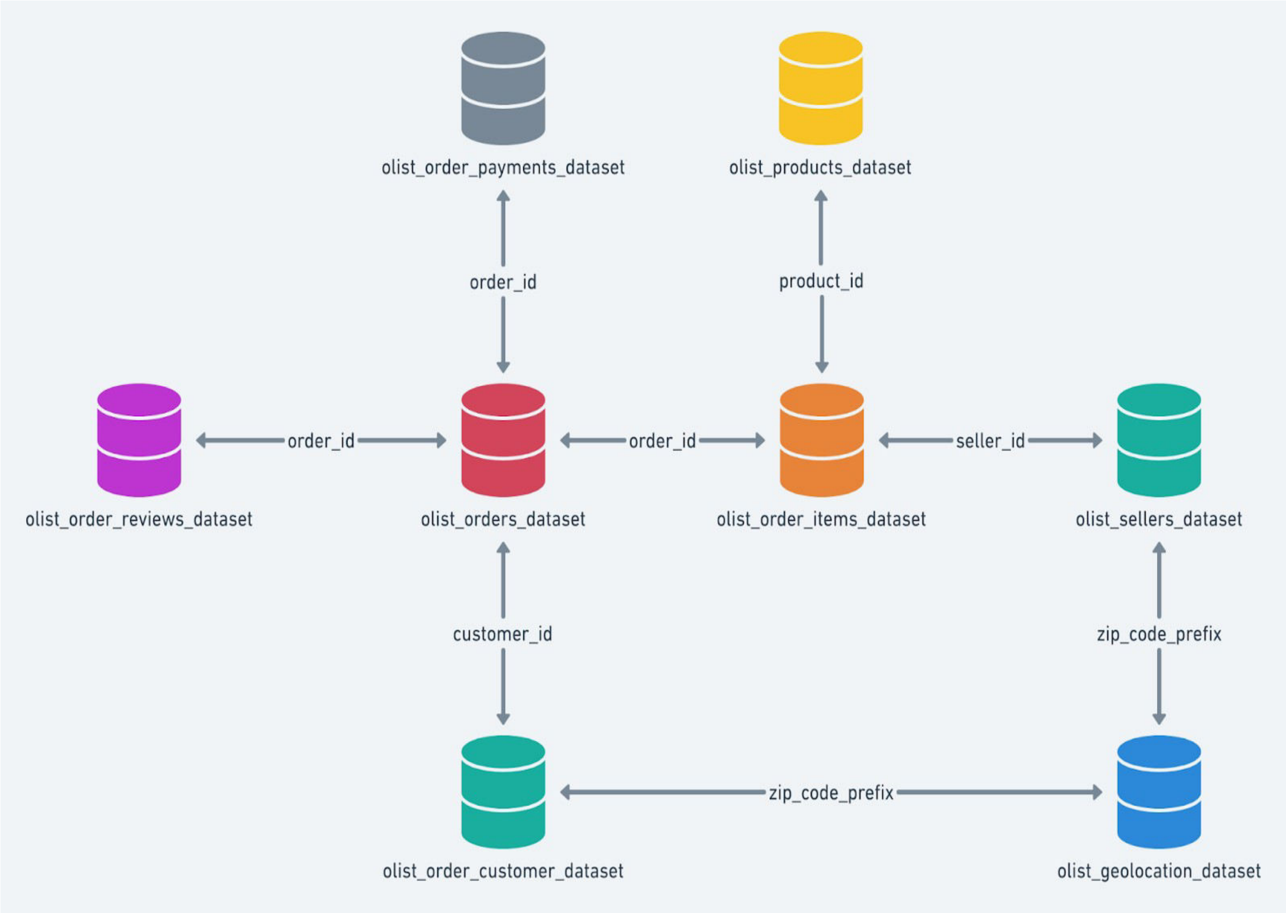
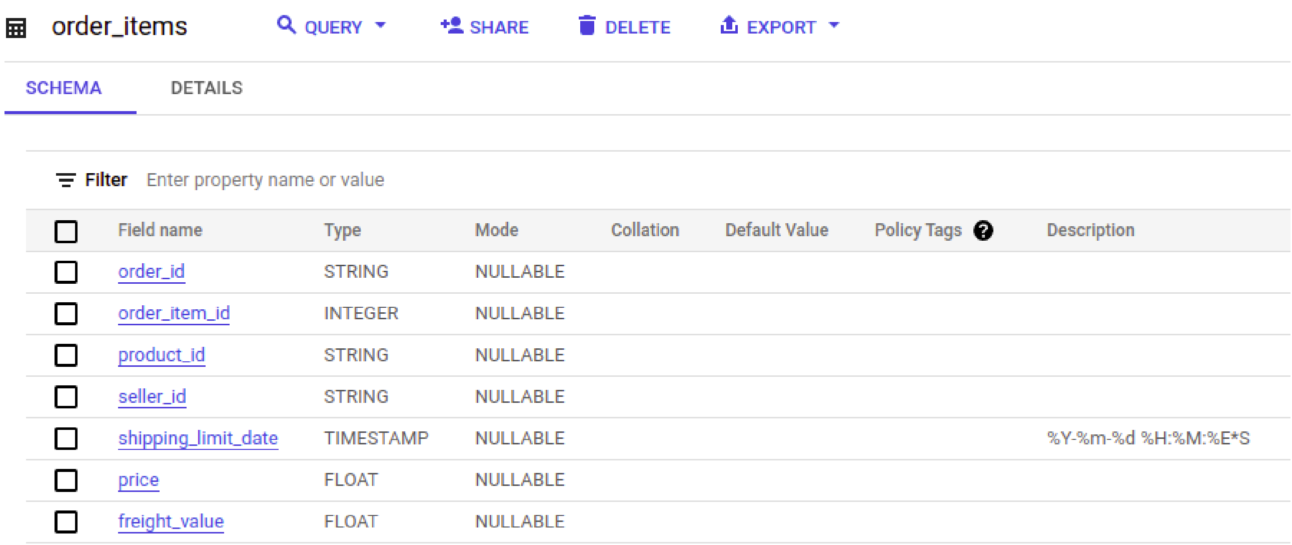
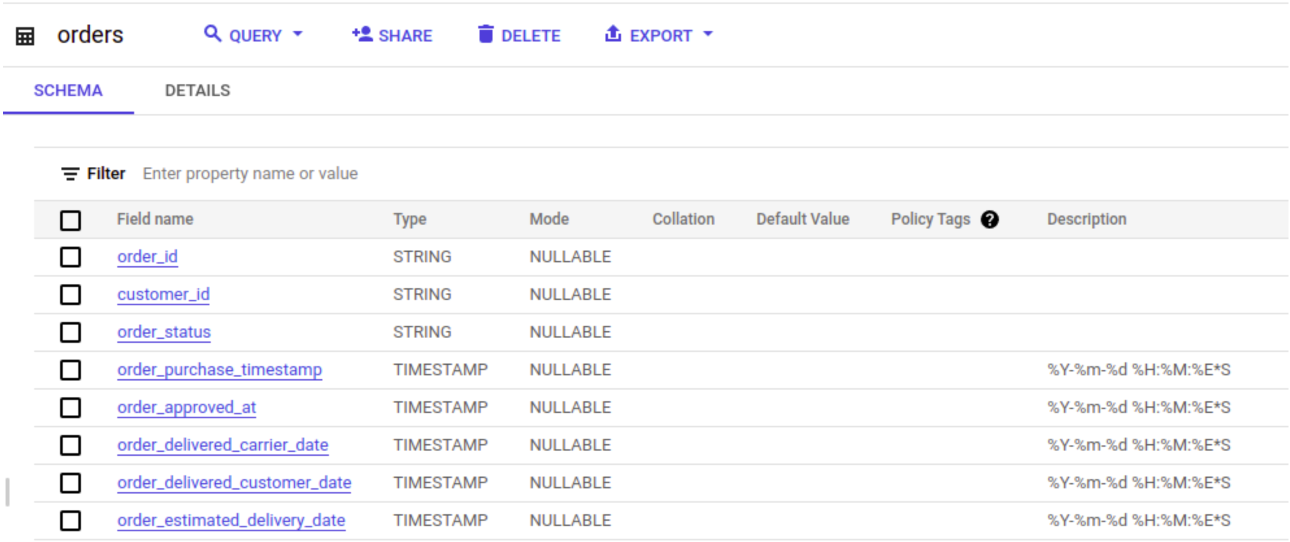
TARGET E-Commerce Brazil (SQL)



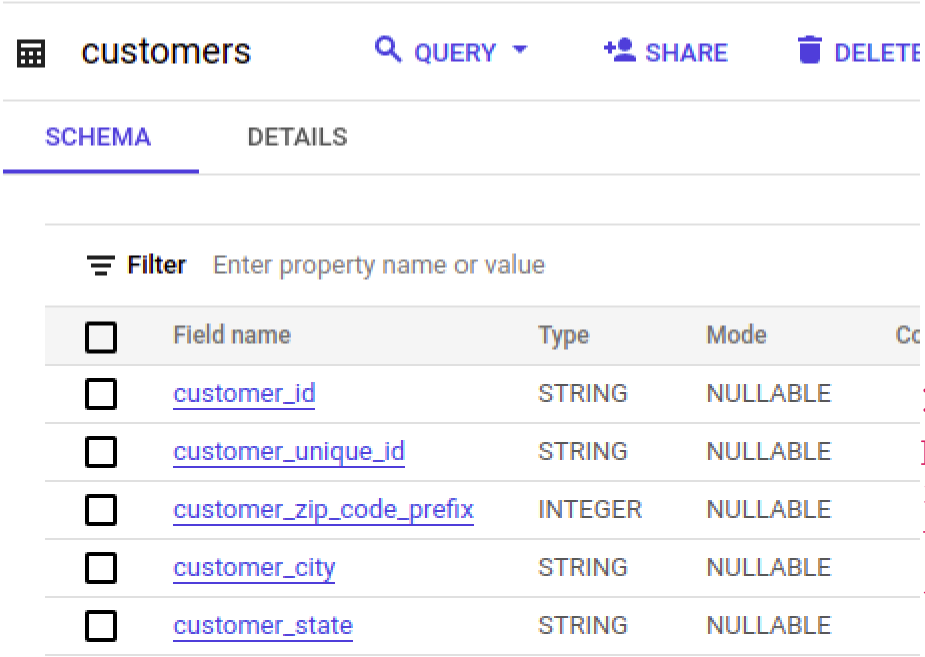
1.1. Data type of columns in a table



□	Field name	Type	Mode	Collation	Default Value	Policy Tags	Description
□	order_id	STRING	NULLABLE				
□	order_item_id	INTEGER	NULLABLE				
□	product_id	STRING	NULLABLE				
□	seller_id	STRING	NULLABLE				
□	shipping_limit_date	TIMESTAMP	NULLABLE				%Y-%m-%d %H:%M:%E*S
□	price	FLOAT	NULLABLE				
□	freight_value	FLOAT	NULLABLE				



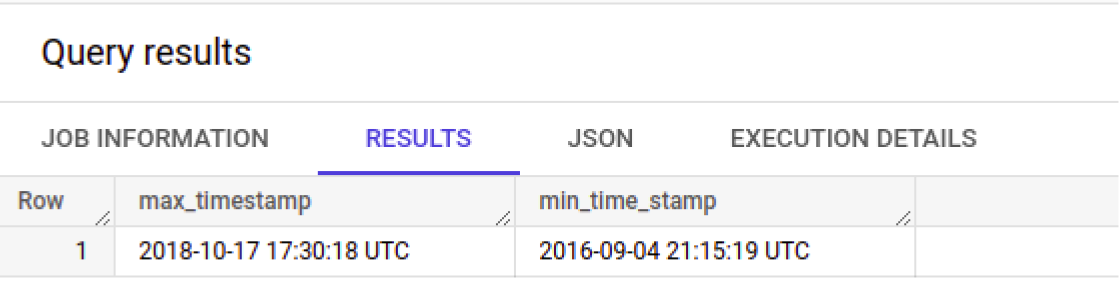
□	Field name	Type	Mode	Collation	Default Value	Policy Tags	Description
□	order_id	STRING	NULLABLE				
□	customer_id	STRING	NULLABLE				
□	order_status	STRING	NULLABLE				
□	order_purchase_timestamp	TIMESTAMP	NULLABLE				%Y-%m-%d %H:%M:%E*S
□	order_approved_at	TIMESTAMP	NULLABLE				%Y-%m-%d %H:%M:%E*S
□	order_delivered_carrier_date	TIMESTAMP	NULLABLE				%Y-%m-%d %H:%M:%E*S
□	order_delivered_customer_date	TIMESTAMP	NULLABLE				%Y-%m-%d %H:%M:%E*S
□	order_estimated_delivery_date	TIMESTAMP	NULLABLE				%Y-%m-%d %H:%M:%E*S



□	Field name	Type	Mode	Co
□	customer_id	STRING	NULLABLE	
□	customer_unique_id	STRING	NULLABLE	
□	customer_zip_code_prefix	INTEGER	NULLABLE	
□	customer_city	STRING	NULLABLE	
□	customer_state	STRING	NULLABLE	

1.2. Time period for which the data is given

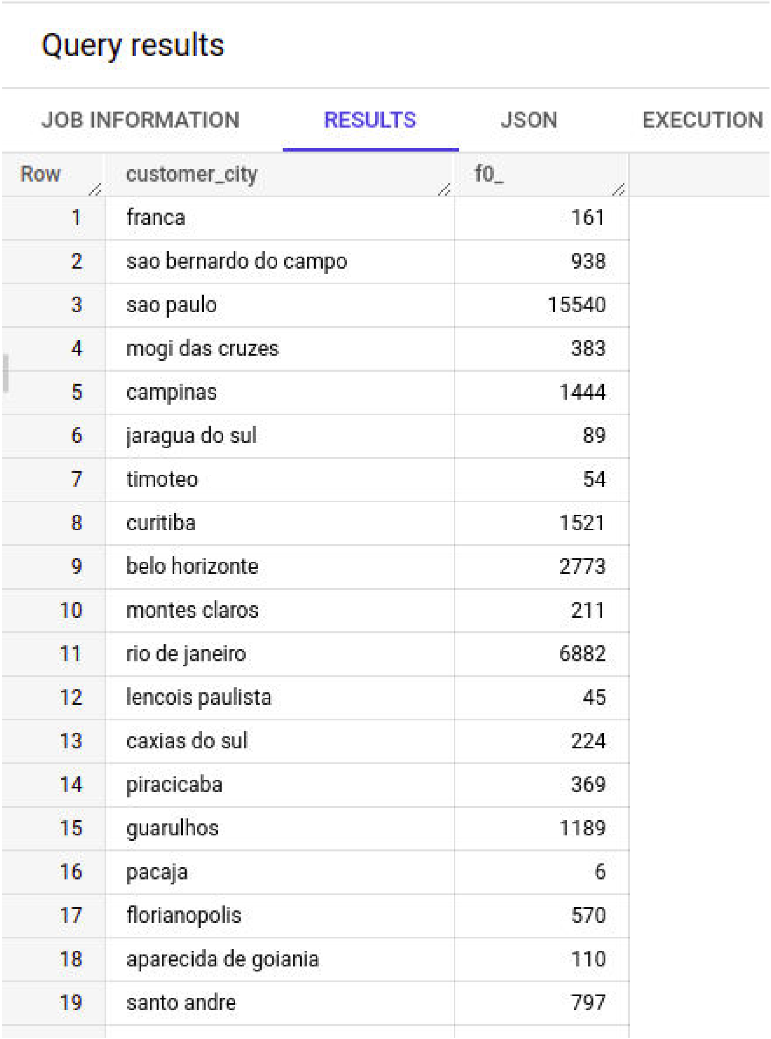
```
Select
max(order_purchase_timestamp) as
max_timestamp,min(order_purchase_timestamp) as
min_time_stamp
From `Target.orders`
```



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

1.3. Cities and States covered in the dataset

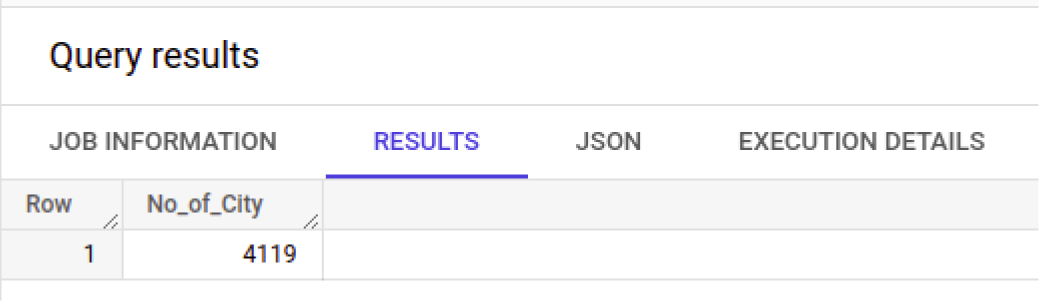
```
Select distinct customer_city,count(customer_city)
From `Target.customers`
GROUP BY customer_city
```



Row	customer_city	f0_
1	franca	161
2	sao bernardo do campo	938
3	sao paulo	15540
4	mogi das cruzes	383
5	campinas	1444
6	jaragua do sul	89
7	timoteo	54
8	curitiba	1521
9	belo horizonte	2773
10	montes claros	211
11	rio de janeiro	6882
12	lencois paulista	45
13	caxias do sul	224
14	piracicaba	369
15	guarulhos	1189
16	pacaja	6
17	florianopolis	570
18	aparecida de goiania	110
19	santo andre	797

#Cities and States covered in the dataset

```
Select Count(*) AS No_of_City
From(Select distinct
customer_city,count(customer_city)
From `Target.customers`
GROUP BY customer_city) x
```

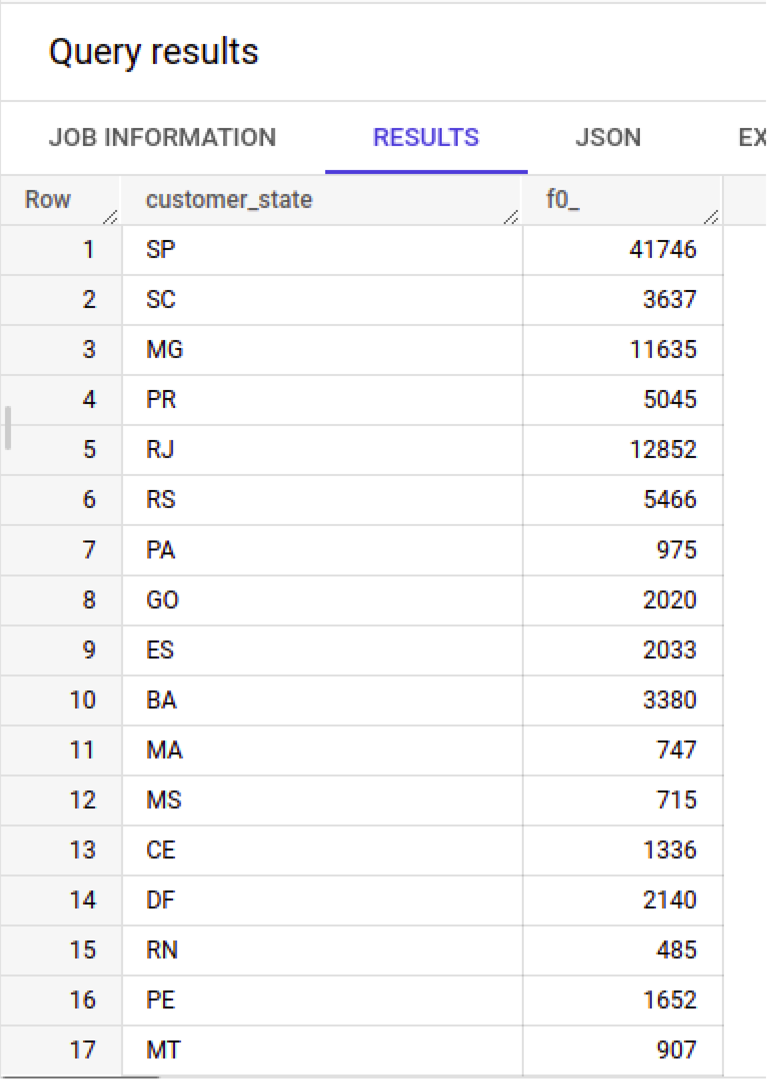


Query results

Row	No_of_City
1	4119

#Cities and States covered in the dataset

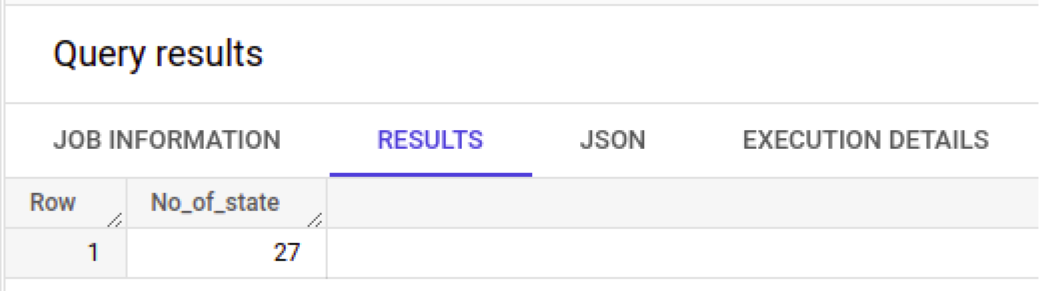
```
Select distinct customer_state,count(customer_state)
From `Target.customers`
GROUP BY customer_state
```



Row	customer_state	f0_
1	SP	41746
2	SC	3637
3	MG	11635
4	PR	5045
5	RJ	12852
6	RS	5466
7	PA	975
8	GO	2020
9	ES	2033
10	BA	3380
11	MA	747
12	MS	715
13	CE	1336
14	DF	2140
15	RN	485
16	PE	1652
17	MT	907

#Cities and States covered in the dataset

```
Select Count(*) AS No_of_state
From(Select distinct
customer_state,count(customer_state)
From `Target.customers`
GROUP BY customer_state) x
```

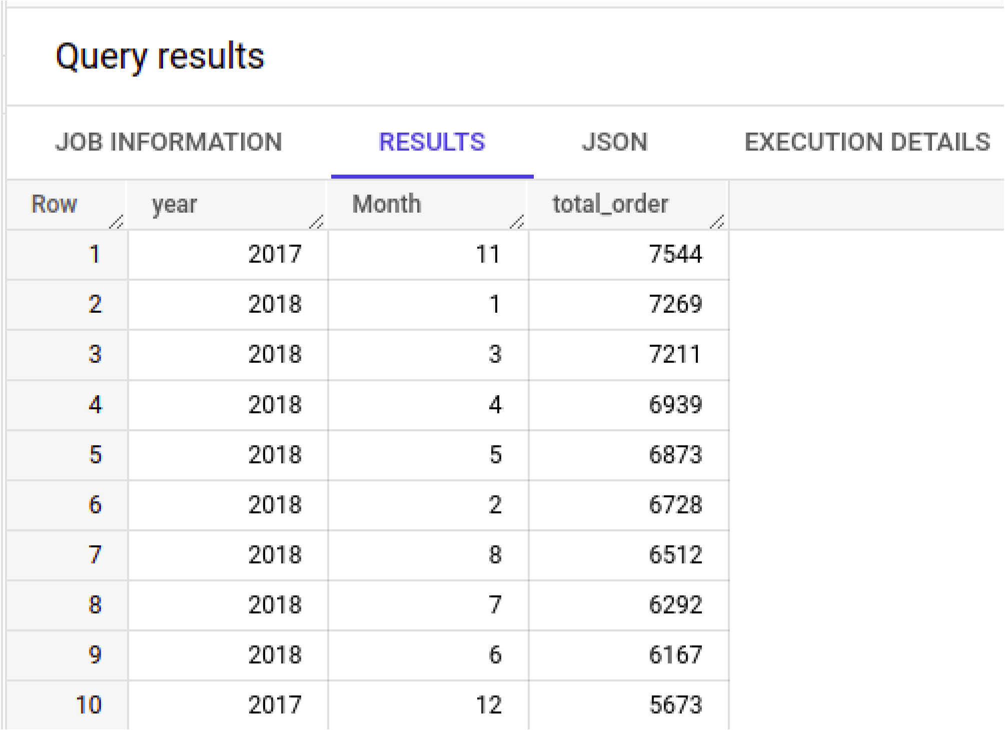


Row	No_of_state
1	27

2. In-depth Exploration:

2.1. 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?

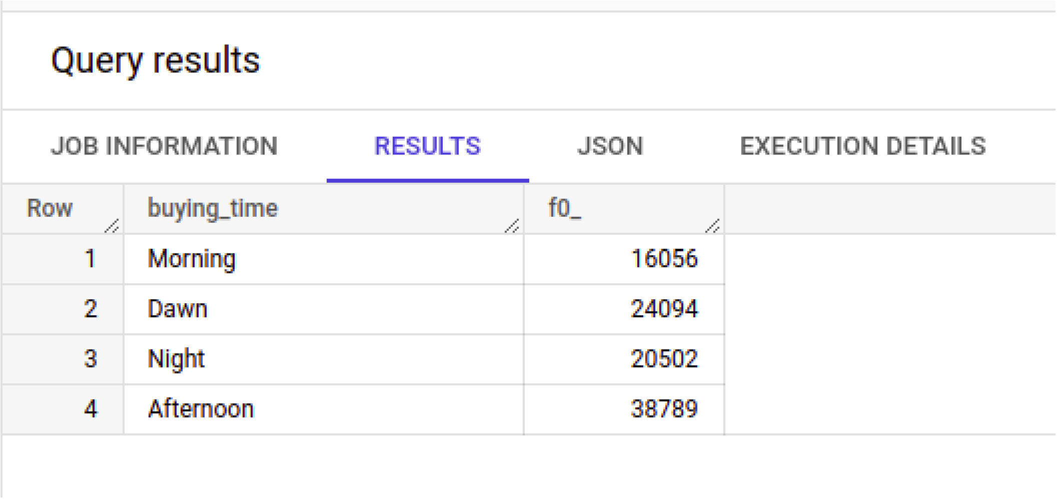
```
Select year,Month,count(*) as total_order
From(SELECT order_purchase_timestamp,
EXTRACT(Year FROM order_purchase_timestamp) as Year,
EXTRACT(Month FROM order_purchase_timestamp) as Month,
EXTRACT(Day FROM order_purchase_timestamp) as day
FROM `big-query-358521.Target.orders` ) x
group by Year,Month
Order by Year,Month
```



Row	year	Month	total_order
1	2017	11	7544
2	2018	1	7269
3	2018	3	7211
4	2018	4	6939
5	2018	5	6873
6	2018	2	6728
7	2018	8	6512
8	2018	7	6292
9	2018	6	6167
10	2017	12	5673

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

```
Select buying_time,count(buying_time)
From(Select *,
CASE When hour Between 4 and 12
Then 'Morning'
When hour Between 12 and 16
Then 'Afternoon'
When hour Between 16 and 20
Then 'Dawn'
When (hour Between 0 and 4) OR (hour Between 20 and 24)
Then 'Night'
END AS buying_time
From(SELECT order_purchase_timestamp,
EXTRACT(Year FROM order_purchase_timestamp) as Year,
EXTRACT(Month FROM order_purchase_timestamp) as Month,
EXTRACT(Day FROM order_purchase_timestamp) as day,
EXTRACT(Hour FROM order_purchase_timestamp) as hour,
EXTRACT(Minute FROM order_purchase_timestamp) as minute,
EXTRACT(Second FROM order_purchase_timestamp) as sec
FROM `big-query-358521.Target.orders` ) x)y
group by buying_time
```

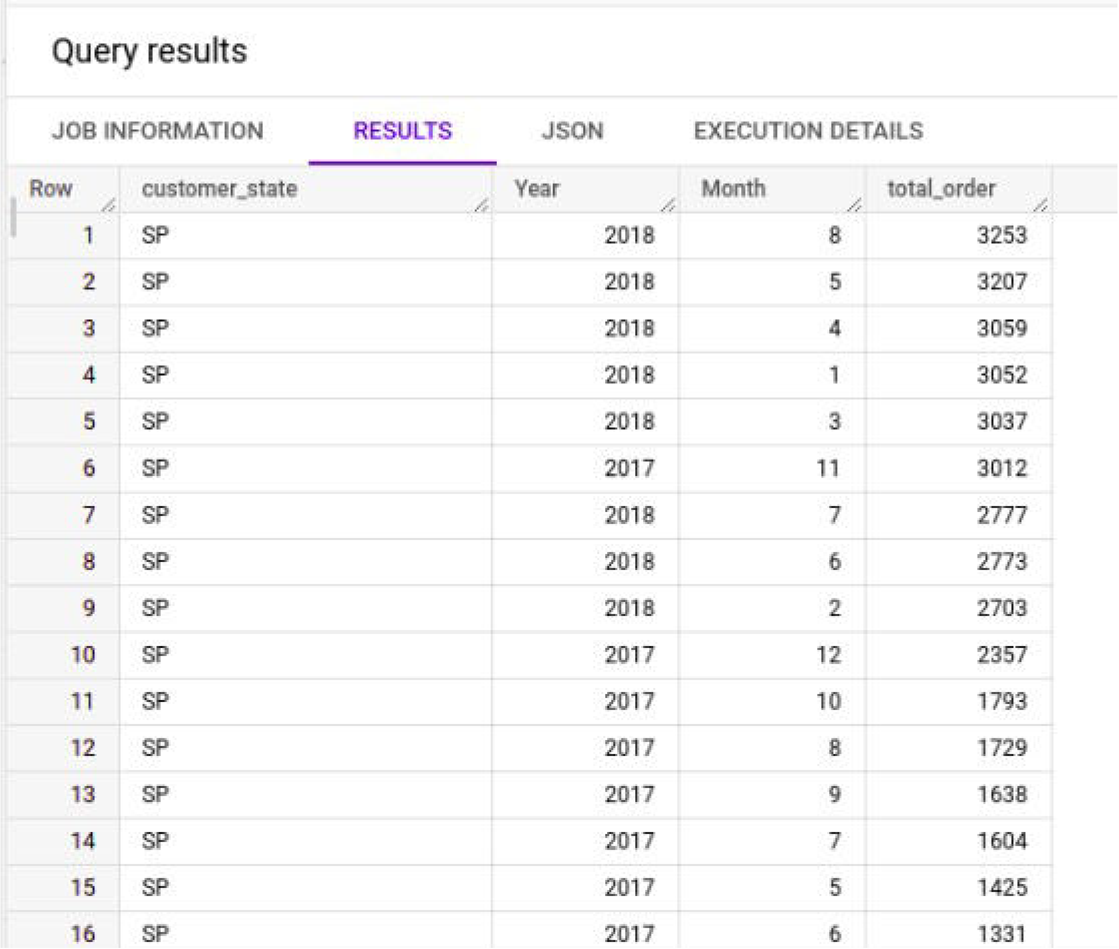


Row	buying_time	f0_
1	Morning	16056
2	Dawn	24094
3	Night	20502
4	Afternoon	38789

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

3.1.Get month on month orders by region, states

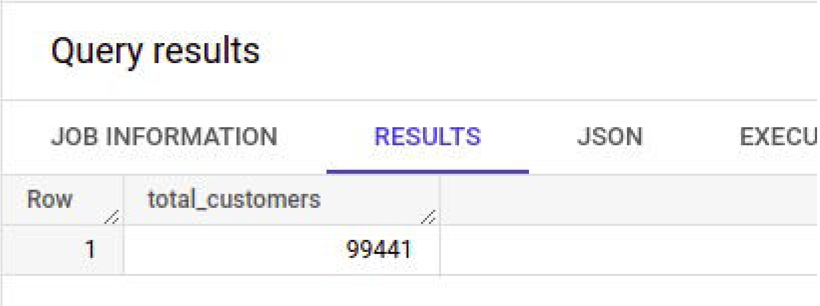
```
SELECT customer_state,Year,Month,COUNT(Month) as total_order
FROM
(SELECT
o.order_id,o.customer_id,order_purchase_timestamp,order_status,cus
tomer_city,customer_state,
EXTRACT(Year FROM order_purchase_timestamp) as Year,
EXTRACT(Month FROM order_purchase_timestamp) as Month,
EXTRACT(Day FROM order_purchase_timestamp) as day
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id) x
group by customer_state,Year,Month
Order by total_order DESC
```



Row	customer_state	Year	Month	total_order
1	SP	2018	8	3253
2	SP	2018	5	3207
3	SP	2018	4	3059
4	SP	2018	1	3052
5	SP	2018	3	3037
6	SP	2017	11	3012
7	SP	2018	7	2777
8	SP	2018	6	2773
9	SP	2018	2	2703
10	SP	2017	12	2357
11	SP	2017	10	1793
12	SP	2017	8	1729
13	SP	2017	9	1638
14	SP	2017	7	1604
15	SP	2017	5	1425
16	SP	2017	6	1331

3.2. How are customers distributed in Brazil

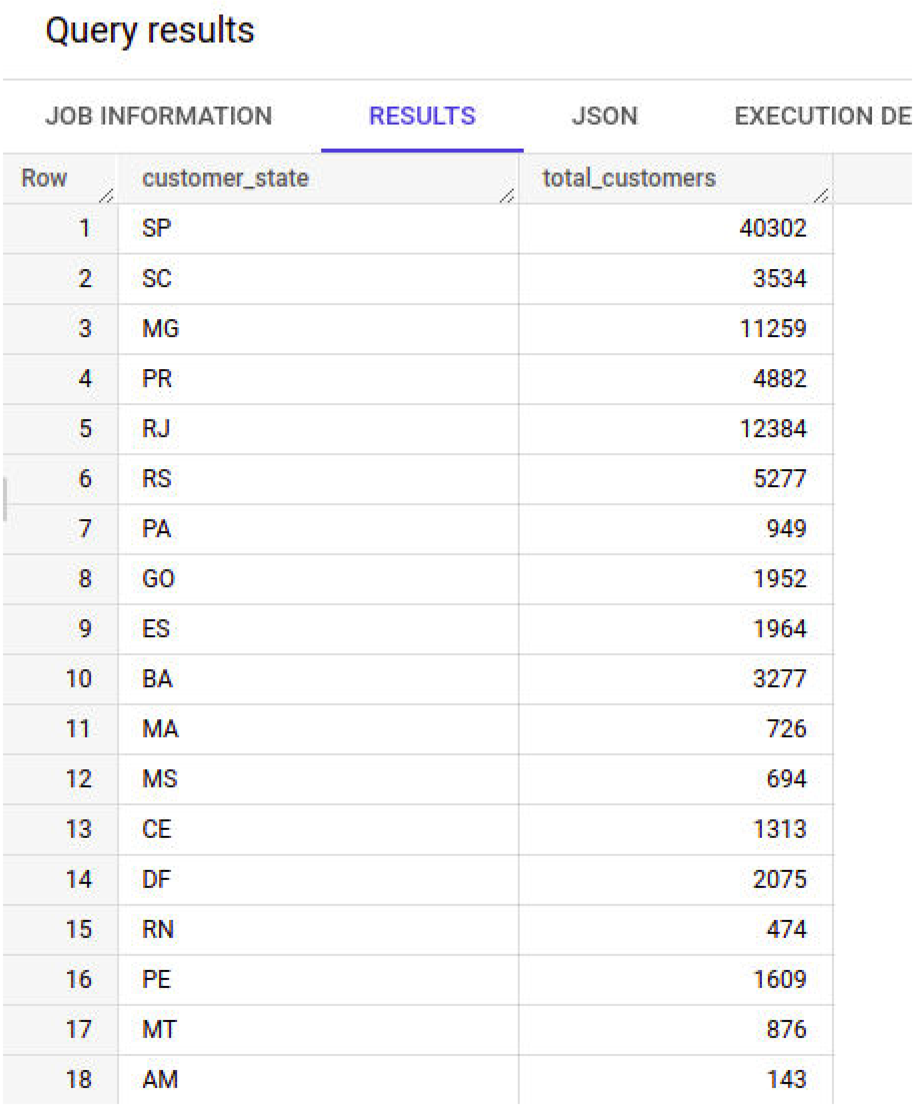
```
#Total Unique Customer
SELECT Count(customer_unique_id) as total_customers FROM `big-
query-358521.Target.customers`
```



Row	total_customers
1	99441

3.2. How are customers distributed in Brazil

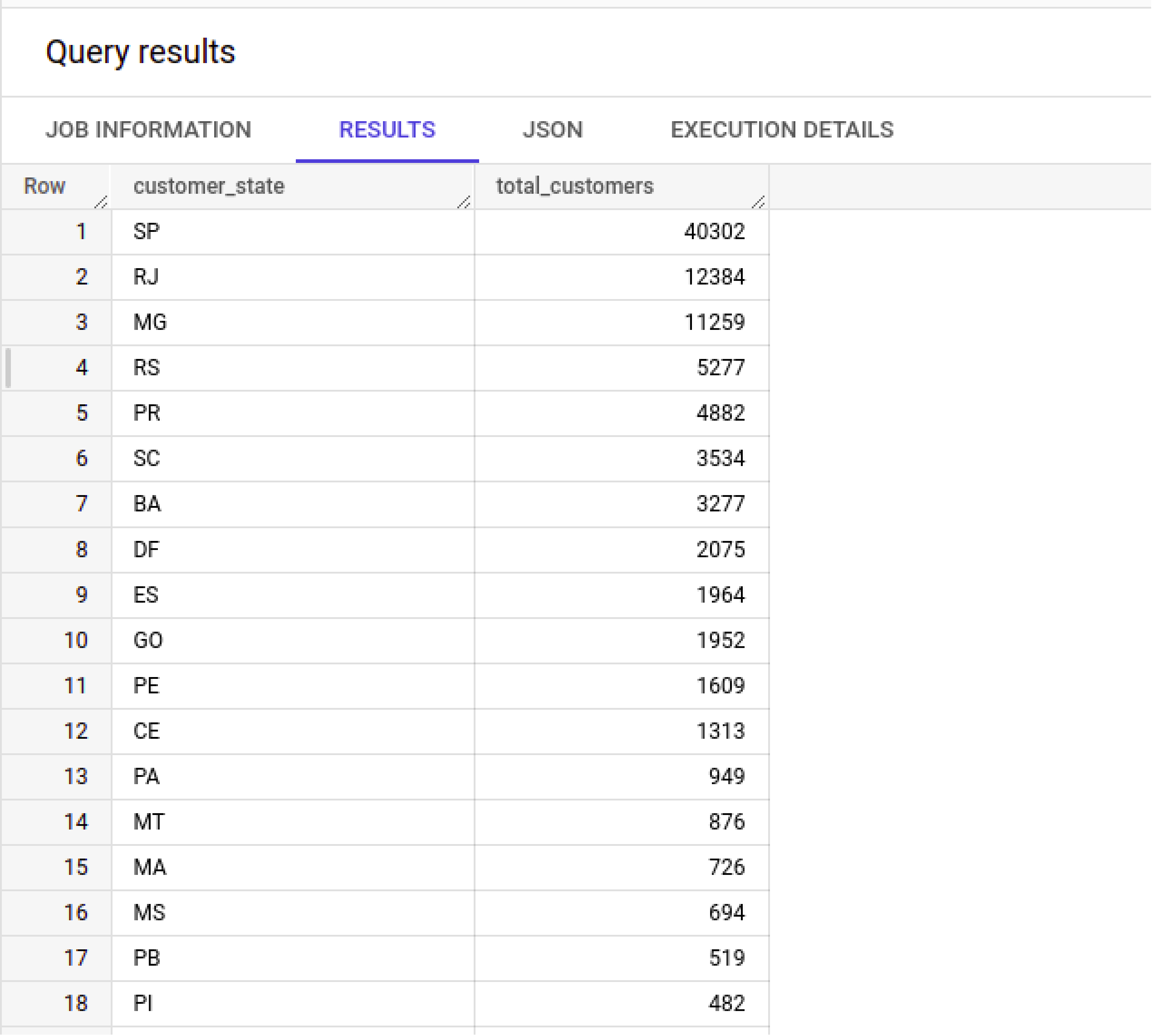
```
SELECT customer_state,Count(*) as total_customers
FROM(SELECT DISTINCT customer_unique_id,customer_state
FROM `big-query-358521.Target.customers`) x
group by customer_state
```



Row	customer_state	total_customers
1	SP	40302
2	SC	3534
3	MG	11259
4	PR	4882
5	RJ	12384
6	RS	5277
7	PA	949
8	GO	1952
9	ES	1964
10	BA	3277
11	MA	726
12	MS	694
13	CE	1313
14	DF	2075
15	RN	474
16	PE	1609
17	MT	876
18	AM	143

3.2. How are customers distributed in Brazil

```
SELECT customer_state,Count(*) as total_customers
FROM(SELECT DISTINCT customer_unique_id,customer_state
FROM `big-query-358521.Target.customers`) x
group by customer_state
ORDER BY total_customers DESC
```

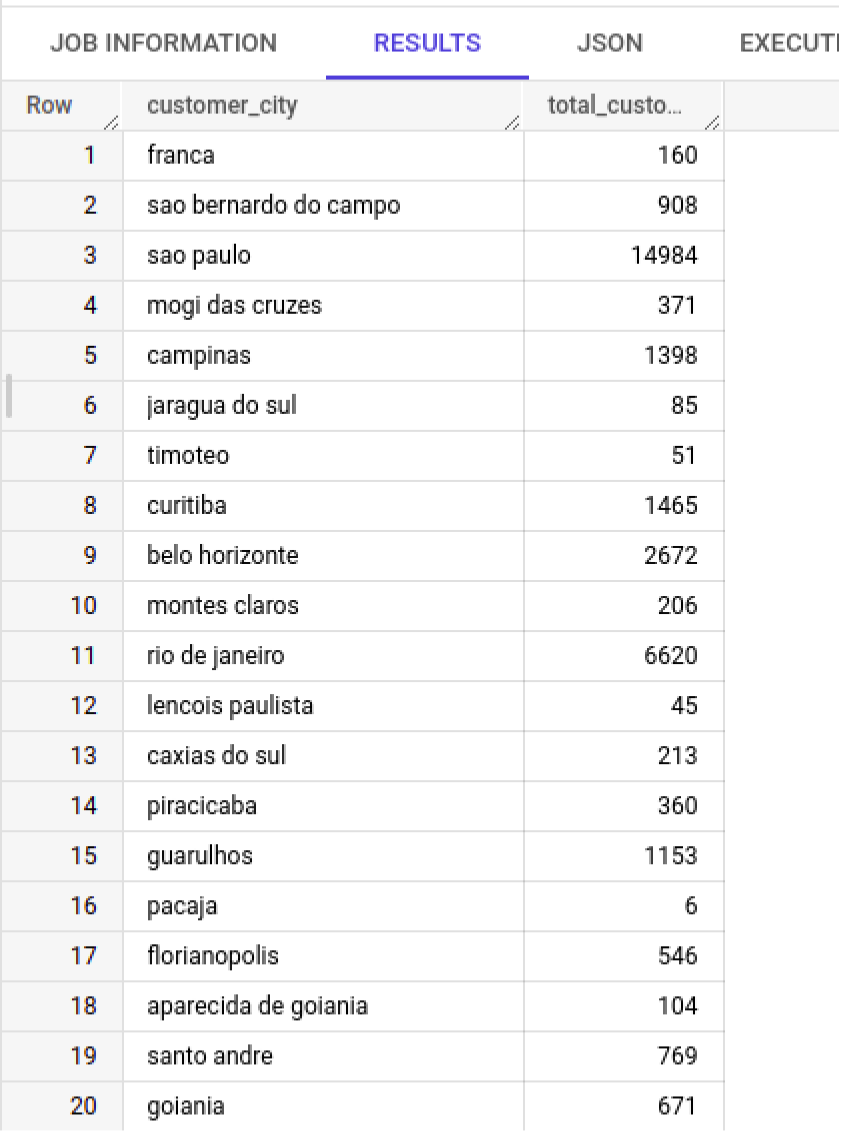


Query results

Row	customer_state	total_customers
1	SP	40302
2	RJ	12384
3	MG	11259
4	RS	5277
5	PR	4882
6	SC	3534
7	BA	3277
8	DF	2075
9	ES	1964
10	GO	1952
11	PE	1609
12	CE	1313
13	PA	949
14	MT	876
15	MA	726
16	MS	694
17	PB	519
18	PI	482

3.2. How are customers distributed in Brazil

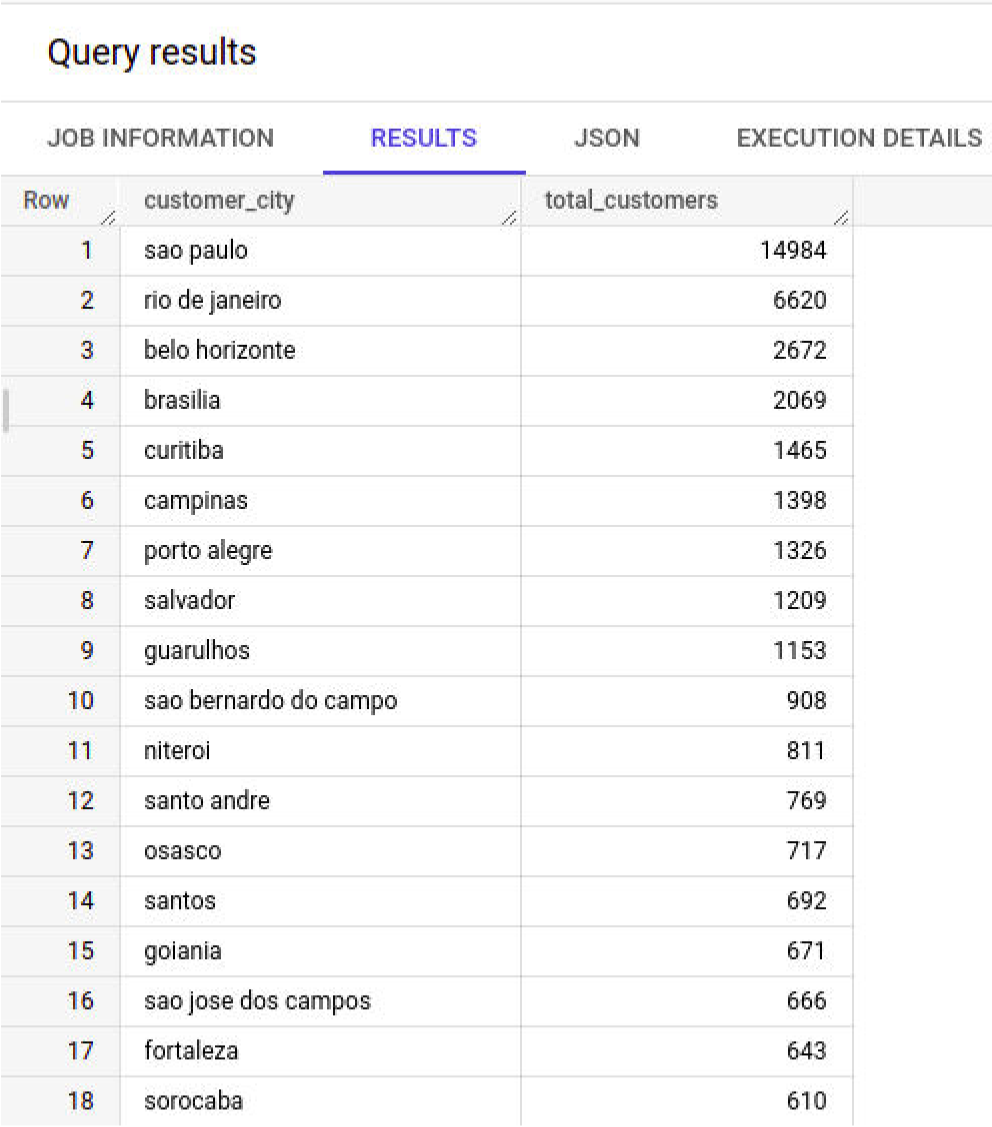
```
SELECT customer_city,Count(*) as total_customers
FROM(SELECT DISTINCT
customer_unique_id,customer_state,customer_city
FROM `big-query-358521.Target.customers`) x
group by customer_city
```



Row	customer_city	total_custo...
1	franca	160
2	sao bernardo do campo	908
3	sao paulo	14984
4	mogi das cruzes	371
5	campinas	1398
6	jaragua do sul	85
7	timoteo	51
8	curitiba	1465
9	belo horizonte	2672
10	montes claros	206
11	rio de janeiro	6620
12	lencois paulista	45
13	caxias do sul	213
14	piracicaba	360
15	guarulhos	1153
16	pacaja	6
17	florianopolis	546
18	aparecida de goiania	104
19	santo andre	769
20	goiania	671

3.2. How are customers distributed in Brazil

```
SELECT customer_city,Count(*) as total_customers
FROM(SELECT DISTINCT
customer_unique_id,customer_state,customer_city
FROM `big-query-358521.Target.customers`) x
group by customer_city
ORDER BY total_customers DESC
```

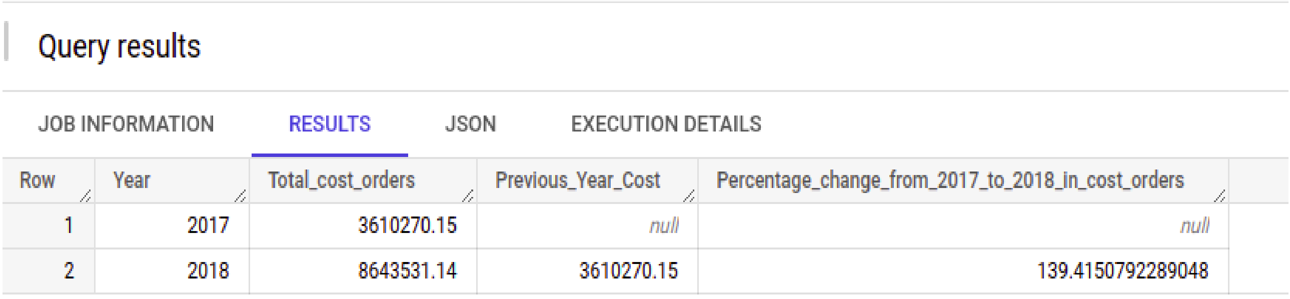


Query results

Row	customer_city	total_customers
1	sao paulo	14984
2	rio de janeiro	6620
3	belo horizonte	2672
4	brasilia	2069
5	curitiba	1465
6	campinas	1398
7	porto alegre	1326
8	salvador	1209
9	guarulhos	1153
10	sao bernardo do campo	908
11	niteroi	811
12	santo andre	769
13	osasco	717
14	santos	692
15	goiania	671
16	sao jose dos campos	666
17	fortaleza	643
18	sorocaba	610

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

```
SELECT *,(((Total_cost_orders-
Previous_Year_Cost)/Previous_Year_Cost)*100) as
Percentage_change_from_2017_to_2018_in_cost_orders
FROM(Select Year,Total_cost_orders,
LAG(Total_cost_orders) OVER(ORDER BY Total_cost_orders) as
Previous_Year_Cost
FROM
(SELECT Year,round(SUM(total_cost),2) as Total_cost_orders
FROM(SELECT *
FROM
(SELECT o.order_id,
EXTRACT(Year FROM order_purchase_timestamp) as Year,
EXTRACT(Month FROM order_purchase_timestamp) as Month,
EXTRACT(Day FROM order_purchase_timestamp) as day,
price,freight_value,
round((price+freight_value),2) as total_cost
FROM `big-query-358521.Target.orders` o
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id) x
Where (Year=2017 OR Year=2018) AND (Month Between 1 and 8)) y
group by year
Order by Year DESC) z) u
```

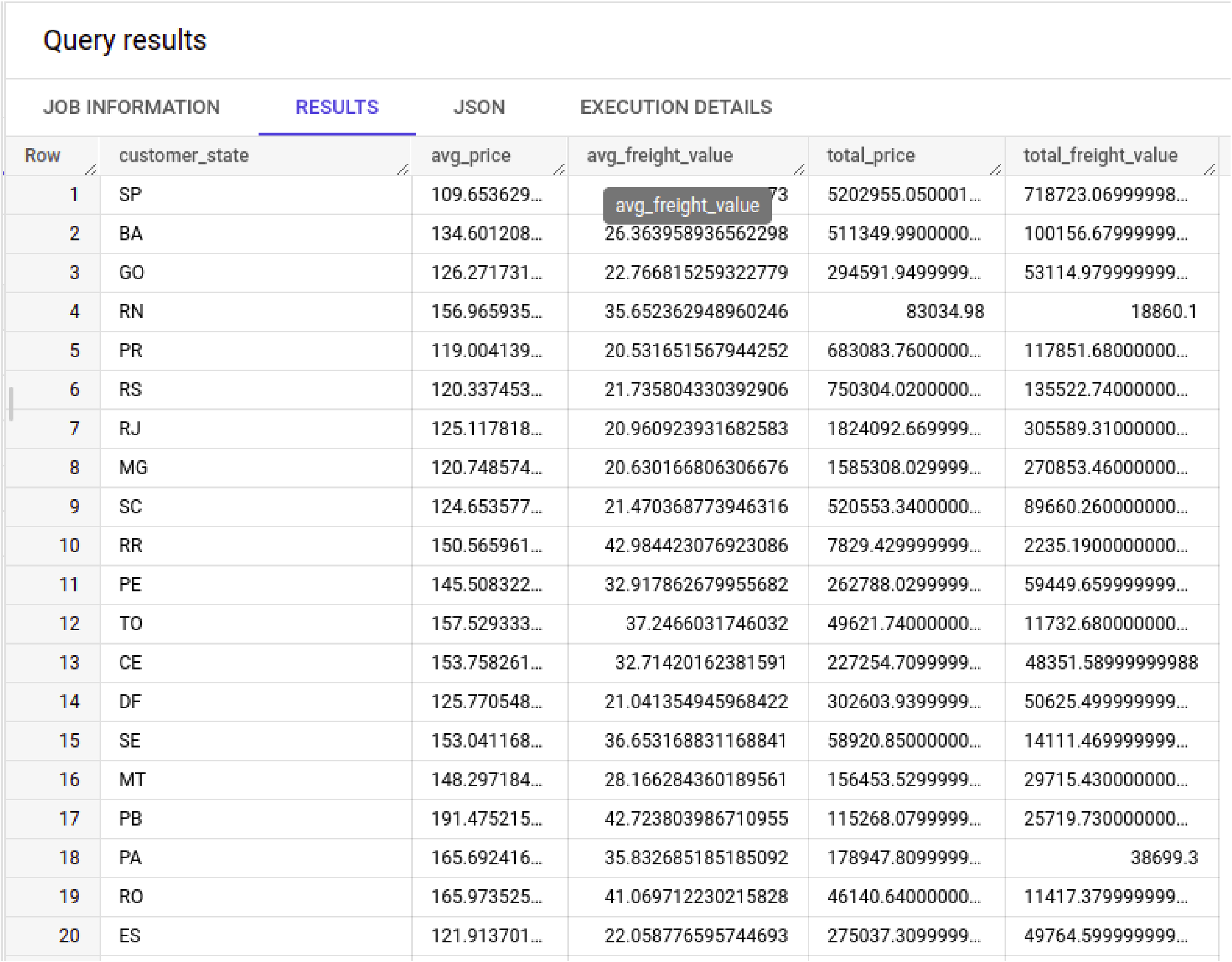


Query results

Row	Year	Total_cost_orders	Previous_Year_Cost	Percentage_change_from_2017_to_2018_in_cost_orders
1	2017	3610270.15	*null*	*null*
2	2018	8643531.14	3610270.15	139.4150792289048

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

```
SELECT customer_state,AVG(price) as avg_price,AVG(freight_value)
as avg_freight_value,
SUM(price) as total_price,SUM(freight_value) as
total_freight_value
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id
group by customer_state
```

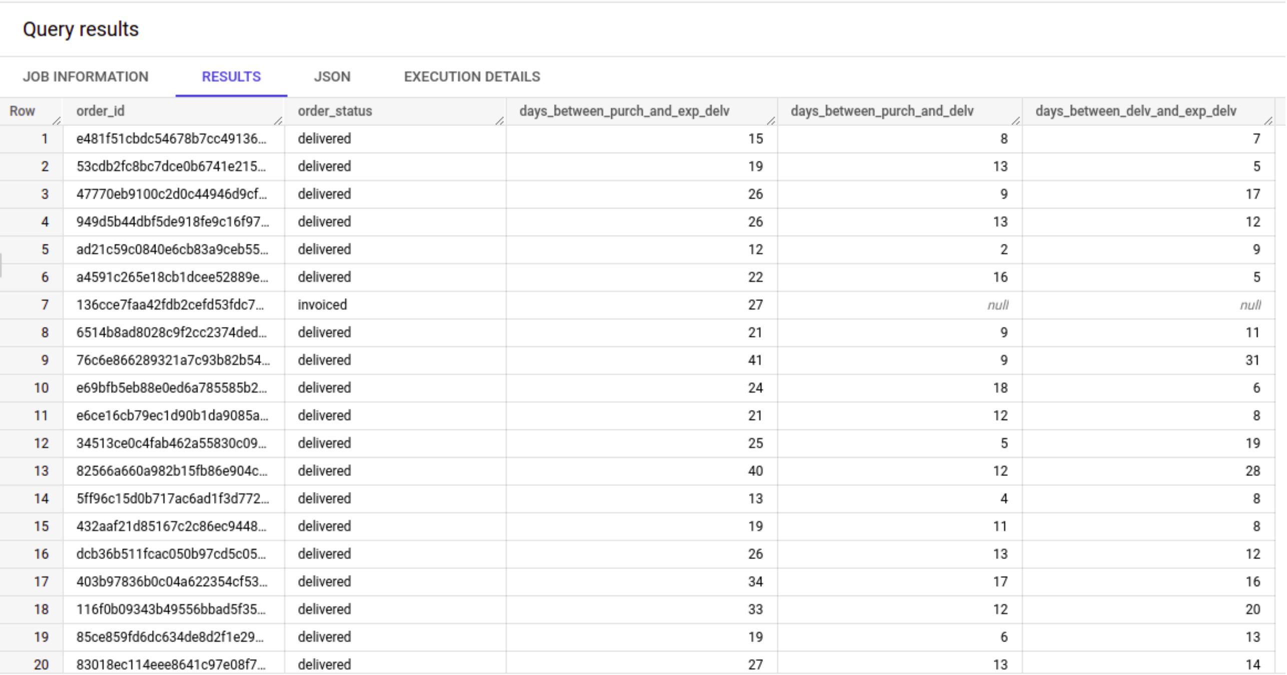


Query results

Row	customer_state	avg_price	avg_freight_value	total_price	total_freight_value
1	SP	109.653629...	73	5202955.050001...	718723.06999998...
2	BA	134.601208...	26.363958936562298	511349.9900000...	100156.67999999...
3	GO	126.271731...	22.766815259322779	294591.9499999...	53114.979999999...
4	RN	156.965935...	35.652362948960246	83034.98	18860.1
5	PR	119.004139...	20.531651567944252	683083.7600000...	117851.68000000...
6	RS	120.337453...	21.735804330392906	750304.0200000...	135522.74000000...
7	RJ	125.117818...	20.960923931682583	1824092.669999...	305589.31000000...
8	MG	120.748574...	20.630166806306676	1585308.029999...	270853.46000000...
9	SC	124.653577...	21.470368773946316	520553.3400000...	89660.260000000...
10	RR	150.565961...	42.984423076923086	7829.429999999...	2235.1900000000...
11	PE	145.508322...	32.917862679955682	262788.0299999...	59449.659999999...
12	TO	157.529333...	37.2466031746032	49621.74000000...	11732.680000000...
13	CE	153.758261...	32.71420162381591	227254.7099999...	48351.58999999988
14	DF	125.770548...	21.041354945968422	302603.9399999...	50625.499999999...
15	SE	153.041168...	36.653168831168841	58920.85000000...	14111.469999999...
16	MT	148.297184...	28.166284360189561	156453.5299999...	29715.430000000...
17	PB	191.475215...	42.723803986710955	115268.0799999...	25719.730000000...
18	PA	165.692416...	35.832685185185092	178947.8099999...	38699.3
19	RO	165.973525...	41.069712230215828	46140.64000000...	11417.379999999...
20	ES	121.913701...	22.058776595744693	275037.3099999...	49764.599999999...

5.1. Calculate days between purchasing, delivering and estimated delivery

```
SELECT order_id,order_status,
DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp,
day) as days_between_purch_and_exp_delv,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as days_between_purch_and_delv,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate, day) as days_between_delv_and_exp_delv
FROM `big-query-358521.Target.orders` LIMIT 20
```

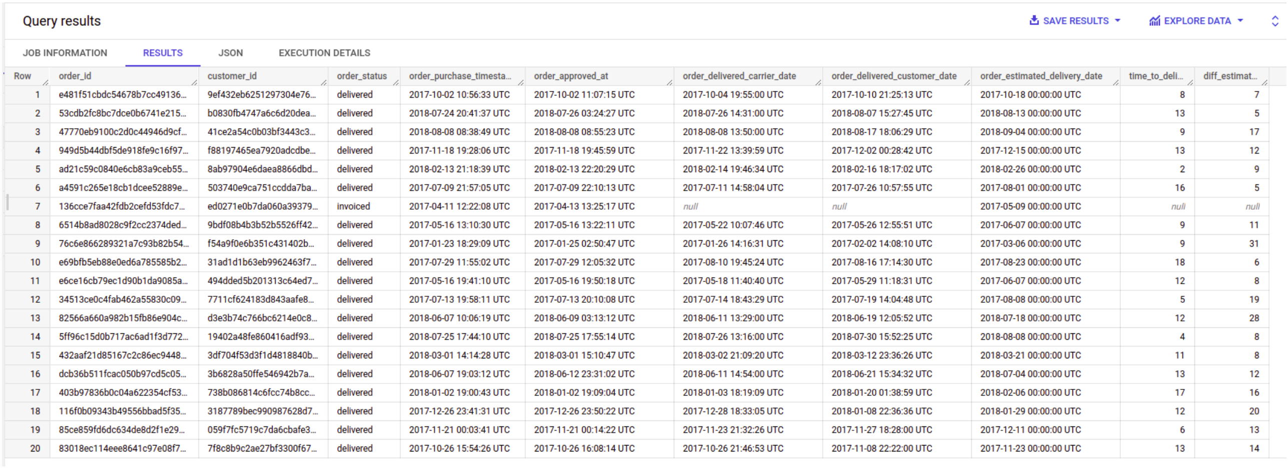


Row	order_id	order_status	days_between_purch_and_exp_delv	days_between_purch_and_delv	days_between_delv_and_exp_delv
1	e481f51cbdc54678b7cc49136...	delivered	15	8	7
2	53cdb2fc8bc7dce0b6741e215...	delivered	19	13	5
3	47770eb9100c2d0c44946d9cf...	delivered	26	9	17
4	949d5b44dbf5de918fe9c16f97...	delivered	26	13	12
5	ad21c59c0840e6cb83a9ceb55...	delivered	12	2	9
6	a4591c265e18cb1dcee52889e...	delivered	22	16	5
7	136cce7faa42fdb2cefd53fdc7...	invoiced	27	*null*	*null*
8	6514b8ad8028c9f2cc2374ded...	delivered	21	9	11
9	76c6e866289321a7c93b82b54...	delivered	41	9	31
10	e69bfb5eb88e0ed6a785585b2...	delivered	24	18	6
11	e6ce16cb79ec1d90b1da9085a...	delivered	21	12	8
12	34513ce0c4fab462a55830c09...	delivered	25	5	19
13	82566a660a982b15fb86e904c...	delivered	40	12	28
14	5ff96c15d0b717ac6ad1f3d772...	delivered	13	4	8
15	432aaf21d85167c2c86ec9448...	delivered	19	11	8
16	dcb36b511fcac050b97cd5c05...	delivered	26	13	12
17	403b97836b0c04a622354cf53...	delivered	34	17	16
18	116f0b09343b49556bbad5f35...	delivered	33	12	20
19	85ce859fd6dc634de8d2f1e29...	delivered	19	6	13
20	83018ec114eee8641c97e08f7...	delivered	27	13	14

5.2. Create columns:

- time_to_delivery = order_purchase_timestamp-order_delivered_customer_date
- diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date

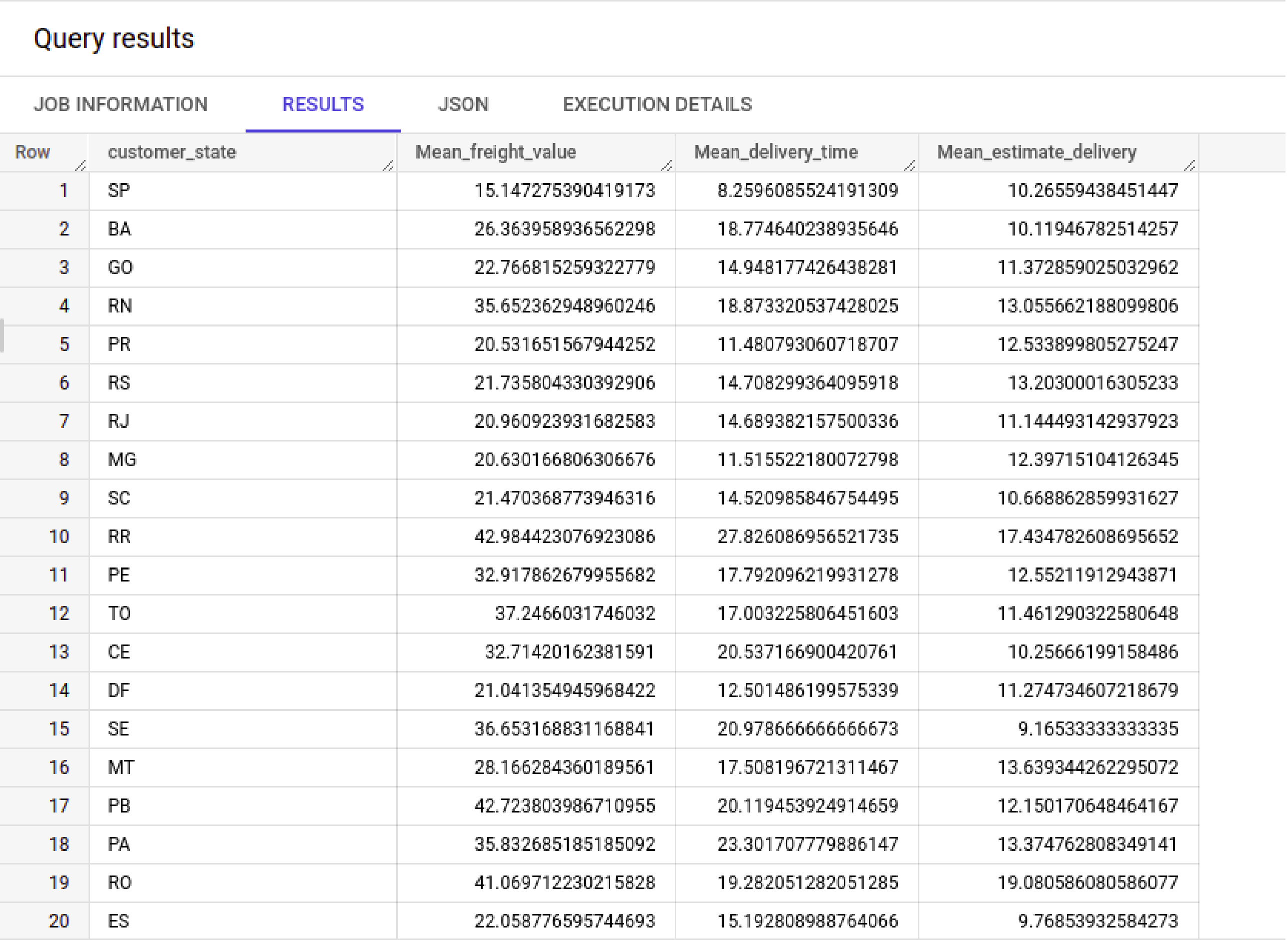
```
SELECT *,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate,day) as diff_estimated_delivery
FROM `big-query-358521.Target.orders` LIMIT 20
```



Row	order_id	customer_id	order_status	order_purchase_timesta...	order_approved_at	order_delivered_carrier_date	order_delivered_customer_date	order_estimated_delivery_date	time_to_deli...	diff_estimat...
1	e481f51cbdc54678b7cc49136...	9ef432eb6251297304e76...	delivered	2017-10-02 10:56:33 UTC	2017-10-02 11:07:15 UTC	2017-10-04 19:55:00 UTC	2017-10-10 21:25:13 UTC	2017-10-18 00:00:00 UTC	8	7
2	53cdb2fc8bc7dce0b6741e215...	b0830fb4747a6c6d20dea...	delivered	2018-07-24 20:41:37 UTC	2018-07-26 03:24:27 UTC	2018-07-26 14:31:00 UTC	2018-08-07 15:27:45 UTC	2018-08-13 00:00:00 UTC	13	5
3	47770eb9100c2d0c44946d9cf...	41ce2a54c0b03bf3443c3...	delivered	2018-08-08 08:38:49 UTC	2018-08-08 08:55:23 UTC	2018-08-08 13:50:00 UTC	2018-08-17 18:06:29 UTC	2018-09-04 00:00:00 UTC	9	17
4	949d5b44dbf5de918fe9c16f97...	f88197465ea7920adcdbe...	delivered	2017-11-18 19:28:06 UTC	2017-11-18 19:45:59 UTC	2017-11-22 13:39:59 UTC	2017-12-02 00:28:42 UTC	2017-12-15 00:00:00 UTC	13	12
5	ad21c59c0840e6cb83a9ceb55...	8ab97904e6daea8866dbd...	delivered	2018-02-13 21:18:39 UTC	2018-02-13 22:20:29 UTC	2018-02-14 19:46:34 UTC	2018-02-16 18:17:02 UTC	2018-02-26 00:00:00 UTC	2	9
6	a4591c265e18cb1dcee52889e...	503740e9ca751ccdda7ba...	delivered	2017-07-09 21:57:05 UTC	2017-07-09 22:10:13 UTC	2017-07-11 14:58:04 UTC	2017-07-26 10:57:55 UTC	2017-08-01 00:00:00 UTC	16	5
7	136cce7faa42fdb2cefd53fdc7...	ed0271e0b7da060a39379...	invoiced	2017-04-11 12:22:08 UTC	2017-04-13 13:25:17 UTC	*null*	*null*	2017-05-09 00:00:00 UTC	*null*	*null*
8	6514b8ad8028c9f2cc2374ded...	9bdf08b4b3b52b5526ff42...	delivered	2017-05-16 13:10:30 UTC	2017-05-16 13:22:11 UTC	2017-05-22 10:07:46 UTC	2017-05-26 12:55:51 UTC	2017-06-07 00:00:00 UTC	9	11
9	76c6e866289321a7c93b82b54...	f54a9f0e6b351c431402b...	delivered	2017-01-23 18:29:09 UTC	2017-01-25 02:50:47 UTC	2017-01-26 14:16:31 UTC	2017-02-02 14:08:10 UTC	2017-03-06 00:00:00 UTC	9	31
10	e69bfb5eb88e0ed6a785585b2...	31ad1d1b63eb9962463f7...	delivered	2017-07-29 11:55:02 UTC	2017-07-29 12:05:32 UTC	2017-08-10 19:45:24 UTC	2017-08-16 17:14:30 UTC	2017-08-23 00:00:00 UTC	18	6
11	e6ce16cb79ec1d90b1da9085a...	494dded5b201313c64ed7...	delivered	2017-05-16 19:41:10 UTC	2017-05-16 19:50:18 UTC	2017-05-18 11:40:40 UTC	2017-05-29 11:18:31 UTC	2017-06-07 00:00:00 UTC	12	8
12	34513ce0c4fab462a55830c09...	7711cf624183d843aafe8...	delivered	2017-07-13 19:58:11 UTC	2017-07-13 20:10:08 UTC	2017-07-14 18:43:29 UTC	2017-07-19 14:04:48 UTC	2017-08-08 00:00:00 UTC	5	19
13	82566a660a982b15fb86e904c...	d3e3b74c766bc6214e0c8...	delivered	2018-06-07 10:06:19 UTC	2018-06-09 03:13:12 UTC	2018-06-11 13:29:00 UTC	2018-06-19 12:05:52 UTC	2018-07-18 00:00:00 UTC	12	28
14	5ff96c15d0b717ac6ad1f3d772...	19402a48fe860416adf93...	delivered	2018-07-25 17:44:10 UTC	2018-07-25 17:55:14 UTC	2018-07-26 13:16:00 UTC	2018-07-30 15:52:25 UTC	2018-08-08 00:00:00 UTC	4	8
15	432aaf21d85167c2c86ec9448...	3df704f53d3f1d4818840b...	delivered	2018-03-01 14:14:28 UTC	2018-03-01 15:10:47 UTC	2018-03-02 21:09:20 UTC	2018-03-12 23:36:26 UTC	2018-03-21 00:00:00 UTC	11	8
16	dcb36b511fcac050b97cd5c05...	3b6828a50ffe546942b7a...	delivered	2018-06-07 19:03:12 UTC	2018-06-12 23:31:02 UTC	2018-06-11 14:54:00 UTC	2018-06-21 15:34:32 UTC	2018-07-04 00:00:00 UTC	13	12
17	403b97836b0c04a622354cf53...	738b086814c6fcc74b8cc...	delivered	2018-01-02 19:00:43 UTC	2018-01-02 19:09:04 UTC	2018-01-03 18:19:09 UTC	2018-01-20 01:38:59 UTC	2018-02-06 00:00:00 UTC	17	16
18	116f0b09343b49556bbad5f35...	3187789bec990987628d7...	delivered	2017-12-26 23:41:31 UTC	2017-12-26 23:50:22 UTC	2017-12-28 18:33:05 UTC	2018-01-08 22:36:36 UTC	2018-01-29 00:00:00 UTC	12	20
19	85ce859fd6dc634de8d2f1e29...	059f7fc5719c7da6cbafe3...	delivered	2017-11-21 00:03:41 UTC	2017-11-21 00:14:22 UTC	2017-11-23 21:32:26 UTC	2017-11-27 18:28:00 UTC	2017-12-11 00:00:00 UTC	6	13
20	83018ec114eee8641c97e08f7...	7f8c8b9c2ae27bf3300f67...	delivered	2017-10-26 15:54:26 UTC	2017-10-26 16:08:14 UTC	2017-10-26 21:46:53 UTC	2017-11-08 22:22:00 UTC	2017-11-23 00:00:00 UTC	13	14

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

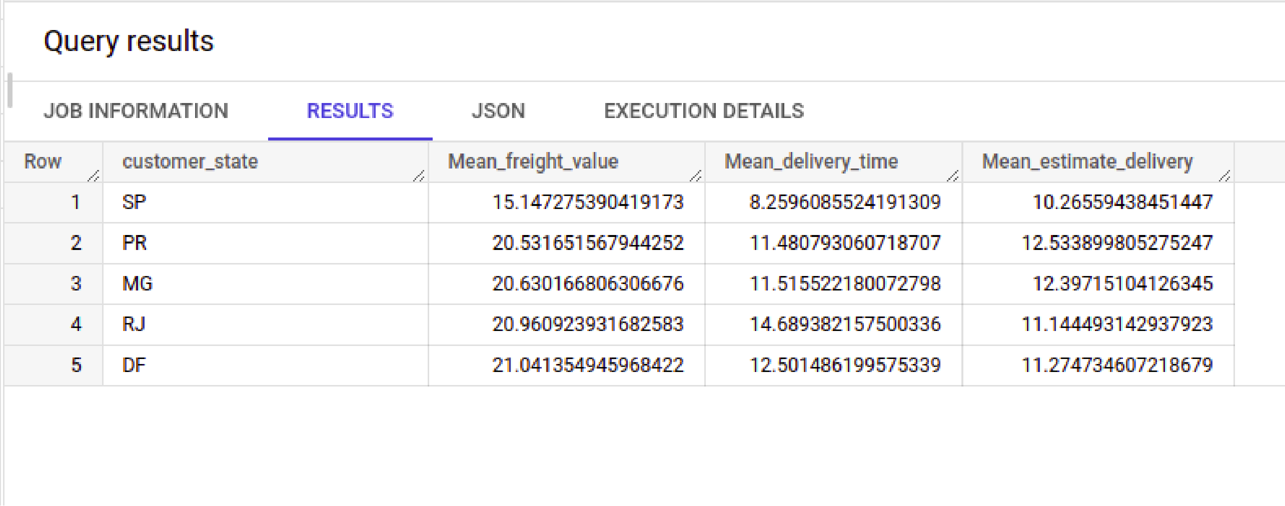
```
SELECT customer_state,AVG(freight_value) as
Mean_freight_value,AVG(time_to_delivery) as
Mean_delivery_time,AVG(diff_estimated_delivery) as
Mean_estimate_delivery
FROM(SELECT
o.order_id,o.customer_id,order_status,customer_city,customer_state
,price,freight_value,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate,day) as diff_estimated_delivery
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id) x
group by customer_state
```



Row	customer_state	Mean_freight_value	Mean_delivery_time	Mean_estimate_delivery
1	SP	15.147275390419173	8.2596085524191309	10.26559438451447
2	BA	26.363958936562298	18.774640238935646	10.11946782514257
3	GO	22.766815259322779	14.948177426438281	11.372859025032962
4	RN	35.652362948960246	18.873320537428025	13.055662188099806
5	PR	20.531651567944252	11.480793060718707	12.533899805275247
6	RS	21.735804330392906	14.708299364095918	13.20300016305233
7	RJ	20.960923931682583	14.689382157500336	11.144493142937923
8	MG	20.630166806306676	11.515522180072798	12.39715104126345
9	SC	21.470368773946316	14.520985846754495	10.668862859931627
10	RR	42.984423076923086	27.826086956521735	17.434782608695652
11	PE	32.917862679955682	17.792096219931278	12.55211912943871
12	TO	37.2466031746032	17.003225806451603	11.461290322580648
13	CE	32.71420162381591	20.537166900420761	10.25666199158486
14	DF	21.041354945968422	12.501486199575339	11.274734607218679
15	SE	36.653168831168841	20.978666666666673	9.16533333333335
16	MT	28.166284360189561	17.508196721311467	13.639344262295072
17	PB	42.723803986710955	20.119453924914659	12.150170648464167
18	PA	35.832685185185092	23.301707779886147	13.374762808349141
19	RO	41.069712230215828	19.282051282051285	19.080586080586077
20	ES	22.058776595744693	15.192808988764066	9.76853932584273

5.4. Sort the data to get the following:

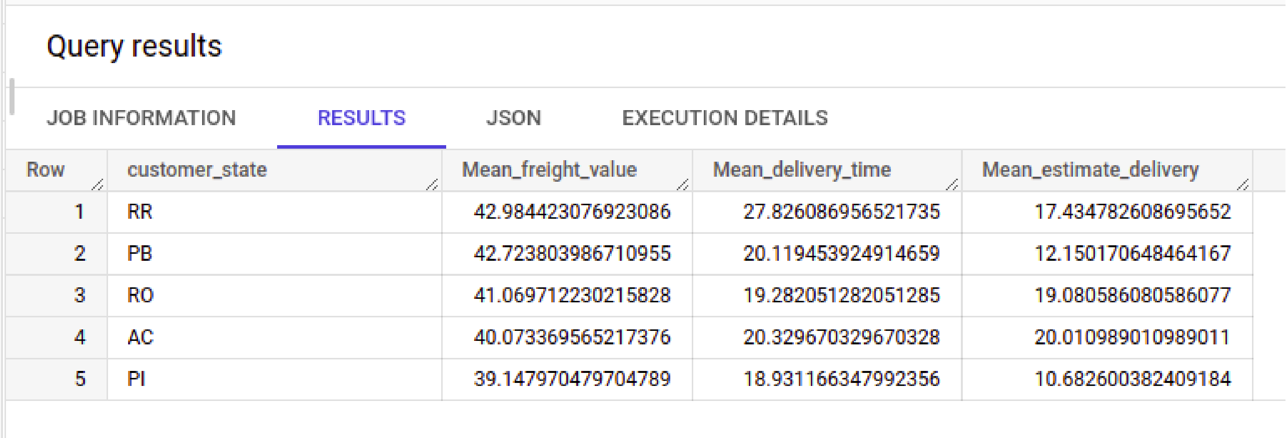
```
#Top 5 states with lowest average freight value - sort in
desc/asc limit 5

SELECT customer_state,AVG(freight_value) as
Mean_freight_value,AVG(time_to_delivery) as
Mean_delivery_time,AVG(diff_estimated_delivery) as
Mean_estimate_delivery
FROM(SELECT
o.order_id,o.customer_id,order_status,customer_city,customer_state
,price,freight_value,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate,day) as diff_estimated_delivery
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id) x
group by customer_state
Order by Mean_freight_value
LIMIT 5
```



Query results

Row	customer_state	Mean_freight_value	Mean_delivery_time	Mean_estimate_delivery
1	SP	15.147275390419173	8.2596085524191309	10.26559438451447
2	PR	20.531651567944252	11.480793060718707	12.533899805275247
3	MG	20.630166806306676	11.515522180072798	12.39715104126345
4	RJ	20.960923931682583	14.689382157500336	11.144493142937923
5	DF	21.041354945968422	12.501486199575339	11.274734607218679

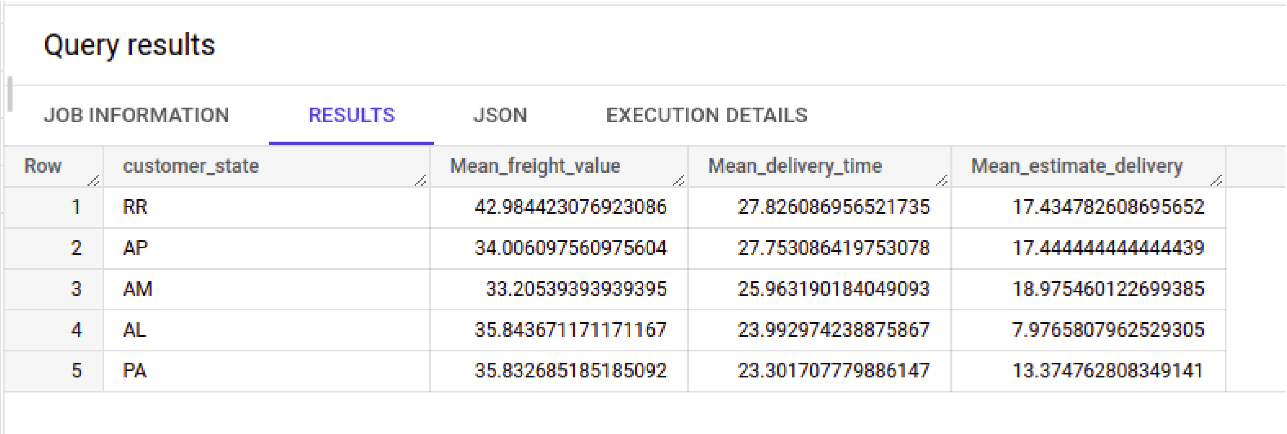
```
#Top 5 states with highest average freight value - sort
in desc/asc limit 5

SELECT customer_state,AVG(freight_value) as
Mean_freight_value,AVG(time_to_delivery) as
Mean_delivery_time,AVG(diff_estimated_delivery) as
Mean_estimate_delivery
FROM(SELECT
o.order_id,o.customer_id,order_status,customer_city,customer_state
,price,freight_value,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate,day) as diff_estimated_delivery
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id) x
group by customer_state
Order by Mean_freight_value DESC
LIMIT 5
```



Query results

Row	customer_state	Mean_freight_value	Mean_delivery_time	Mean_estimate_delivery
1	RR	42.984423076923086	27.826086956521735	17.434782608695652
2	PB	42.723803986710955	20.119453924914659	12.150170648464167
3	RO	41.069712230215828	19.282051282051285	19.080586080586077
4	AC	40.073369565217376	20.329670329670328	20.010989010989011
5	PI	39.147970479704789	18.931166347992356	10.682600382409184

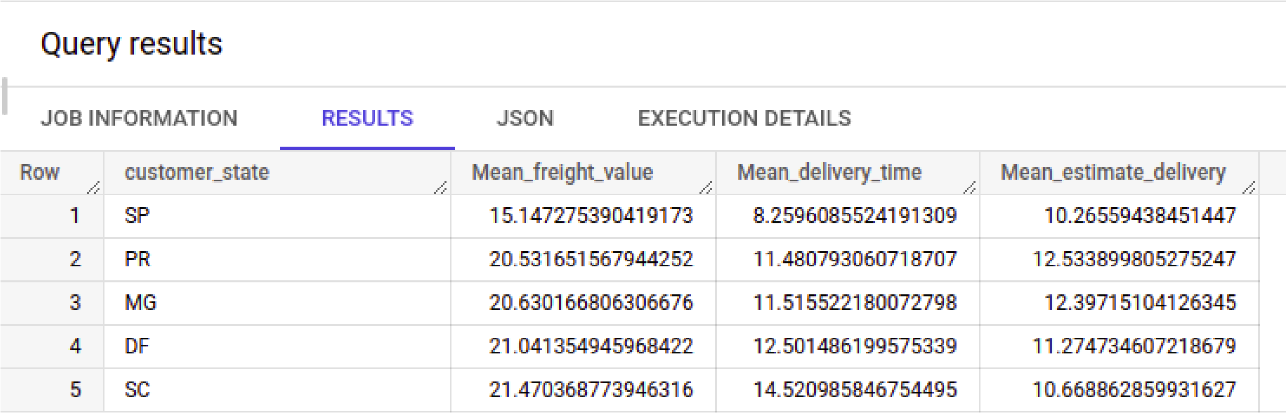
```
#Top 5 states with highest average time to delivery

SELECT customer_state,AVG(freight_value) as
Mean_freight_value,AVG(time_to_delivery) as
Mean_delivery_time,AVG(diff_estimated_delivery) as
Mean_estimate_delivery
FROM(SELECT
o.order_id,o.customer_id,order_status,customer_city,customer_state
,price,freight_value,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate,day) as diff_estimated_delivery
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id) x
group by customer_state
Order by Mean_delivery_time DESC
LIMIT 5
```



Query results

Row	customer_state	Mean_freight_value	Mean_delivery_time	Mean_estimate_delivery
1	RR	42.984423076923086	27.826086956521735	17.434782608695652
2	AP	34.006097560975604	27.753086419753078	17.444444444444439
3	AM	33.20539393939395	25.963190184049093	18.975460122699385
4	AL	35.843671171171167	23.992974238875867	7.9765807962529305
5	PA	35.832685185185092	23.301707779886147	13.374762808349141

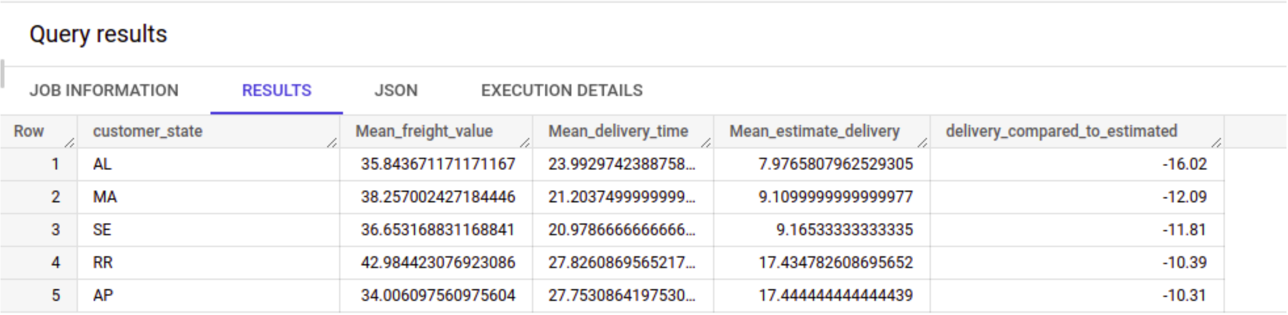
```
#Top 5 states with lowest average time to delivery

SELECT customer_state,AVG(freight_value) as
Mean_freight_value,AVG(time_to_delivery) as
Mean_delivery_time,AVG(diff_estimated_delivery) as
Mean_estimate_delivery
FROM(SELECT
o.order_id,o.customer_id,order_status,customer_city,customer_state
,price,freight_value,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate,day) as diff_estimated_delivery
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id) x
group by customer_state
Order by Mean_delivery_time
LIMIT 5
```


Query results

Row	customer_state	Mean_freight_value	Mean_delivery_time	Mean_estimate_delivery
1	SP	15.147275390419173	8.2596085524191309	10.26559438451447
2	PR	20.531651567944252	11.480793060718707	12.533899805275247
3	MG	20.630166806306676	11.515522180072798	12.39715104126345
4	DF	21.041354945968422	12.501486199575339	11.274734607218679
5	SC	21.470368773946316	14.520985846754495	10.668862859931627

```
#Top 5 states where delivery is really not so fast
compared to estimated date

SELECT customer_state,AVG(freight_value) as
Mean_freight_value,AVG(time_to_delivery) as
Mean_delivery_time,AVG(diff_estimated_delivery) as
Mean_estimate_delivery,
round((AVG(diff_estimated_delivery)-AVG(time_to_delivery)),2) as
delivery_compared_to_estimated
```

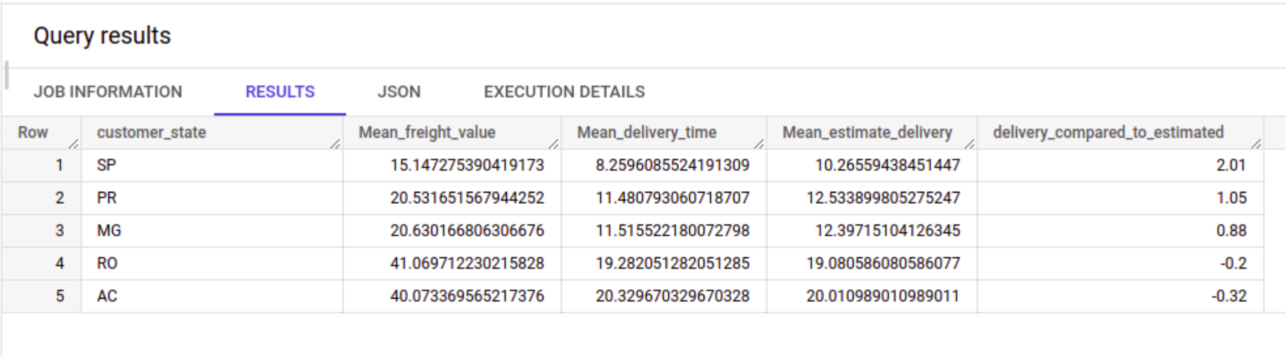
```
FROM(SELECT
o.order_id,o.customer_id,order_status,customer_city,customer_state
,price,freight_value,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate,day) as diff_estimated_delivery
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id) x
group by customer_state
Order by delivery_compared_to_estimated
LIMIT 5
```



Query results

Row	customer_state	Mean_freight_value	Mean_delivery_time	Mean_estimate_delivery	delivery_compared_to_estimated
1	AL	35.843671171171167	23.9929742388758...	7.9765807962529305	-16.02
2	MA	38.257002427184446	21.2037499999999...	9.1099999999999977	-12.09
3	SE	36.653168831168841	20.9786666666666...	9.16533333333335	-11.81
4	RR	42.984423076923086	27.8260869565217...	17.434782608695652	-10.39
5	AP	34.006097560975604	27.7530864197530...	17.444444444444439	-10.31

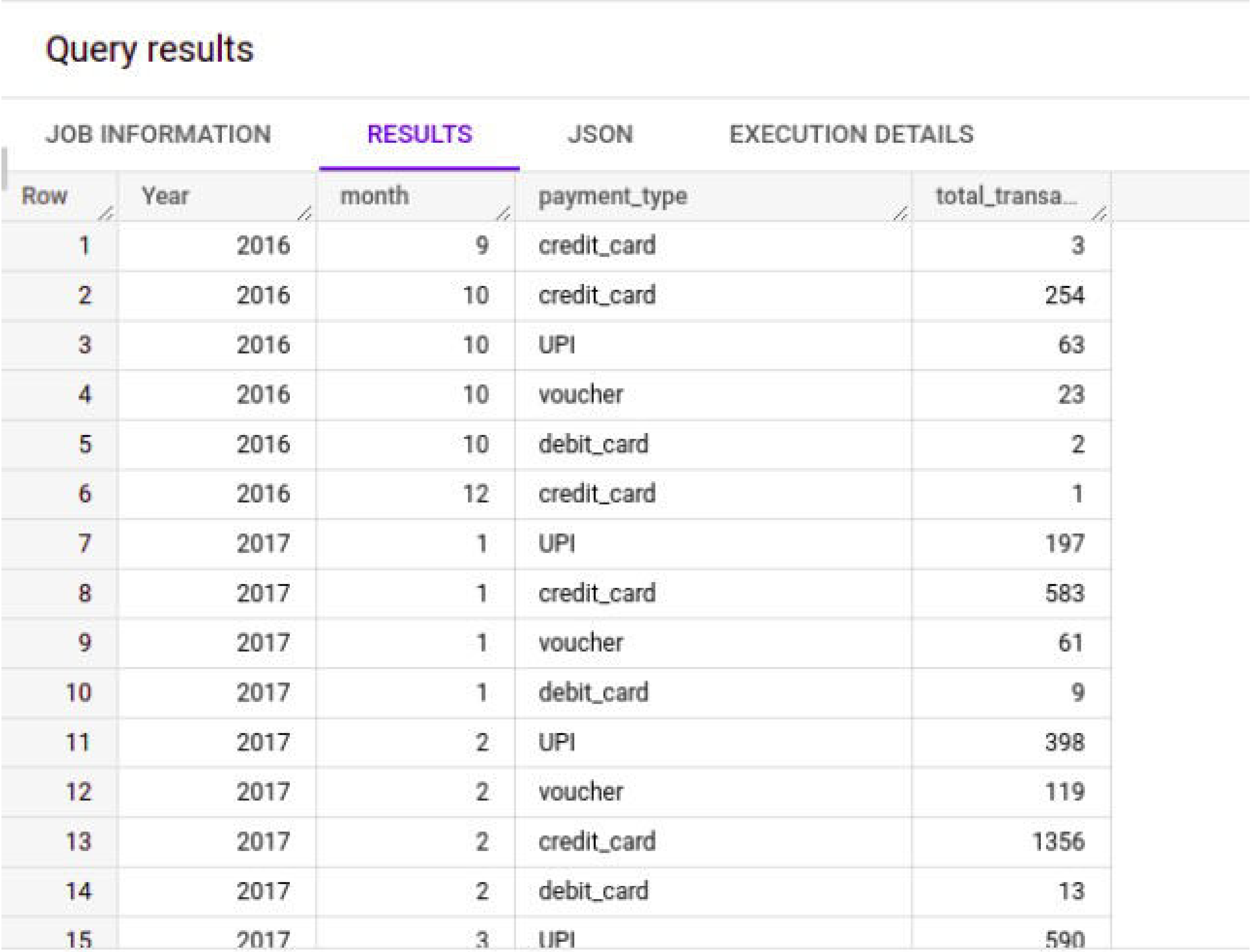
```
#Top 5 states where delivery is really  fast compared to
estimated date

SELECT customer_state,AVG(freight_value) as
Mean_freight_value,AVG(time_to_delivery) as
Mean_delivery_time,AVG(diff_estimated_delivery) as
Mean_estimate_delivery,
round((AVG(diff_estimated_delivery)-AVG(time_to_delivery)),2) as
delivery_compared_to_estimated
FROM(SELECT
o.order_id,o.customer_id,order_status,customer_city,customer_state
,price,freight_value,
DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,
day) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_d
ate,day) as diff_estimated_delivery
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.order_items` oi
On o.order_id=oi.order_id) x
group by customer_state
Order by delivery_compared_to_estimated DESC
LIMIT 5
```


Query results

Row	customer_state	Mean_freight_value	Mean_delivery_time	Mean_estimate_delivery	delivery_compared_to_estimated
1	SP	15.147275390419173	8.2596085524191309	10.26559438451447	2.01
2	PR	20.531651567944252	11.480793060718707	12.533899805275247	1.05
3	MG	20.630166806306676	11.515522180072798	12.39715104126345	0.88
4	RO	41.069712230215828	19.282051282051285	19.080586080586077	-0.2
5	AC	40.073369565217376	20.329670329670328	20.010989010989011	-0.32

6.1. Month over Month count of orders for different payment types

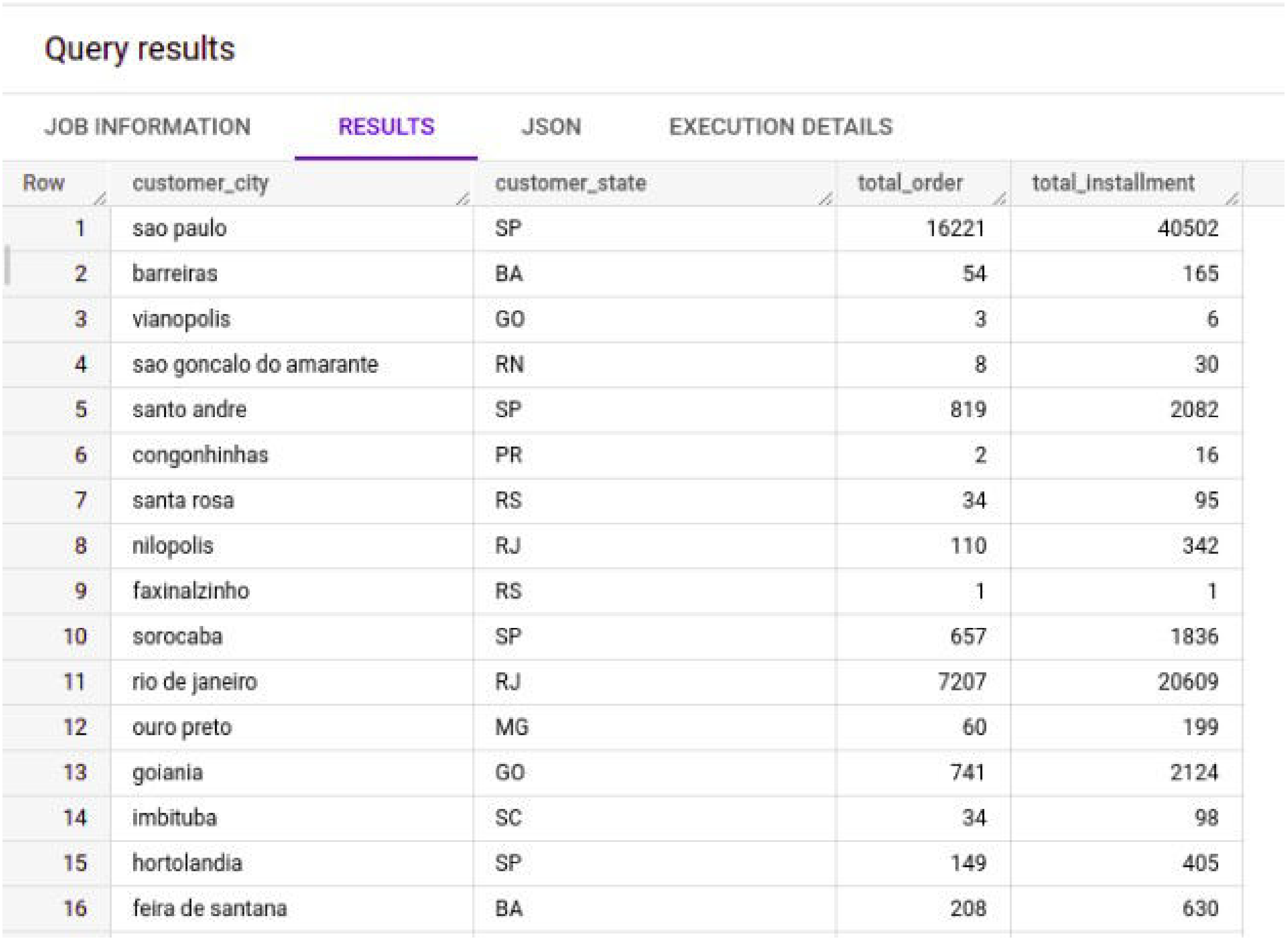
```
SELECT Year,month,payment_type,count(payment_type) as
total_transaction
FROM
(SELECT
o.order_id,o.customer_id,order_status,customer_city,customer_state
,payment_type,payment_installments, payment_value,
EXTRACT(Year FROM order_purchase_timestamp) as Year,
EXTRACT(Month FROM order_purchase_timestamp) as Month,
EXTRACT(Day FROM order_purchase_timestamp) as day
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.payments` p
On o.order_id=p.order_id) x
group by Year,Month,payment_type
ORDER BY Year,Month
```



Row	Year	month	payment_type	total_transa...
1	2016	9	credit_card	3
2	2016	10	credit_card	254
3	2016	10	UPI	63
4	2016	10	voucher	23
5	2016	10	debit_card	2
6	2016	12	credit_card	1
7	2017	1	UPI	197
8	2017	1	credit_card	583
9	2017	1	voucher	61
10	2017	1	debit_card	9
11	2017	2	UPI	398
12	2017	2	voucher	119
13	2017	2	credit_card	1356
14	2017	2	debit_card	13
15	2017	3	UPI	590

6.2. Distribution of payment installments and count of orders

```
SELECT customer_city,customer_state,count(*) as
total_order,sum(payment_installments) as total_installment
FROM `big-query-358521.Target.customers` c
Join `big-query-358521.Target.orders` o
On c.customer_id=o.customer_id
Join `big-query-358521.Target.payments` p
On o.order_id=p.order_id
group by customer_state,customer_city
```



Row	customer_city	customer_state	total_order	total_installment
1	sao paulo	SP	16221	40502
2	barreiras	BA	54	165
3	vianopolis	GO	3	6
4	sao goncalo do amarante	RN	8	30
5	santo andre	SP	819	2082
6	congonhinhas	PR	2	16
7	santa rosa	RS	34	95
8	nilopolis	RJ	110	342
9	faxinalzinho	RS	1	1
10	sorocaba	SP	657	1836
11	rio de janeiro	RJ	7207	20609
12	ouro preto	MG	60	199
13	goiania	GO	741	2124
14	imbituba	SC	34	98
15	hortolandia	SP	149	405
16	feira de santana	BA	208	630