Problem Statement:
Assuming you are a data analyst/ scientist at Target, you have been assigned the task of analyzing the given dataset to extract valuable insights and provide actionable recommendations.

What does 'good' look like?

1. **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.
    2. Get the time range between which the orders were placed.
    3. Count the Cities & States of customers who ordered during the given period.
2. **In-depth Exploration:**
    1. Is there a growing trend in the no. of orders placed over the past years?
    2. Can we see some kind of monthly seasonality in terms of the no. of orders being placed?
    3. During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)
        - 0-6 hrs : Dawn
        - 7-12 hrs : Mornings
        - 13-18 hrs : Afternoon
        - 19-23 hrs : Night
3. **Evolution of E-commerce orders in the Brazil region:**
    1. Get the month on month no. of orders placed in each state.
    2. How are the customers distributed across all the states?
4. **Impact on Economy:** Analyze 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).
       You can use the "payment_value" column in the payments table to get the cost of orders.
    2. Calculate the Total & Average value of order price for each state.
    3. Calculate the Total & Average value of order freight for each state.
5. **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_delivered_customer_date - order_estimated_delivery_date

    2. Find out the top 5 states with the highest & lowest average freight value.
    3. Find out the top 5 states with the highest & lowest average delivery time.
    4. Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery.
       You can use the difference between the averages of actual & estimated delivery date to figure out how fast the delivery was for each state.

6. **Analysis based on the payments:**
    1. Find the month on month no. of orders placed using different payment types.
    2. Find the no. of orders placed on the basis of the payment installments that have been paid.

Dataset:

This particular business case focuses on the operations of Target in Brazil and provides insightful information about 100,000 orders placed between 2016 and 2018.

The dataset offers a comprehensive view of various dimensions including the order status, price, payment and freight performance, customer location, product attributes, and customer reviews.

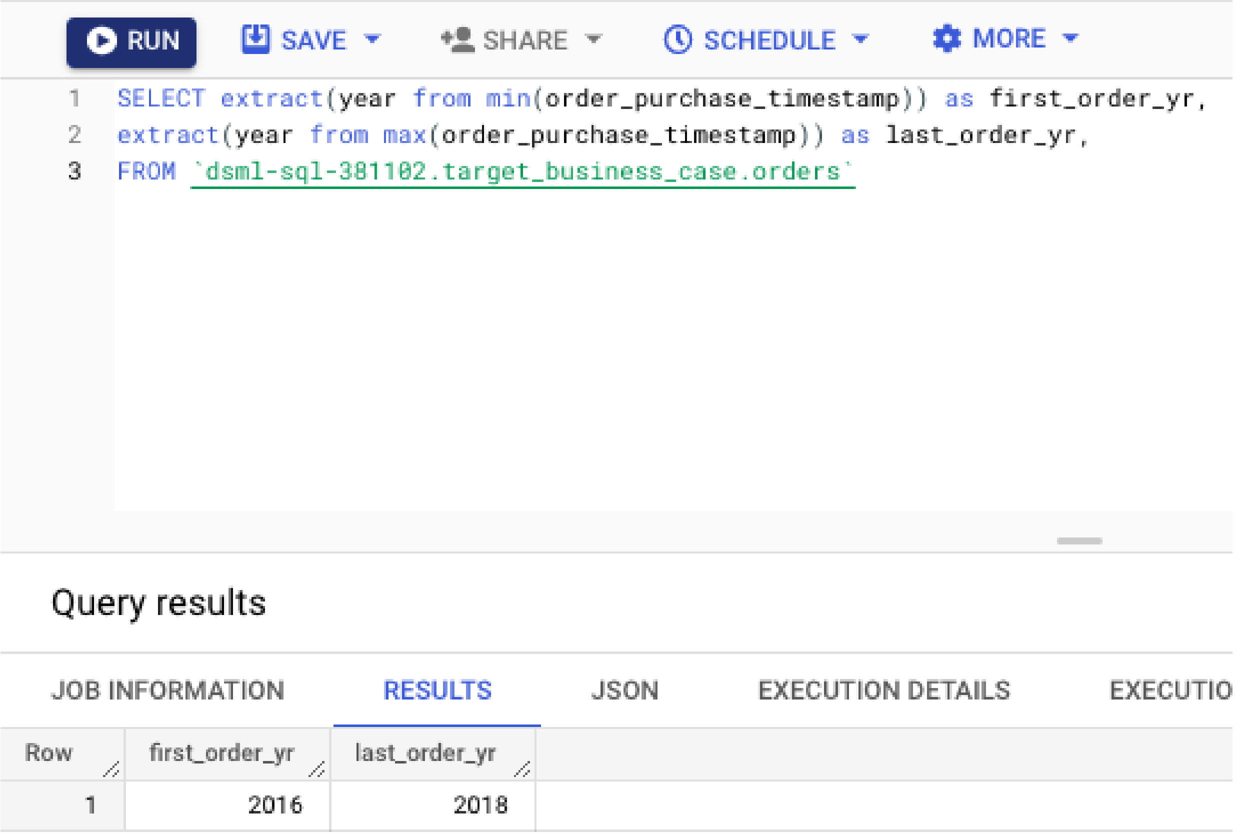
The data is available in 8 csv files:

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

What does ‘good’ look like?

1. Import the dataset and do usual **exploratory analysis** steps like checking the structure & characteristics of the dataset
    1. Data type of columns in a table
    2. Time period for which the data is given
    3. Cities and States of customers ordered during the given period
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)?
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
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
5. **Analysis on sales, freight and delivery time**
    - Calculate days between purchasing, delivering and estimated delivery
    - Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:
        a. time_to_delivery = order_purchase_timestamp-order_delivered_customer_date
        b. diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date
    - Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery
    - Sort the data to get the following:
    - Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5
    - Top 5 states with highest/lowest average time to delivery
    - Top 5 states where delivery is really fast/ not so fast compared to estimated date
6. **Payment type analysis:**
    1. Month over Month count of orders for different payment types
    2. Count of orders based on the no. of payment installments

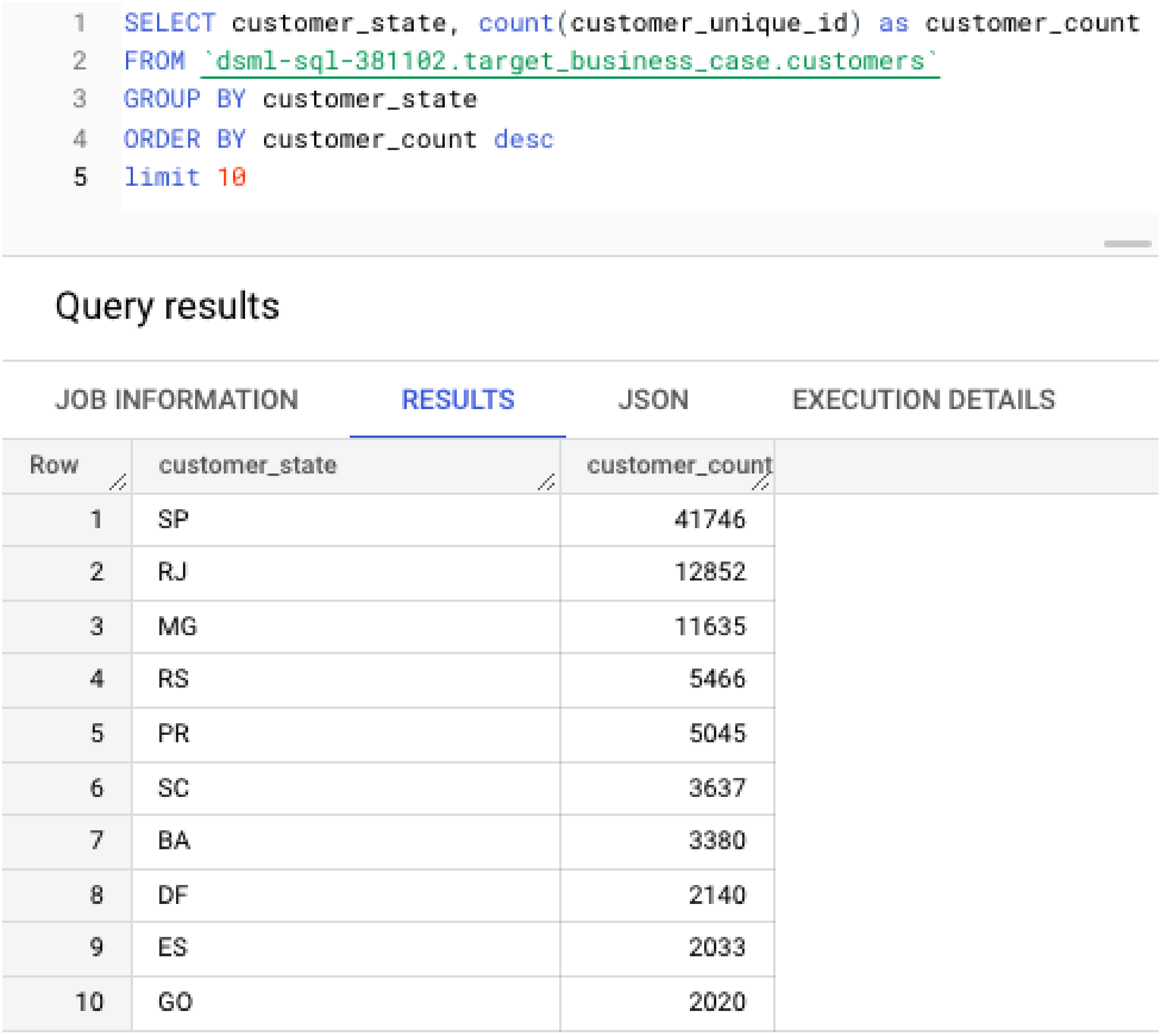
1b. Time period for which the data is given



```
SELECT extract(year from min(order_purchase_timestamp)) as first_order_yr,
extract(year from max(order_purchase_timestamp)) as last_order_yr,
FROM `dsml-sql-381102.target_business_case.orders`
```

Row	first_order_yr	last_order_yr
1	2016	2018

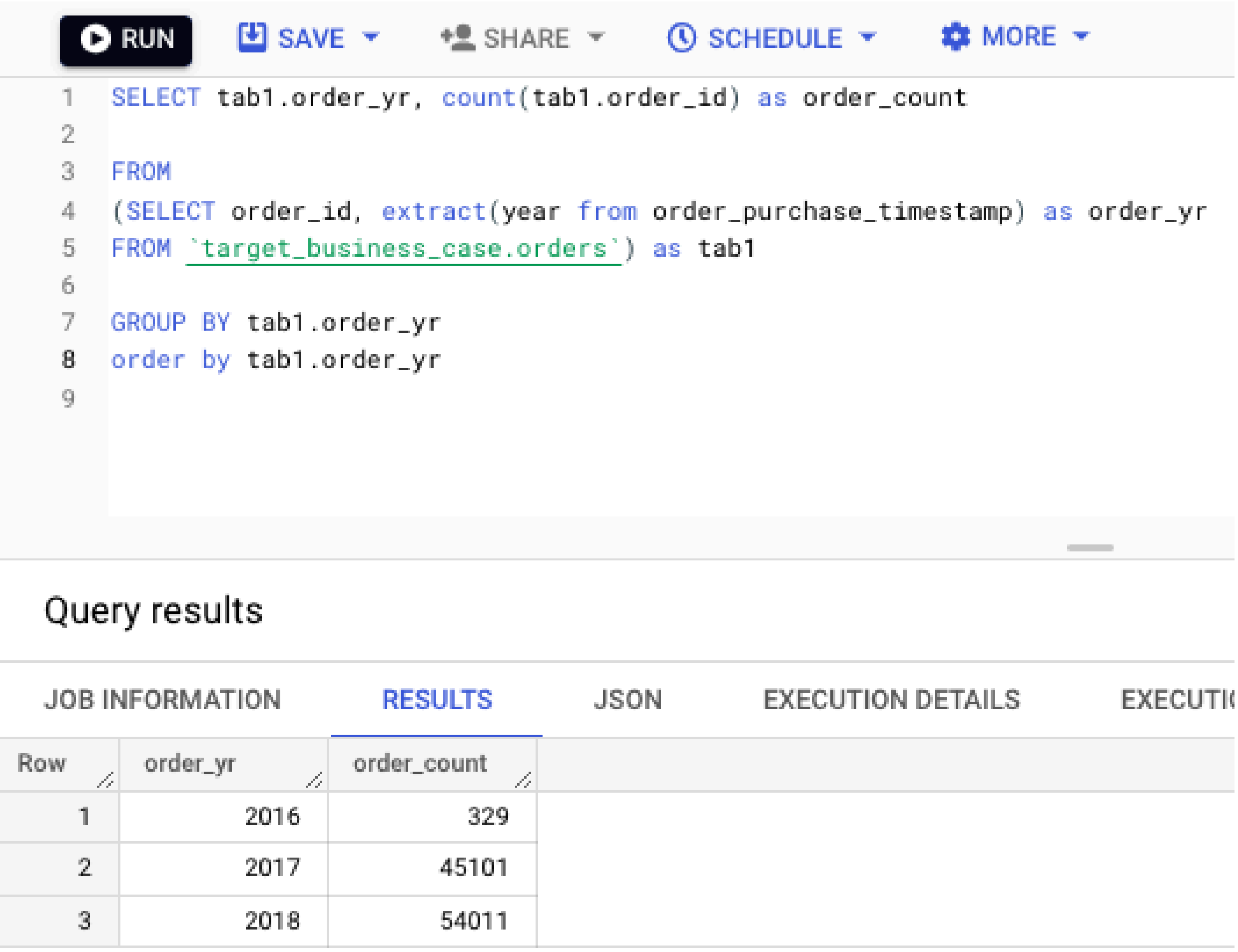
1c. Cities and States of customers ordered during the given period



```sql
SELECT customer_state, count(customer_unique_id) as customer_count
FROM `dsml-sql-381102.target_business_case.customers`
GROUP BY customer_state
ORDER BY customer_count desc
limit 10
```

Row	customer_state	customer_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

2a. Is there a growing trend on e-commerce in Brazil? - year wise order count is increasing from 2016 to 2018

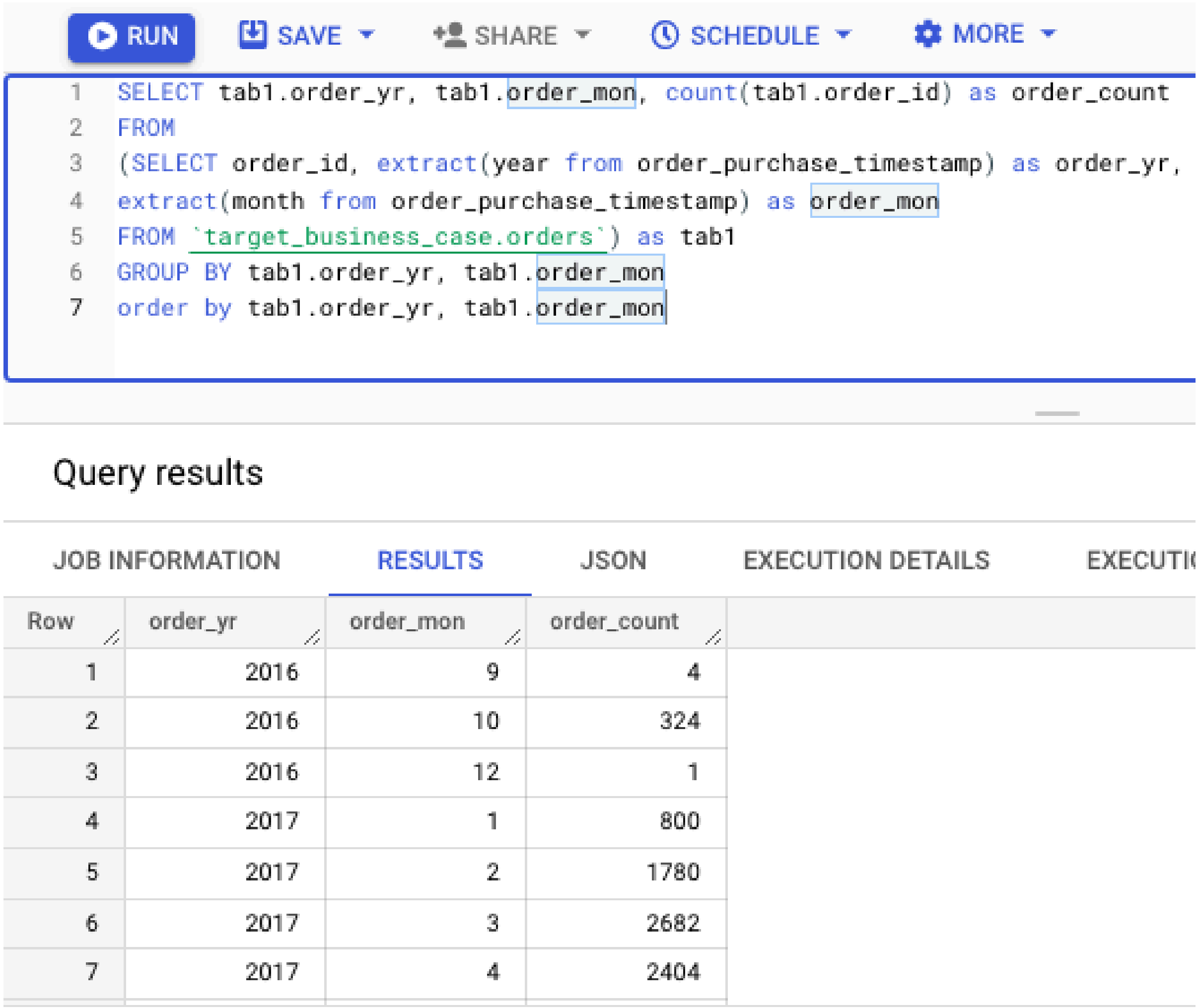


```sql
SELECT tab1.order_yr, count(tab1.order_id) as order_count

FROM
(SELECT order_id, extract(year from order_purchase_timestamp) as order_yr
FROM `target_business_case.orders`) as tab1

GROUP BY tab1.order_yr
order by tab1.order_yr

```

Query results

Row	order_yr	order_count
1	2016	329
2	2017	45101
3	2018	54011

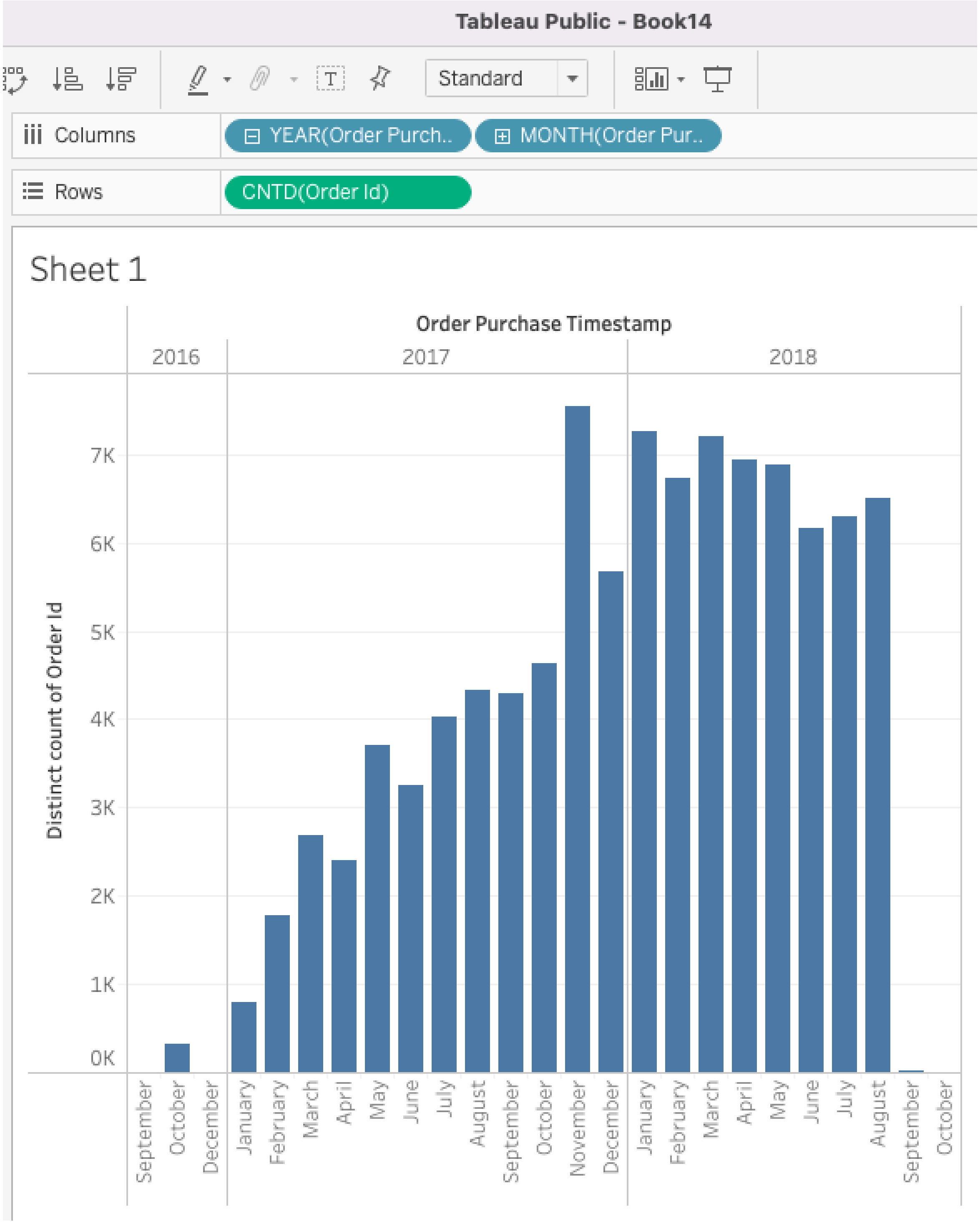
2c. We don’t see any seasonality yet. Find below the month wise order count from 2016 to 2018.



```
SELECT tab1.order_yr, tab1.order_mon, count(tab1.order_id) as order_count
FROM
(SELECT order_id, extract(year from order_purchase_timestamp) as order_yr,
extract(month from order_purchase_timestamp) as order_mon
FROM `target_business_case.orders`) as tab1
GROUP BY tab1.order_yr, tab1.order_mon
order by tab1.order_yr, tab1.order_mon
```

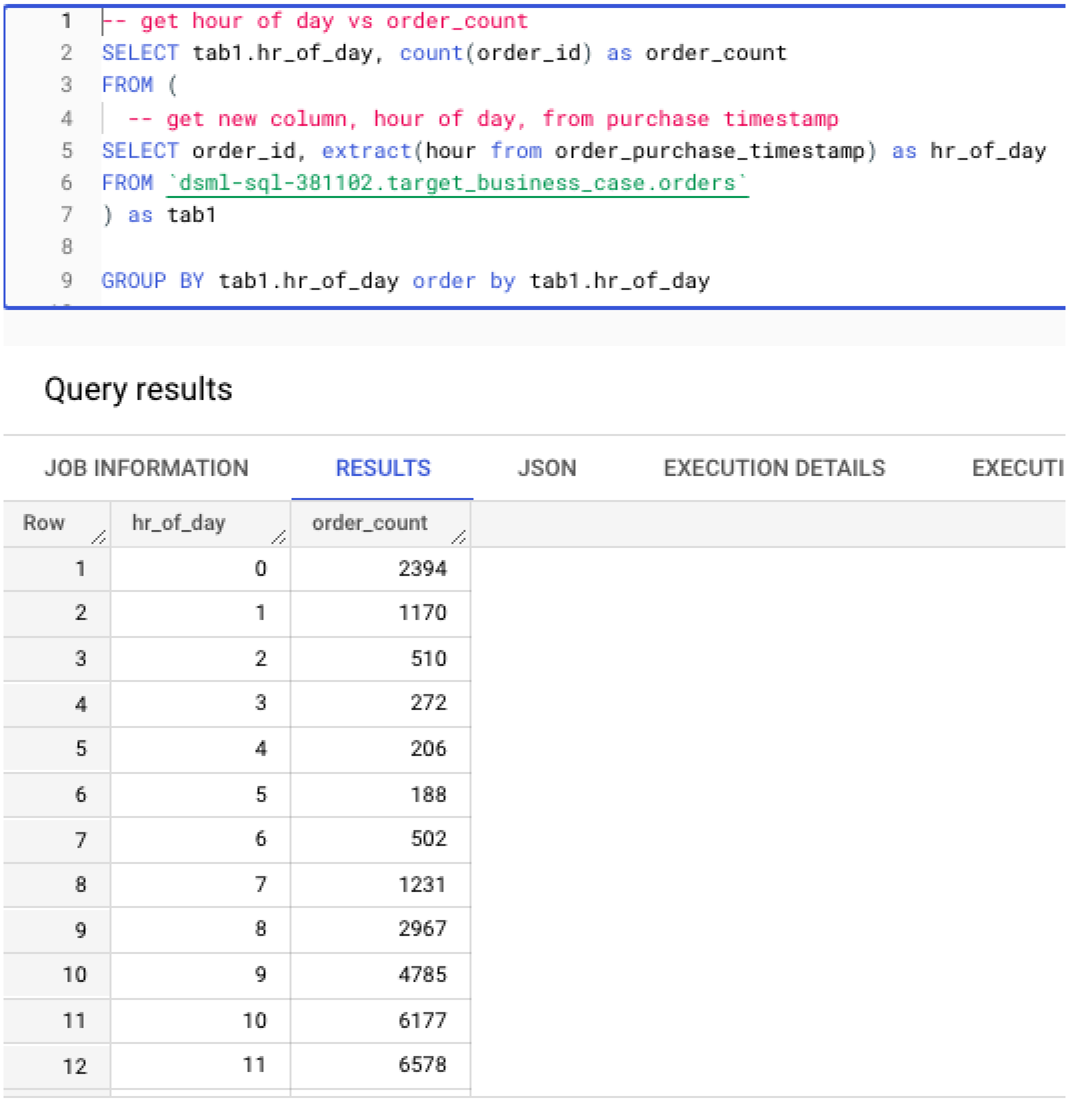
Query results

Row	order_yr	order_mon	order_count
1	2016	9	4
2	2016	10	324
3	2016	12	1
4	2017	1	800
5	2017	2	1780
6	2017	3	2682
7	2017	4	2404



2b. What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

Ans. During afternoon, from 1 pm to 5 pm, there were max no.of orders.



```sql
-- get hour of day vs order_count
SELECT tab1.hr_of_day, count(order_id) as order_count
FROM (
  -- get new column, hour of day, from purchase timestamp
SELECT order_id, extract(hour from order_purchase_timestamp) as hr_of_day
FROM `dsml-sql-381102.target_business_case.orders`
) as tab1

GROUP BY tab1.hr_of_day order by tab1.hr_of_day
```

Query results

Row	hr_of_day	order_count
1	0	2394
2	1	1170
3	2	510
4	3	272
5	4	206
6	5	188
7	6	502
8	7	1231
9	8	2967
10	9	4785
11	10	6177
12	11	6578

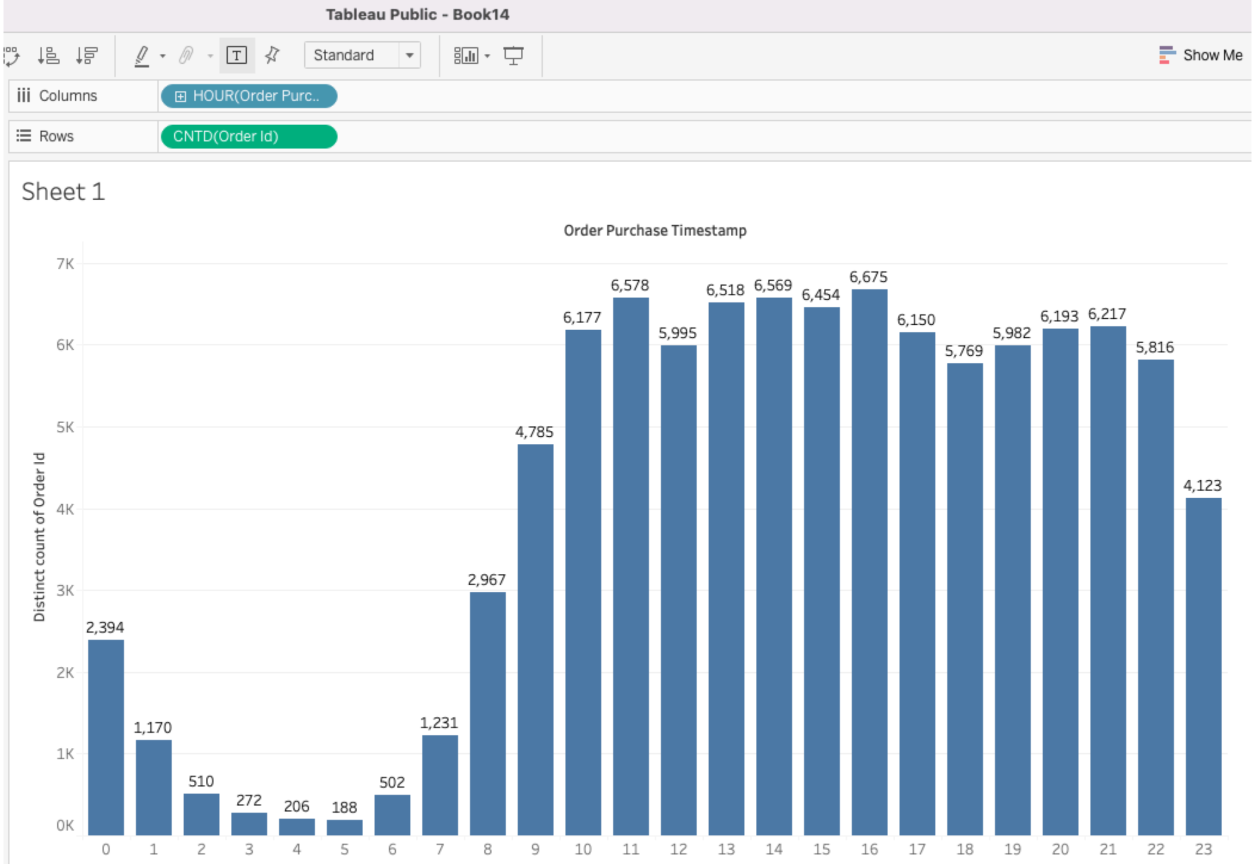


Tableau Public - Book14

Columns: HOUR(Order Purc..

Rows: CNTD(Order Id)

Sheet 1

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

3a. Get month on month orders by states

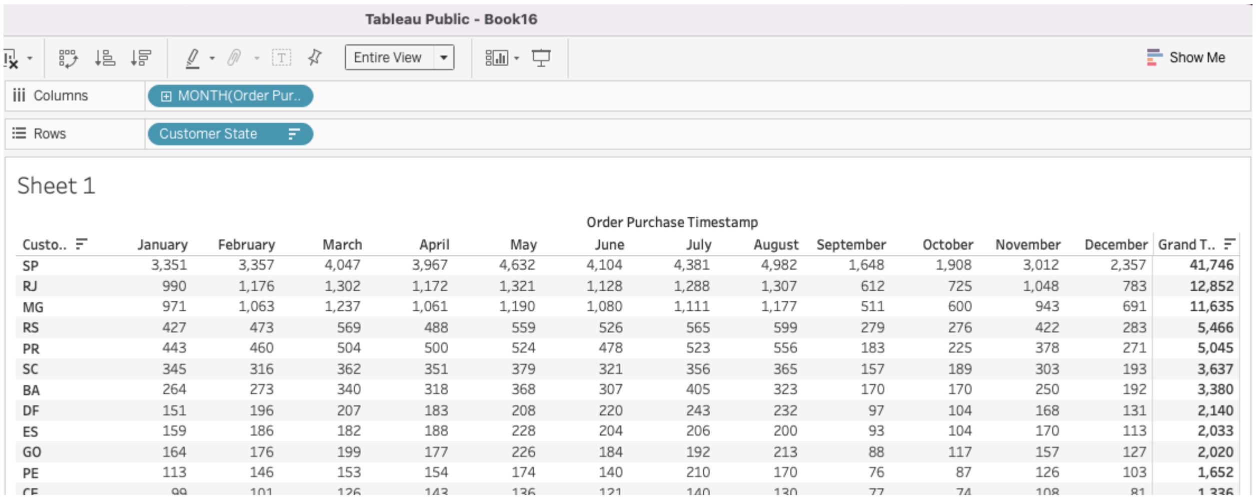


Tableau Public - Book16

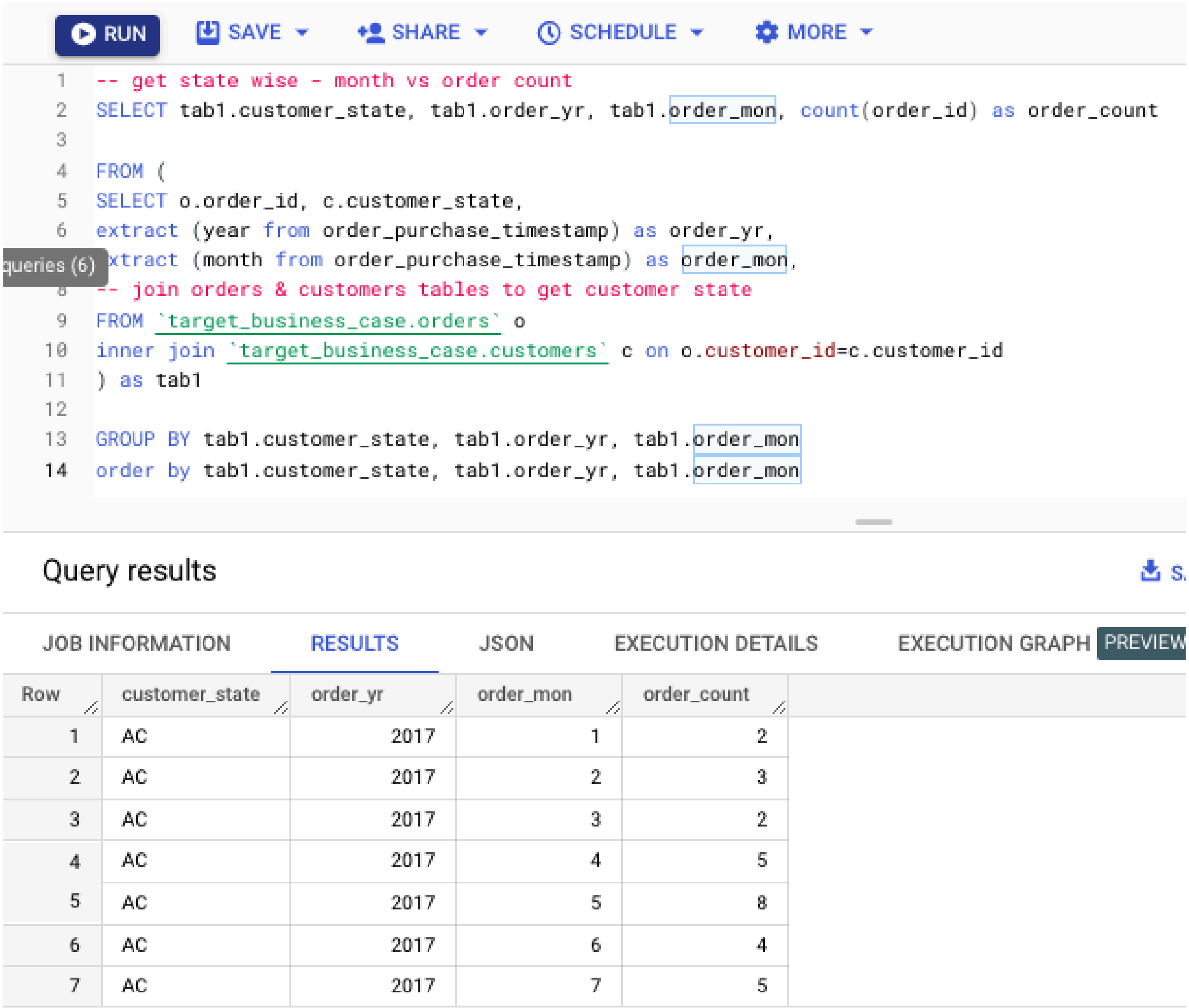
Columns: MONTH(Order Pur..

Rows: Customer State

Sheet 1

Order Purchase Timestamp

Custo..	January	February	March	April	May	June	July	August	September	October	November	December	Grand T..
SP	3,351	3,357	4,047	3,967	4,632	4,104	4,381	4,982	1,648	1,908	3,012	2,357	**41,746**
RJ	990	1,176	1,302	1,172	1,321	1,128	1,288	1,307	612	725	1,048	783	**12,852**
MG	971	1,063	1,237	1,061	1,190	1,080	1,111	1,177	511	600	943	691	**11,635**
RS	427	473	569	488	559	526	565	599	279	276	422	283	**5,466**
PR	443	460	504	500	524	478	523	556	183	225	378	271	**5,045**
SC	345	316	362	351	379	321	356	365	157	189	303	193	**3,637**
BA	264	273	340	318	368	307	405	323	170	170	250	192	**3,380**
DF	151	196	207	183	208	220	243	232	97	104	168	131	**2,140**
ES	159	186	182	188	228	204	206	200	93	104	170	113	**2,033**
GO	164	176	199	177	226	184	192	213	88	117	157	127	**2,020**
PE	113	146	153	154	174	140	210	170	76	87	126	103	**1,652**
CE	99	101	126	143	136	121	140	130	77	74	108	81	**1,336**

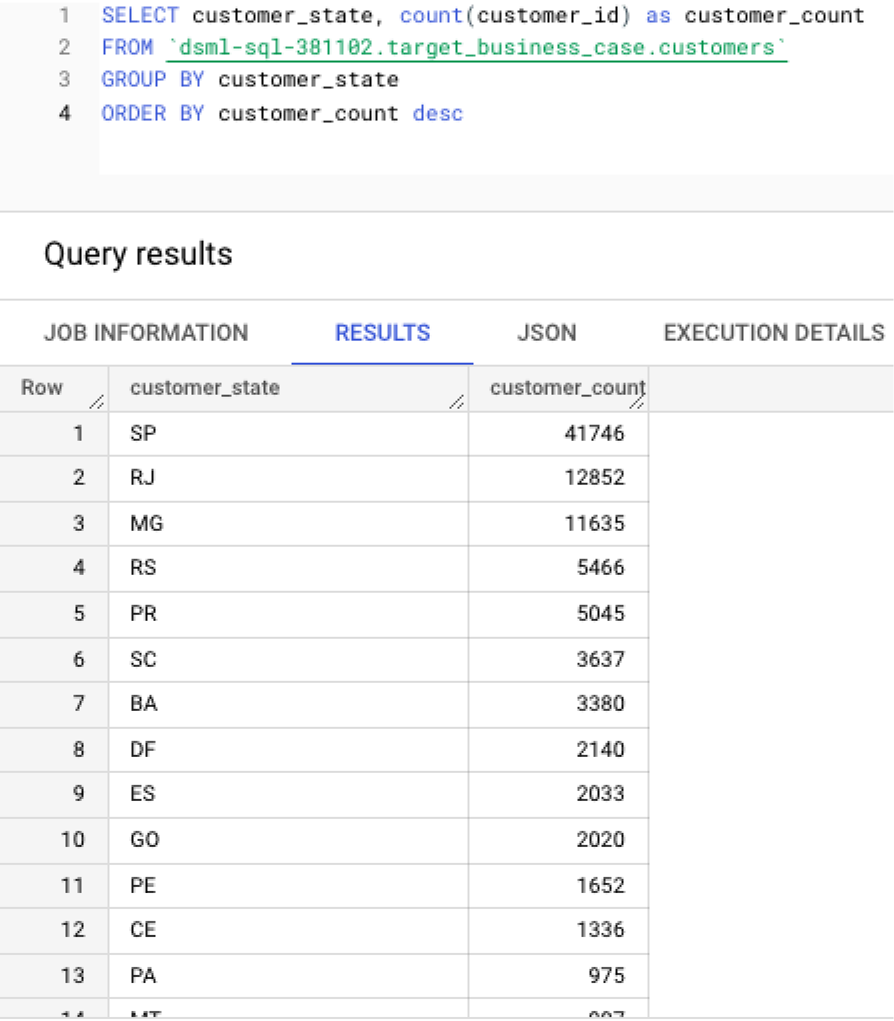


```sql
-- get state wise - month vs order count
SELECT tab1.customer_state, tab1.order_yr, tab1.order_mon, count(order_id) as order_count

FROM (
SELECT o.order_id, c.customer_state,
extract (year from order_purchase_timestamp) as order_yr,
xtract (month from order_purchase_timestamp) as order_mon,
-- join orders & customers tables to get customer state
FROM `target_business_case.orders` o
inner join `target_business_case.customers` c on o.customer_id=c.customer_id
) as tab1

GROUP BY tab1.customer_state, tab1.order_yr, tab1.order_mon
order by tab1.customer_state, tab1.order_yr, tab1.order_mon
```

Query results

Row	customer_state	order_yr	order_mon	order_count
1	AC	2017	1	2
2	AC	2017	2	3
3	AC	2017	3	2
4	AC	2017	4	5
5	AC	2017	5	8
6	AC	2017	6	4
7	AC	2017	7	5

3b. Distribution of customers across the states in Brazil



```sql
SELECT customer_state, count(customer_id) as customer_count
FROM `dsml-sql-381102.target_business_case.customers`
GROUP BY customer_state
ORDER BY customer_count desc
```

Query results

Row	customer_state	customer_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
11	PE	1652
12	CE	1336
13	PA	975
14	MT	907

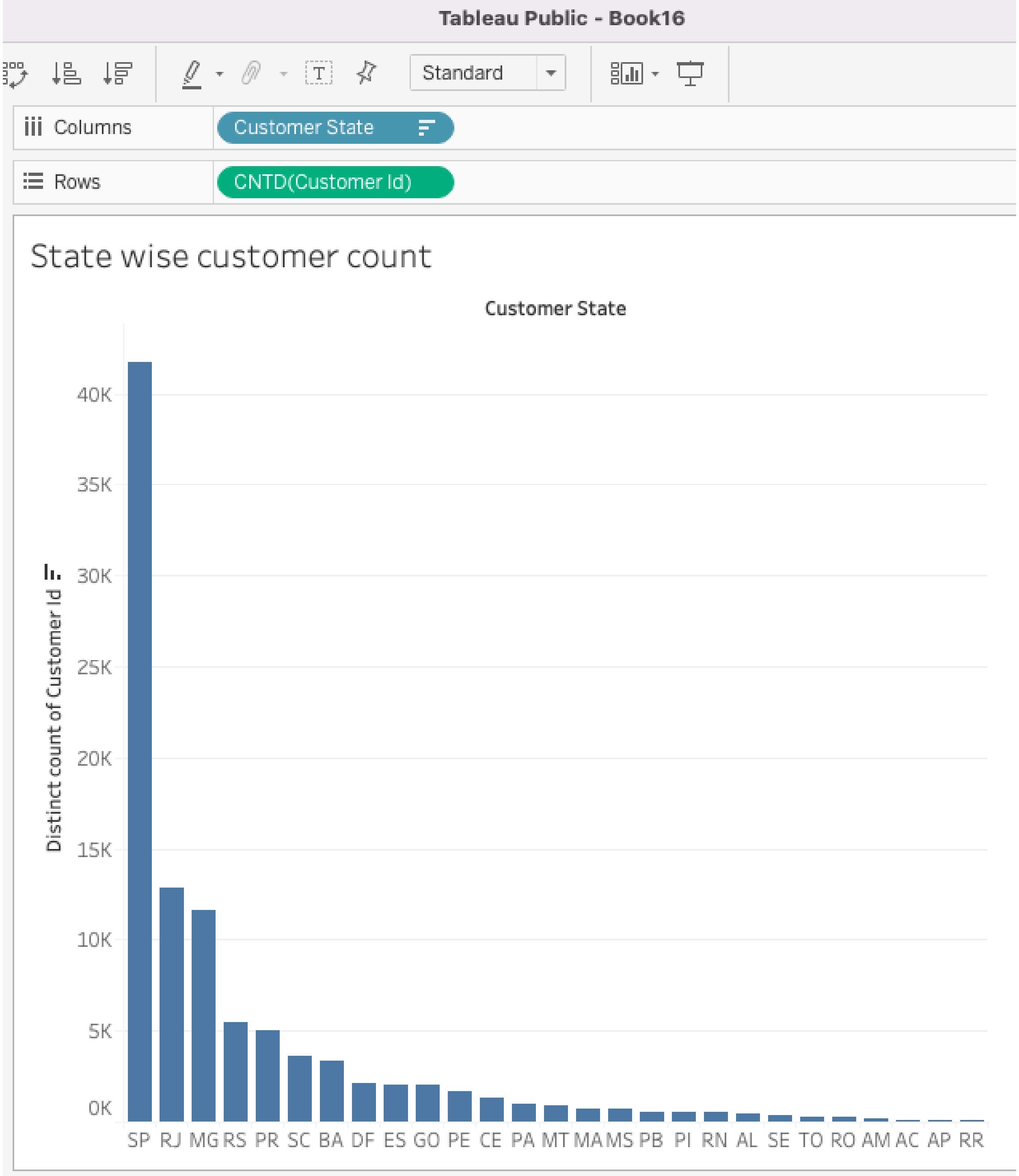


Tableau Public - Book16

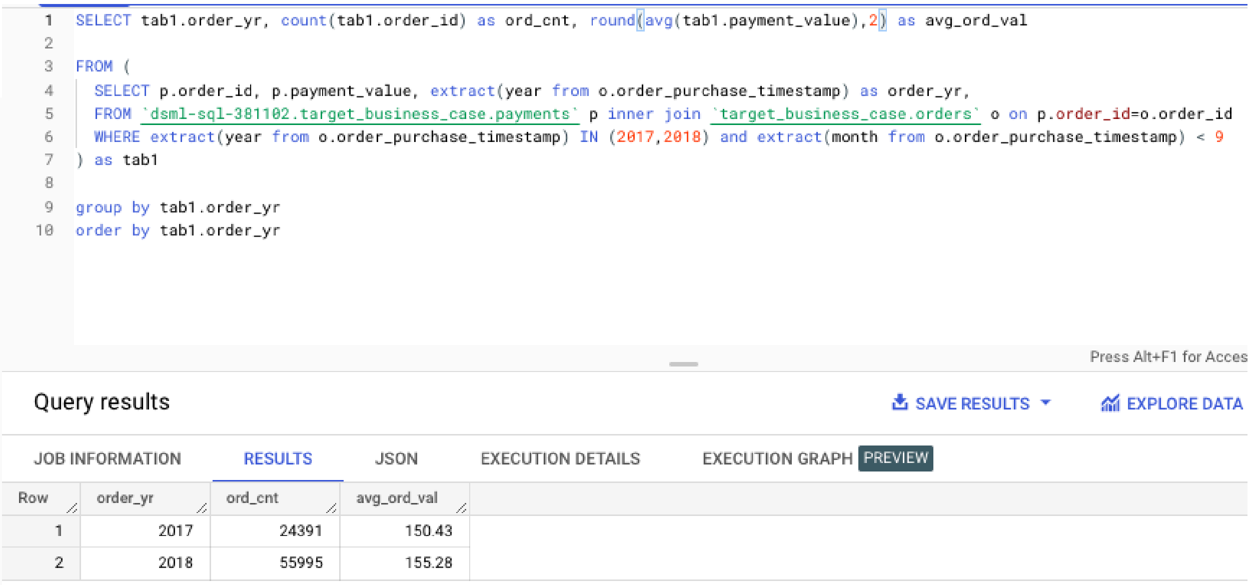
Columns: Customer State

Rows: CNTD(Customer Id)

Standard
State wise customer count
Customer State
Distinct count of Customer Id
40K
35K
30K
25K
20K
15K
10K
5K
0K
SP RJ MG RS PR SC BA DF ES GO PE CE PA MT MA MS PB PI RN AL SE TO RO AM AC AP RR

4. Impact on Economy

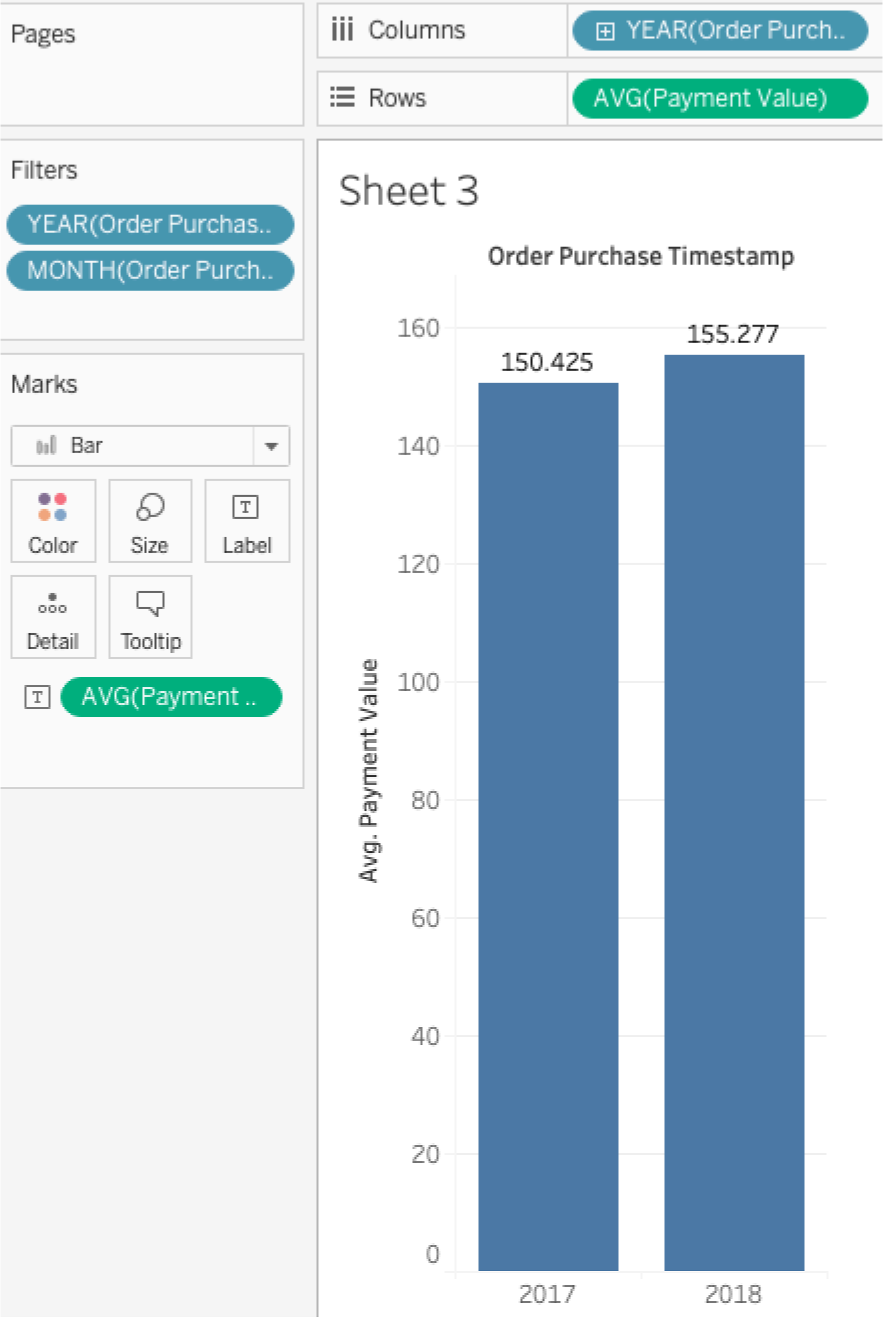
4a. Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only) - You can use “payment_value” column in payments table



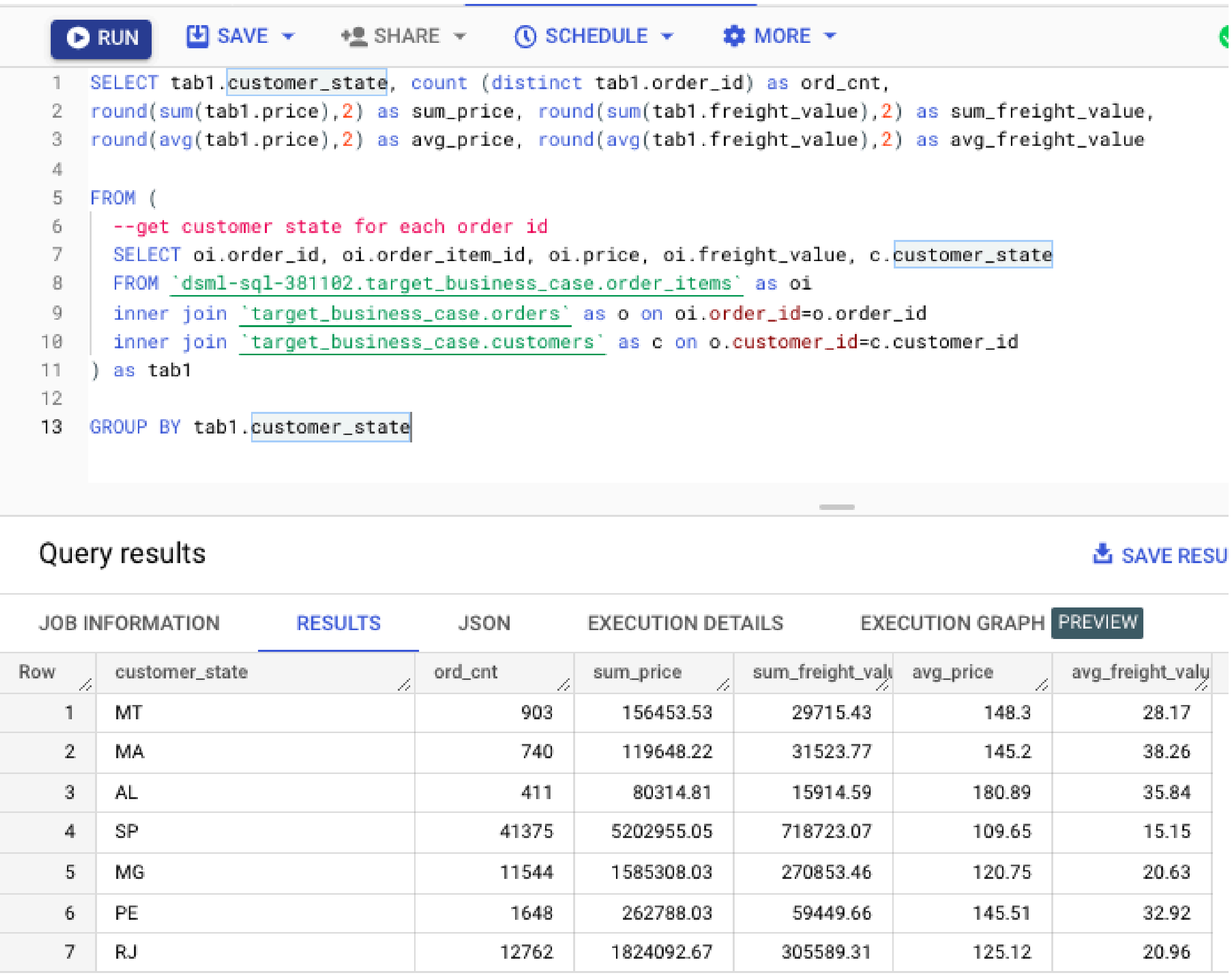
```
SELECT tab1.order_yr, count(tab1.order_id) as ord_cnt, round(avg(tab1.payment_value),2) as avg_ord_val

FROM (
  SELECT p.order_id, p.payment_value, extract(year from o.order_purchase_timestamp) as order_yr,
  FROM `dsml-sql-381102.target_business_case.payments` p inner join `target_business_case.orders` o on p.order_id=o.order_id
  WHERE extract(year from o.order_purchase_timestamp) IN (2017,2018) and extract(month from o.order_purchase_timestamp) < 9
) as tab1

group by tab1.order_yr
order by tab1.order_yr
```

Row	order_yr	ord_cnt	avg_ord_val
1	2017	24391	150.43
2	2018	55995	155.28

There is a 3.22% increase in the avg order value from 2017 ($150.43) to 2018 ($155.28)

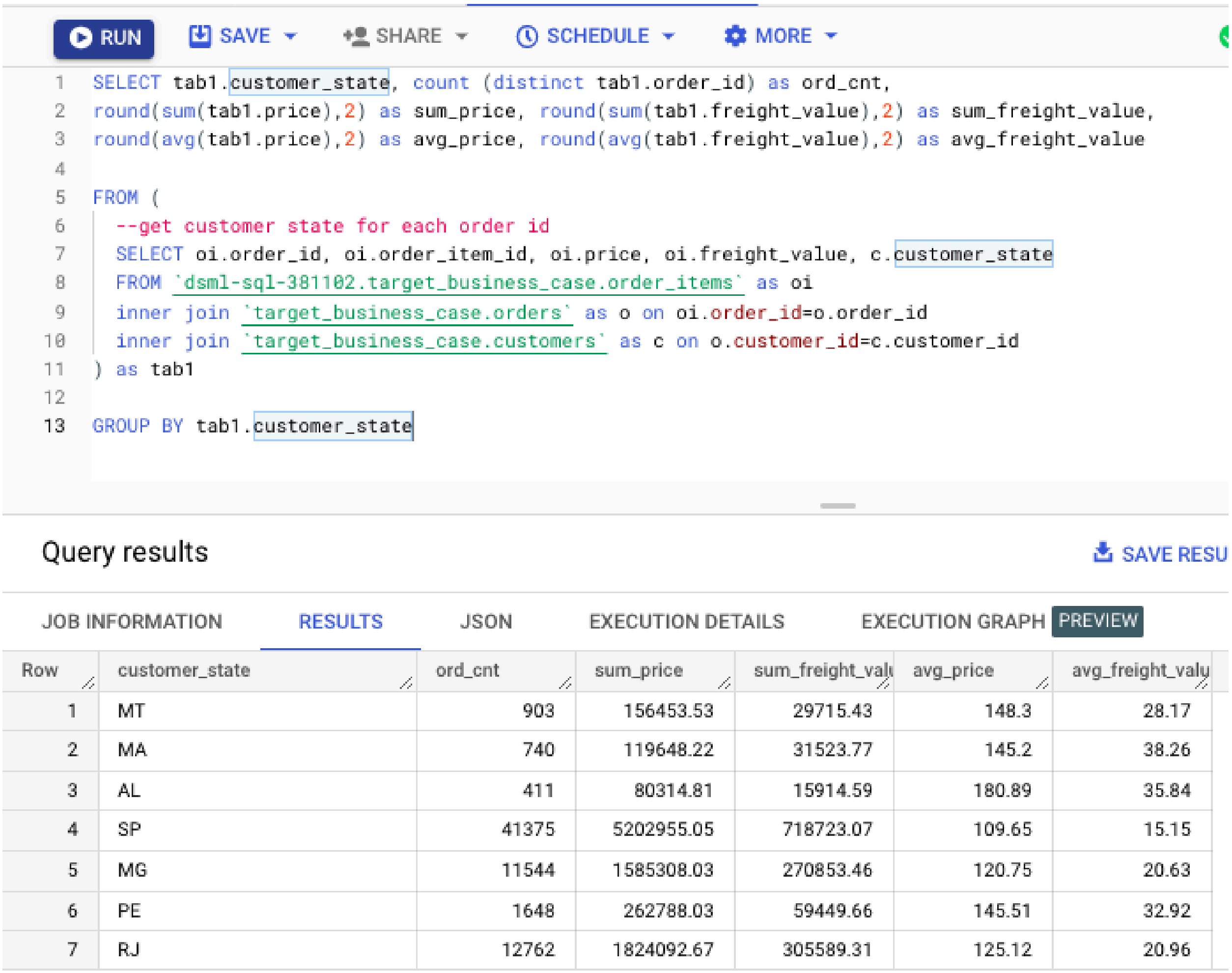


4b. Mean & Sum of price and freight value by customer state



```sql
SELECT tab1.customer_state, count (distinct tab1.order_id) as ord_cnt,
round(sum(tab1.price),2) as sum_price, round(sum(tab1.freight_value),2) as sum_freight_value,
round(avg(tab1.price),2) as avg_price, round(avg(tab1.freight_value),2) as avg_freight_value

FROM (
  --get customer state for each order id
  SELECT oi.order_id, oi.order_item_id, oi.price, oi.freight_value, c.customer_state
  FROM `dsml-sql-381102.target_business_case.order_items` as oi
  inner join `target_business_case.orders` as o on oi.order_id=o.order_id
  inner join `target_business_case.customers` as c on o.customer_id=c.customer_id
) as tab1

GROUP BY tab1.customer_state
```

Row	customer_state	ord_cnt	sum_price	sum_freight_valu	avg_price	avg_freight_valu
1	MT	903	156453.53	29715.43	148.3	28.17
2	MA	740	119648.22	31523.77	145.2	38.26
3	AL	411	80314.81	15914.59	180.89	35.84
4	SP	41375	5202955.05	718723.07	109.65	15.15
5	MG	11544	1585308.03	270853.46	120.75	20.63
6	PE	1648	262788.03	59449.66	145.51	32.92
7	RJ	12762	1824092.67	305589.31	125.12	20.96



```sql
SELECT tab1.customer_state, count (distinct tab1.order_id) as ord_cnt,
round(sum(tab1.price),2) as sum_price, round(sum(tab1.freight_value),2) as sum_freight_value,
round(avg(tab1.price),2) as avg_price, round(avg(tab1.freight_value),2) as avg_freight_value

FROM (
  --get customer state for each order id
  SELECT oi.order_id, oi.order_item_id, oi.price, oi.freight_value, c.customer_state
  FROM `dsml-sql-381102.target_business_case.order_items` as oi
  inner join `target_business_case.orders` as o on oi.order_id=o.order_id
  inner join `target_business_case.customers` as c on o.customer_id=c.customer_id
) as tab1

GROUP BY tab1.customer_state
```

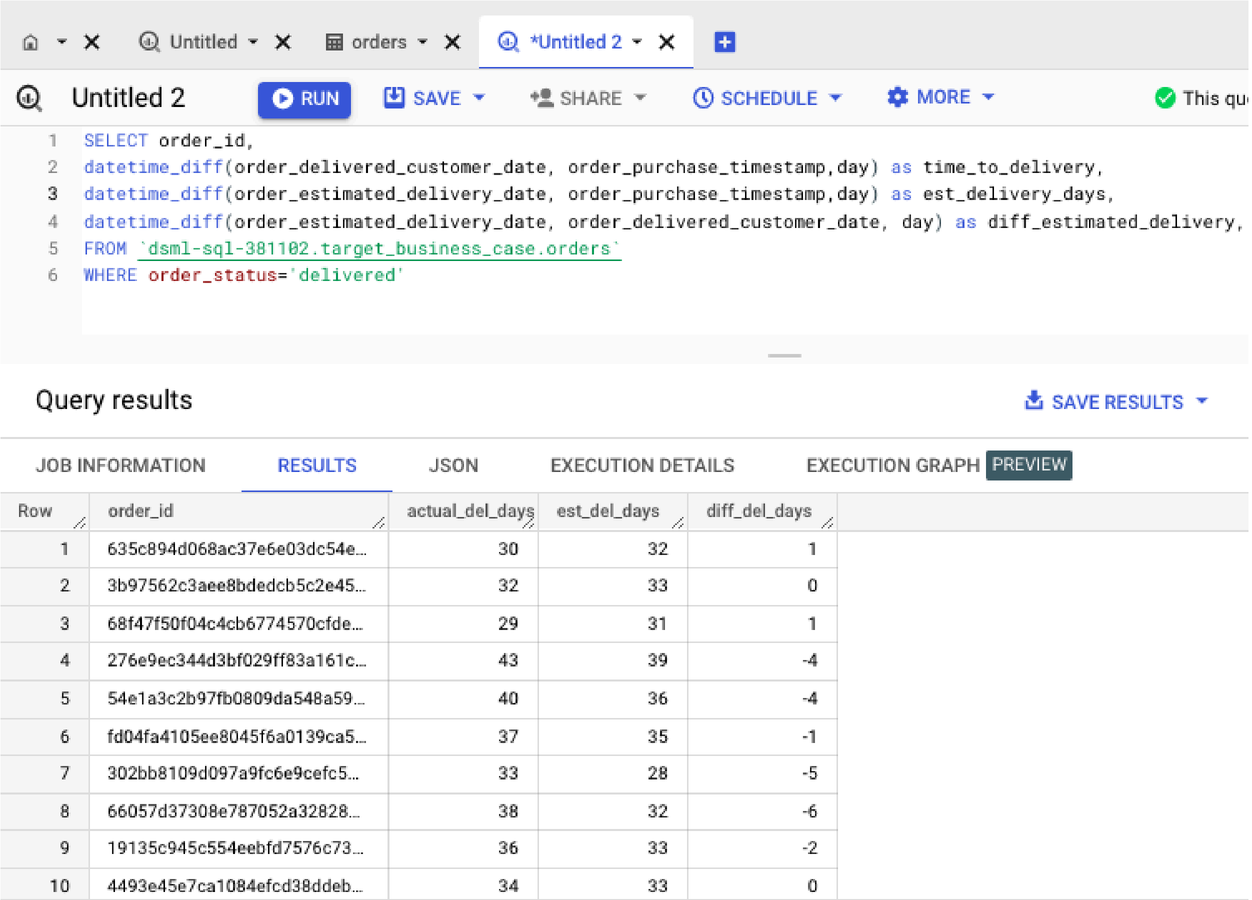
Row	customer_state	ord_cnt	sum_price	sum_freight_valu	avg_price	avg_freight_valu
1	MT	903	156453.53	29715.43	148.3	28.17
2	MA	740	119648.22	31523.77	145.2	38.26
3	AL	411	80314.81	15914.59	180.89	35.84
4	SP	41375	5202955.05	718723.07	109.65	15.15
5	MG	11544	1585308.03	270853.46	120.75	20.63
6	PE	1648	262788.03	59449.66	145.51	32.92
7	RJ	12762	1824092.67	305589.31	125.12	20.96

5. Analysis on sales, freight and delivery time

5a. Calculate days between purchasing, delivering and estimated delivery

Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:

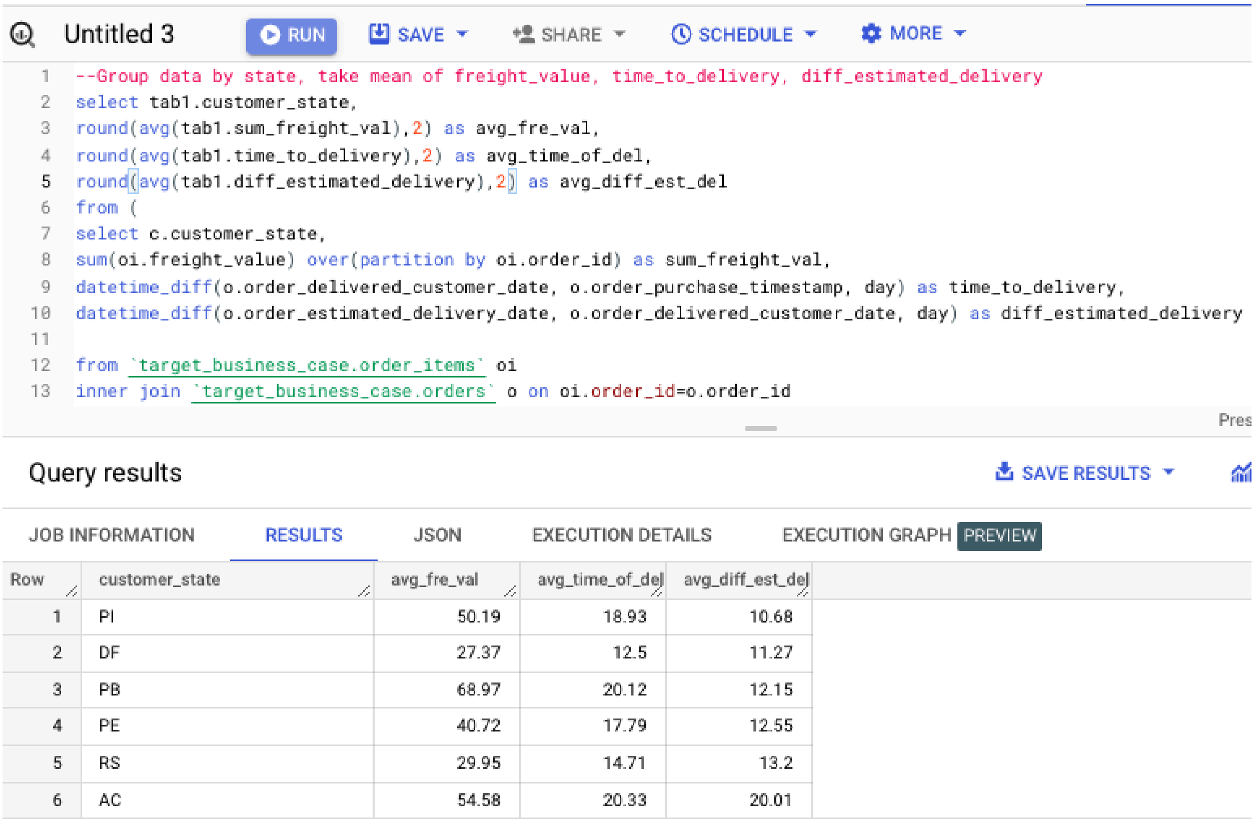
- time_to_delivery = order_purchase_timestamp-order_delivered_customer_date
- diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date



```sql
SELECT order_id,
datetime_diff(order_delivered_customer_date, order_purchase_timestamp,day) as time_to_delivery,
datetime_diff(order_estimated_delivery_date, order_purchase_timestamp,day) as est_delivery_days,
datetime_diff(order_estimated_delivery_date, order_delivered_customer_date, day) as diff_estimated_delivery,
FROM `dsml-sql-381102.target_business_case.orders`
WHERE order_status='delivered'
```

Row	order_id	actual_del_days	est_del_days	diff_del_days
1	635c894d068ac37e6e03dc54e...	30	32	1
2	3b97562c3aee8bdedcb5c2e45...	32	33	0
3	68f47f50f04c4cb6774570cfde...	29	31	1
4	276e9ec344d3bf029ff83a161c...	43	39	-4
5	54e1a3c2b97fb0809da548a59...	40	36	-4
6	fd04fa4105ee8045f6a0139ca5...	37	35	-1
7	302bb8109d097a9fc6e9cefc5...	33	28	-5
8	66057d37308e787052a32828...	38	32	-6
9	19135c945c554eebfd7576c73...	36	33	-2
10	4493e45e7ca1084efcd38ddeb...	34	33	0

5c. Group data by state, take mean of freight_value, `time_to_delivery`, `diff_estimated_delivery`

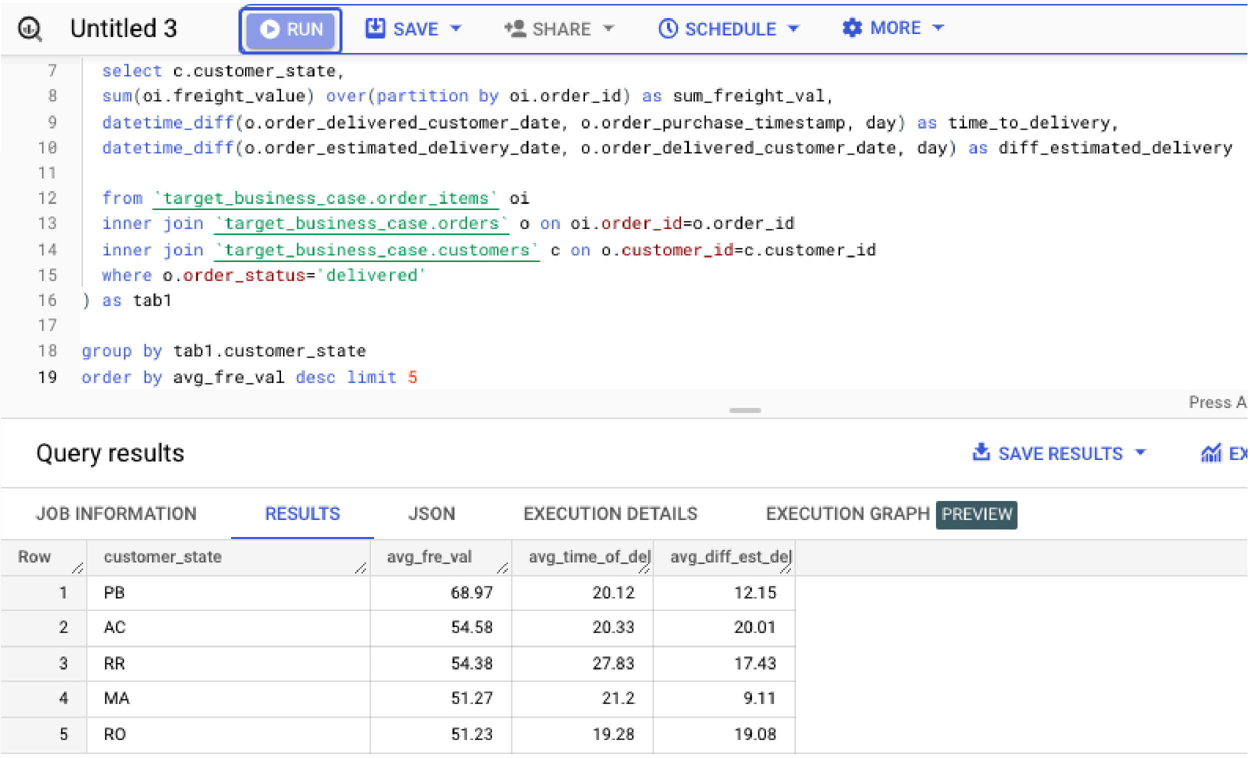


```
--Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery
select tab1.customer_state,
round(avg(tab1.sum_freight_val),2) as avg_fre_val,
round(avg(tab1.time_to_delivery),2) as avg_time_of_del,
round(avg(tab1.diff_estimated_delivery),2) as avg_diff_est_del
from (
select c.customer_state,
sum(oi.freight_value) over(partition by oi.order_id) as sum_freight_val,
datetime_diff(o.order_delivered_customer_date, o.order_purchase_timestamp, day) as time_to_delivery,
datetime_diff(o.order_estimated_delivery_date, o.order_delivered_customer_date, day) as diff_estimated_delivery

from `target_business_case.order_items` oi
inner join `target_business_case.orders` o on oi.order_id=o.order_id
```

Row	customer_state	avg_fre_val	avg_time_of_del	avg_diff_est_del
1	PI	50.19	18.93	10.68
2	DF	27.37	12.5	11.27
3	PB	68.97	20.12	12.15
4	PE	40.72	17.79	12.55
5	RS	29.95	14.71	13.2
6	AC	54.58	20.33	20.01

5d. Sort the data to get the following:

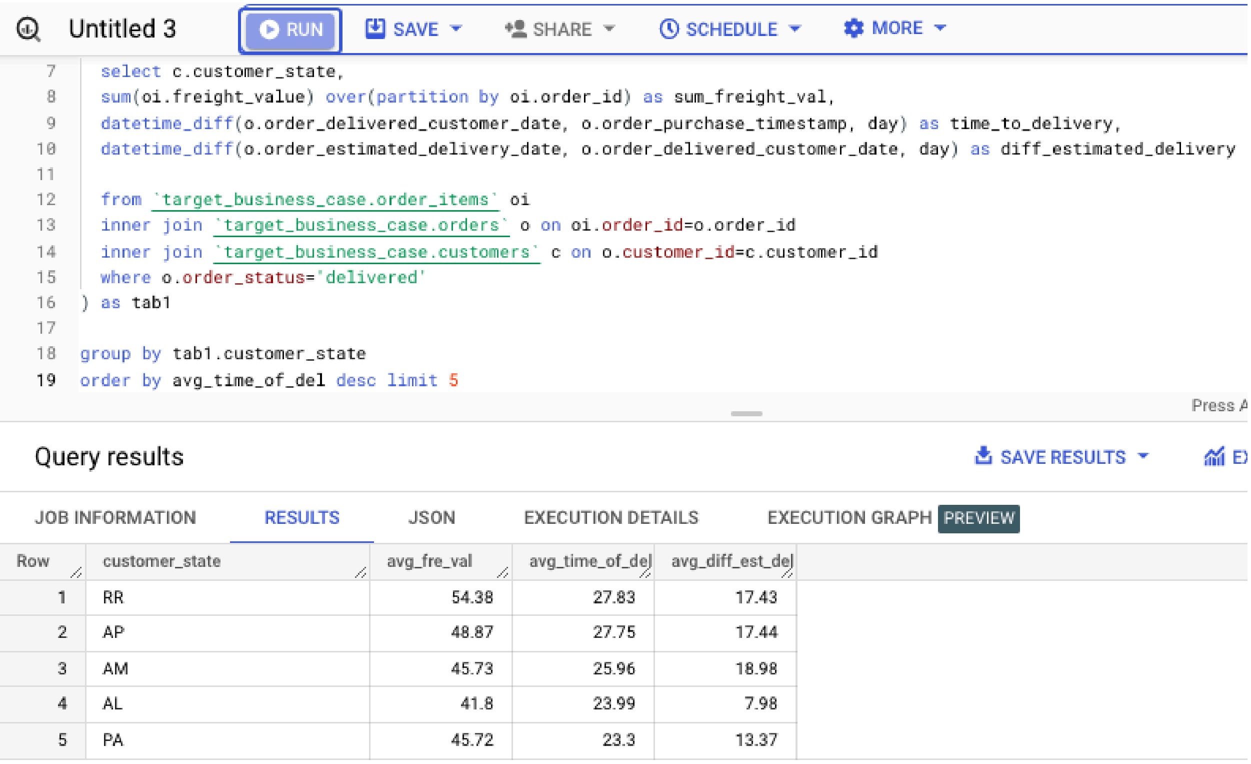
- Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5



```
  select c.customer_state,
  sum(oi.freight_value) over(partition by oi.order_id) as sum_freight_val,
  datetime_diff(o.order_delivered_customer_date, o.order_purchase_timestamp, day) as time_to_delivery,
  datetime_diff(o.order_estimated_delivery_date, o.order_delivered_customer_date, day) as diff_estimated_delivery

  from `target_business_case.order_items` oi
  inner join `target_business_case.orders` o on oi.order_id=o.order_id
  inner join `target_business_case.customers` c on o.customer_id=c.customer_id
  where o.order_status='delivered'
) as tab1

group by tab1.customer_state
order by avg_fre_val desc limit 5
```

Row	customer_state	avg_fre_val	avg_time_of_del	avg_diff_est_del
1	PB	68.97	20.12	12.15
2	AC	54.58	20.33	20.01
3	RR	54.38	27.83	17.43
4	MA	51.27	21.2	9.11
5	RO	51.23	19.28	19.08

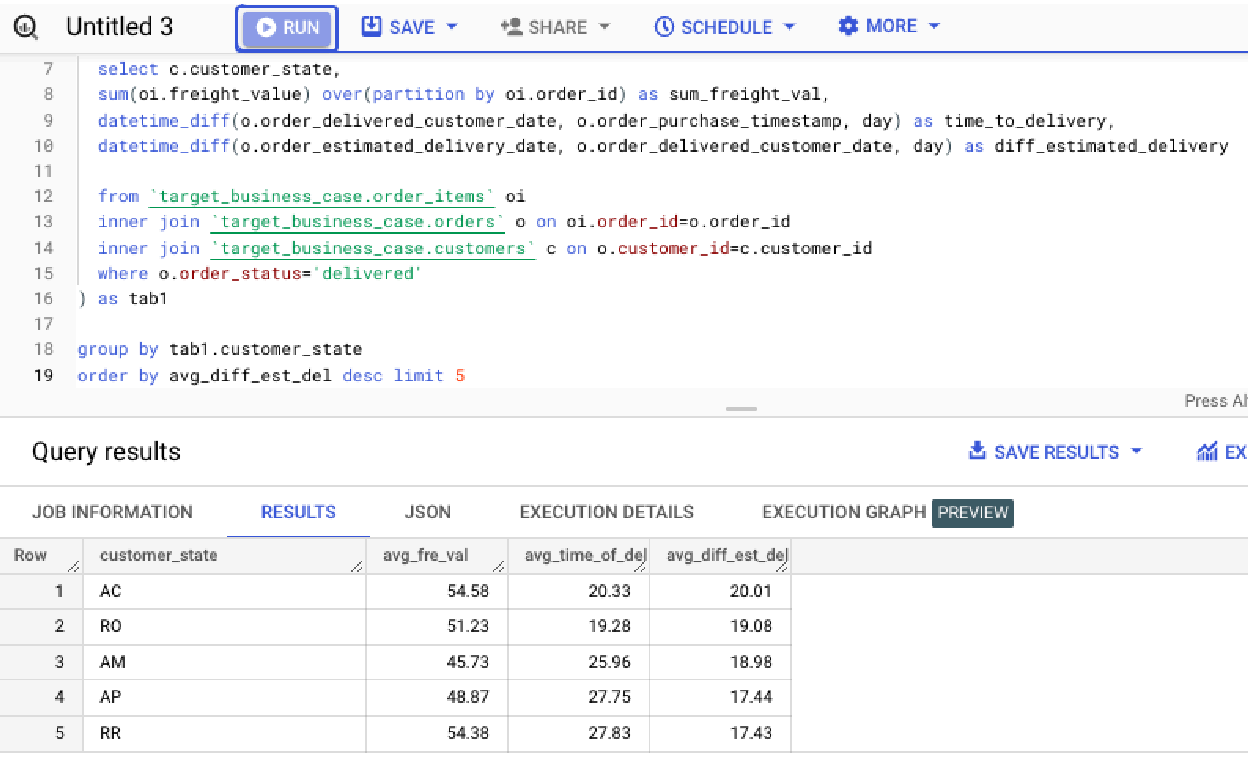
- Top 5 states with highest/lowest average time to delivery



```
  select c.customer_state,
  sum(oi.freight_value) over(partition by oi.order_id) as sum_freight_val,
  datetime_diff(o.order_delivered_customer_date, o.order_purchase_timestamp, day) as time_to_delivery,
  datetime_diff(o.order_estimated_delivery_date, o.order_delivered_customer_date, day) as diff_estimated_delivery

  from `target_business_case.order_items` oi
  inner join `target_business_case.orders` o on oi.order_id=o.order_id
  inner join `target_business_case.customers` c on o.customer_id=c.customer_id
  where o.order_status='delivered'
) as tab1

group by tab1.customer_state
order by avg_time_of_del desc limit 5
```

Row	customer_state	avg_fre_val	avg_time_of_del	avg_diff_est_del
1	RR	54.38	27.83	17.43
2	AP	48.87	27.75	17.44
3	AM	45.73	25.96	18.98
4	AL	41.8	23.99	7.98
5	PA	45.72	23.3	13.37

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

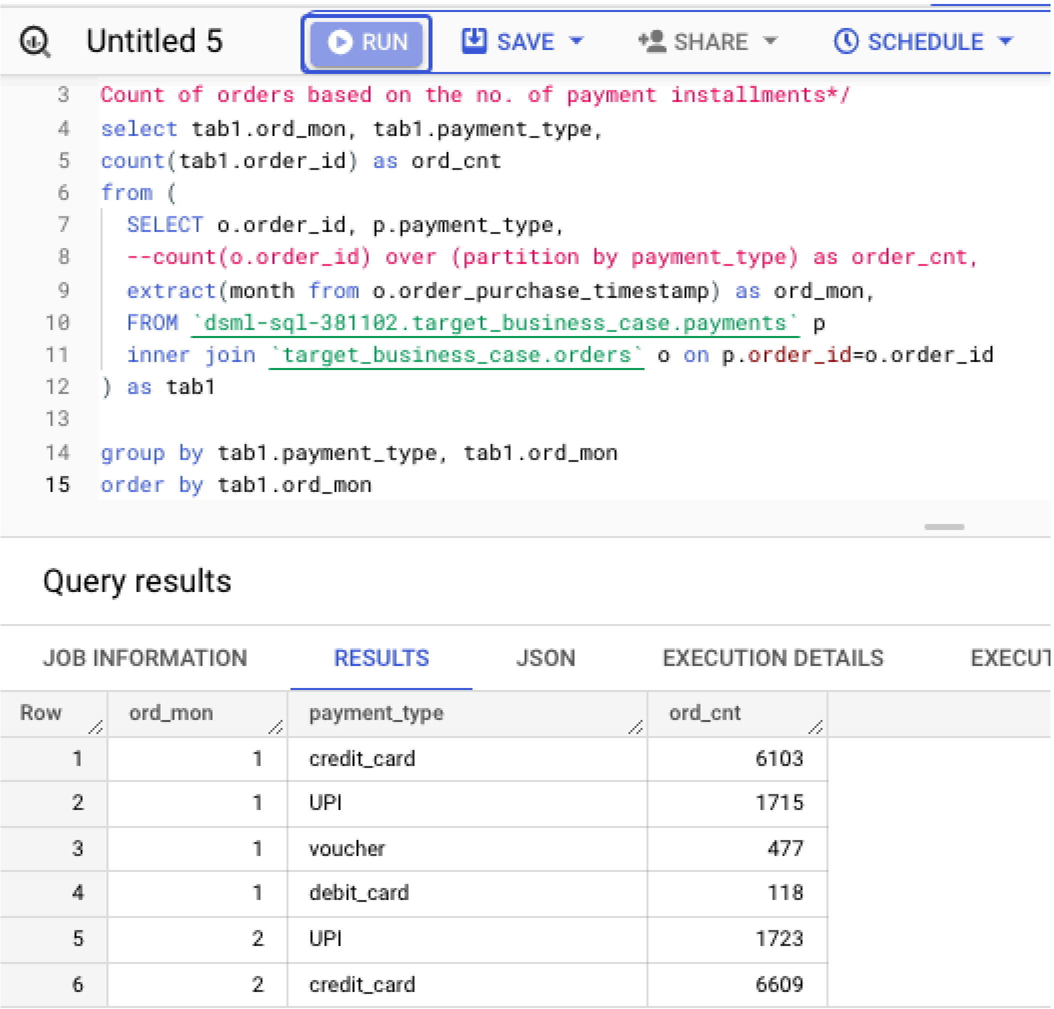


```
  select c.customer_state,
  sum(oi.freight_value) over(partition by oi.order_id) as sum_freight_val,
  datetime_diff(o.order_delivered_customer_date, o.order_purchase_timestamp, day) as time_to_delivery,
  datetime_diff(o.order_estimated_delivery_date, o.order_delivered_customer_date, day) as diff_estimated_delivery

  from `target_business_case.order_items` oi
  inner join `target_business_case.orders` o on oi.order_id=o.order_id
  inner join `target_business_case.customers` c on o.customer_id=c.customer_id
  where o.order_status='delivered'
) as tab1

group by tab1.customer_state
order by avg_diff_est_del desc limit 5
```

Row	customer_state	avg_fre_val	avg_time_of_del	avg_diff_est_del
1	AC	54.58	20.33	20.01
2	RO	51.23	19.28	19.08
3	AM	45.73	25.96	18.98
4	AP	48.87	27.75	17.44
5	RR	54.38	27.83	17.43

6. Payment type analysis:

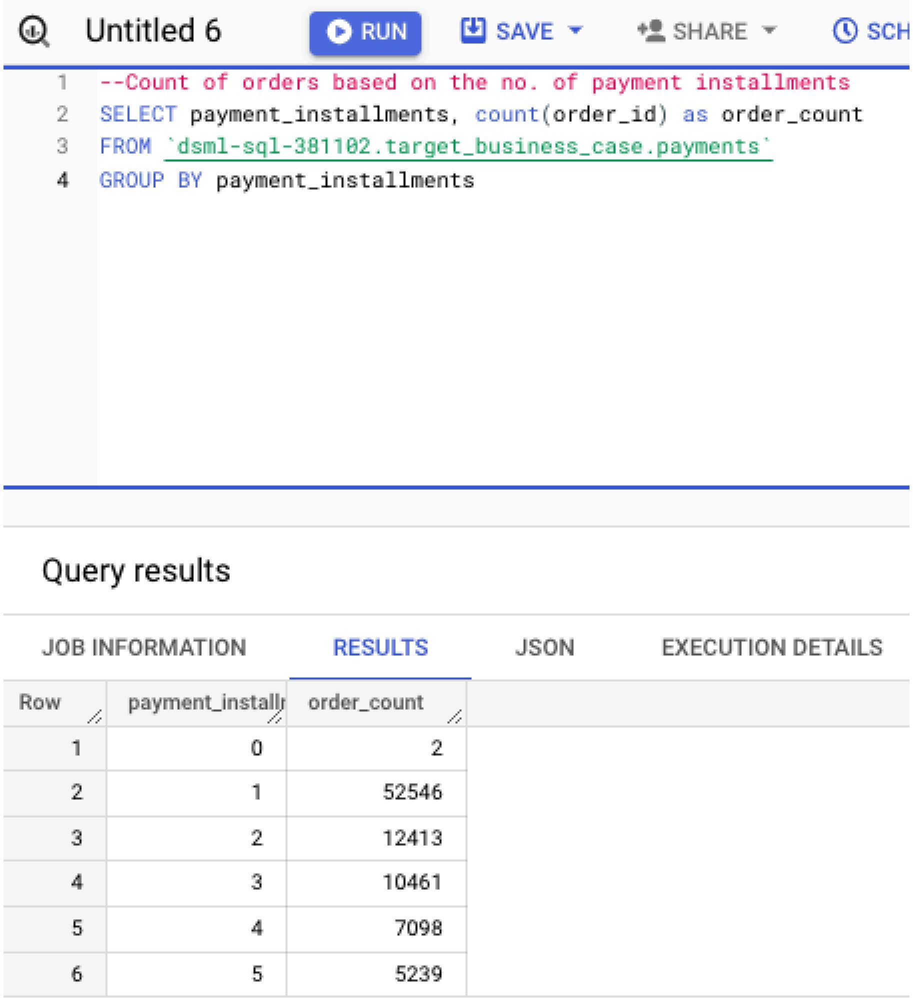
6a. Month over Month count of orders for different payment types



```
Count of orders based on the no. of payment installments*/
select tab1.ord_mon, tab1.payment_type,
count(tab1.order_id) as ord_cnt
from (
  SELECT o.order_id, p.payment_type,
  --count(o.order_id) over (partition by payment_type) as order_cnt,
  extract(month from o.order_purchase_timestamp) as ord_mon,
  FROM `dsml-sql-381102.target_business_case.payments` p
  inner join `target_business_case.orders` o on p.order_id=o.order_id
) as tab1

group by tab1.payment_type, tab1.ord_mon
order by tab1.ord_mon
```

Row	ord_mon	payment_type	ord_cnt
1	1	credit_card	6103
2	1	UPI	1715
3	1	voucher	477
4	1	debit_card	118
5	2	UPI	1723
6	2	credit_card	6609

6b. Count of orders based on the no. of payment installments



```
--Count of orders based on the no. of payment installments
SELECT payment_installments, count(order_id) as order_count
FROM `dsml-sql-381102.target_business_case.payments`
GROUP BY payment_installments
```

Row	payment_installı	order_count
1	0	2
2	1	52546
3	2	12413
4	3	10461
5	4	7098
6	5	5239

📊 Summary of Key Insights

The dataset spans **2016 to 2018**, with the most transactions recorded in **2018**.

- Orders were mostly placed in the **afternoon hours** (12 PM to 4 PM).
- **São Paulo** had the highest number of orders, followed by **Rio de Janeiro** and **Minas Gerais**.
- There was a noticeable **increase in average order value and freight value** from 2016 to 2018.
- Average **time taken to deliver an order** was around **12 days**.
- The **difference between estimated and actual delivery** was highest in **Amazonas** and **Roraima**, indicating consistent delays in those regions.
- **Credit cards** were the most preferred payment method throughout.
- Most orders were placed using **1-installment** plans, followed by **2 and 3 installments**.

✅ Business Recommendations

Focus on **afternoon campaigns** (12–4 PM) when customer activity is at its peak.

- Investigate **logistics performance in Amazonas and Roraima**, where actual delivery is significantly slower than estimated.
- Highlight **credit card and 1-installment options** in the checkout flow to align with customer payment preferences.
- Leverage **high-performing regions** like São Paulo for loyalty programs or regional growth pilots.
- Continue tracking **order value and freight trends** annually to assess pricing strategy and shipping efficiency.