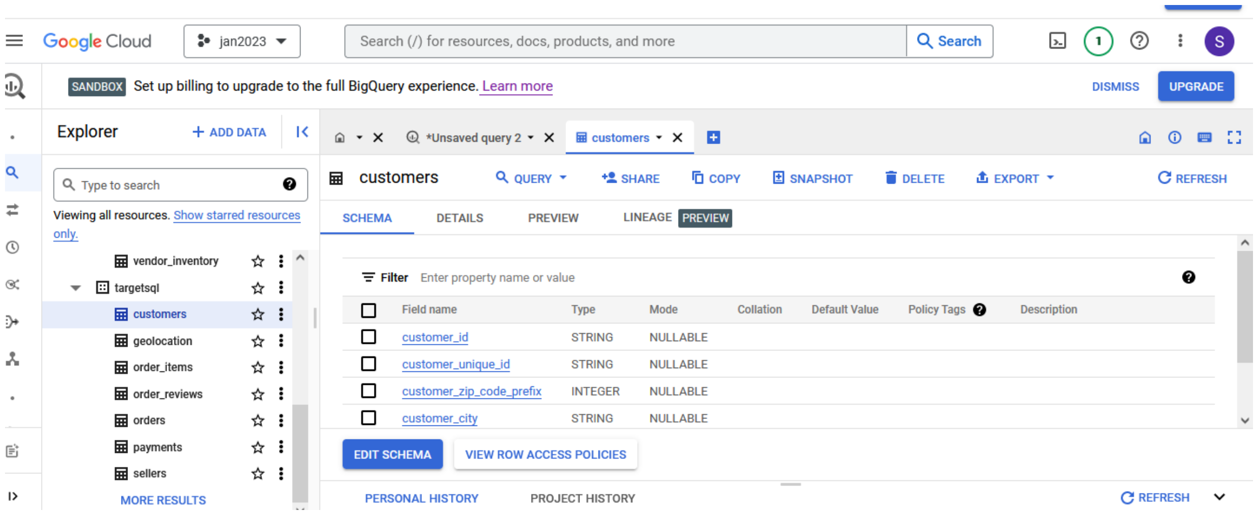
Question 1)

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

a. Data type of columns in a table



Google Cloud | jan2023 | Search (/) for resources, docs, products, and more | Search

SANDBOX Set up billing to upgrade to the full BigQuery experience. Learn more | DISMISS | UPGRADE

Explorer | + ADD DATA

Type to search

Viewing all resources. Show starred resources only.

- vendor_inventory
- targetsql
  - customers
  - geolocation
  - order_items
  - order_reviews
  - orders
  - payments
  - sellers

MORE RESULTS

*Unsaved query 2 | customers

customers | QUERY | SHARE | COPY | SNAPSHOT | DELETE | EXPORT | REFRESH

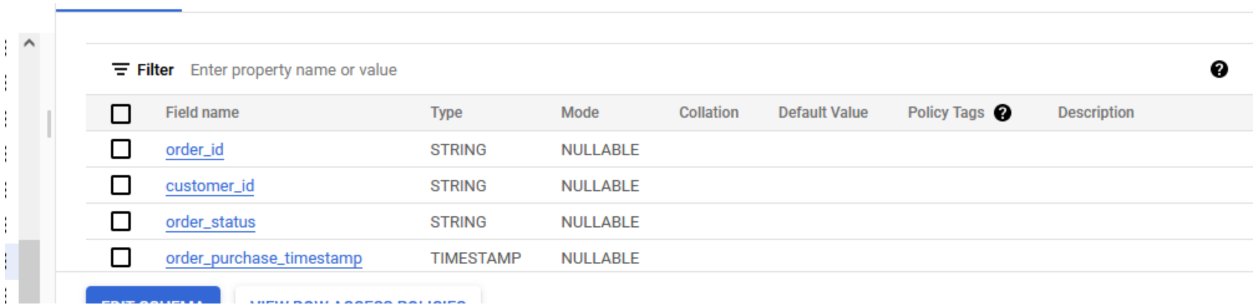
SCHEMA | DETAILS | PREVIEW | LINEAGE PREVIEW

Filter Enter property name or value

Field name	Type	Mode	Collation	Default Value	Policy Tags	Description
customer_id	STRING	NULLABLE				
customer_unique_id	STRING	NULLABLE				
customer_zip_code_prefix	INTEGER	NULLABLE				
customer_city	STRING	NULLABLE				

EDIT SCHEMA | VIEW ROW ACCESS POLICIES

PERSONAL HISTORY | PROJECT HISTORY | REFRESH

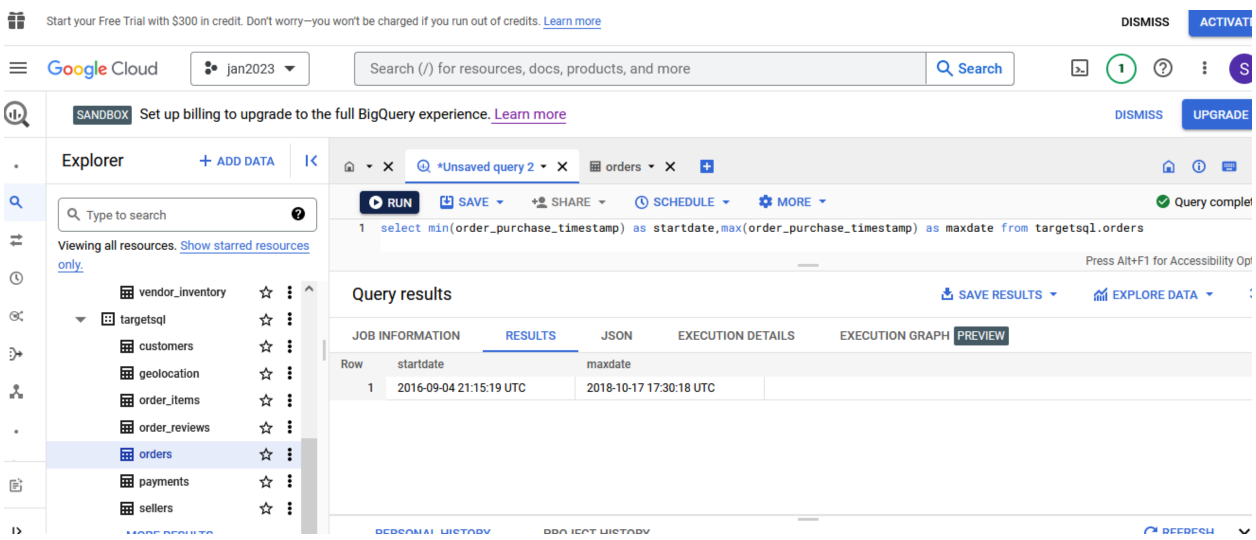


Filter Enter property name or value

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				

b. Time period for which the data is given

Select min(order_purchase_timestamp) as startdate, max(order_purchase_timestamp) as maxdate from targetsql.orders


Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more

SANDBOX Set up billing to upgrade to the full BigQuery experience. Learn more

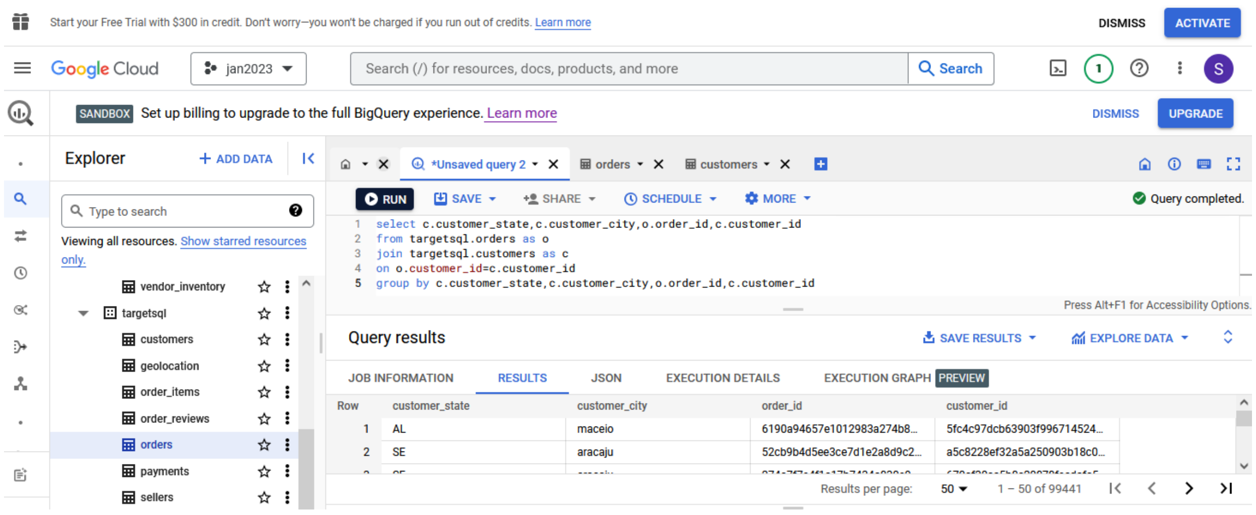
```
select min(order_purchase_timestamp) as startdate,max(order_purchase_timestamp) as maxdate from targetsql.orders
```

Query results

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

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

```
select c.customer_state,c.customer_city
from targetsql.orders as o
join targetsql.customers as c
on o.customer_id=c.customer_id
group by c.customer_state,c.customer_city
```


Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more

SANDBOX Set up billing to upgrade to the full BigQuery experience. Learn more

```
select c.customer_state,c.customer_city,o.order_id,c.customer_id
from targetsql.orders as o
join targetsql.customers as c
on o.customer_id=c.customer_id
group by c.customer_state,c.customer_city,o.order_id,c.customer_id
```

Query completed.

Query results

Row	customer_state	customer_city	order_id	customer_id
1	AL	maceio	6190a94657e1012983a274b8...	5fc4c97dcb63903f996714524...
2	SE	aracaju	52cb9b4d5ee3ce7d1e2a8d9c2...	a5c8228ef32a5a250903b18c0...

Results per page: 50 1 – 50 of 99441

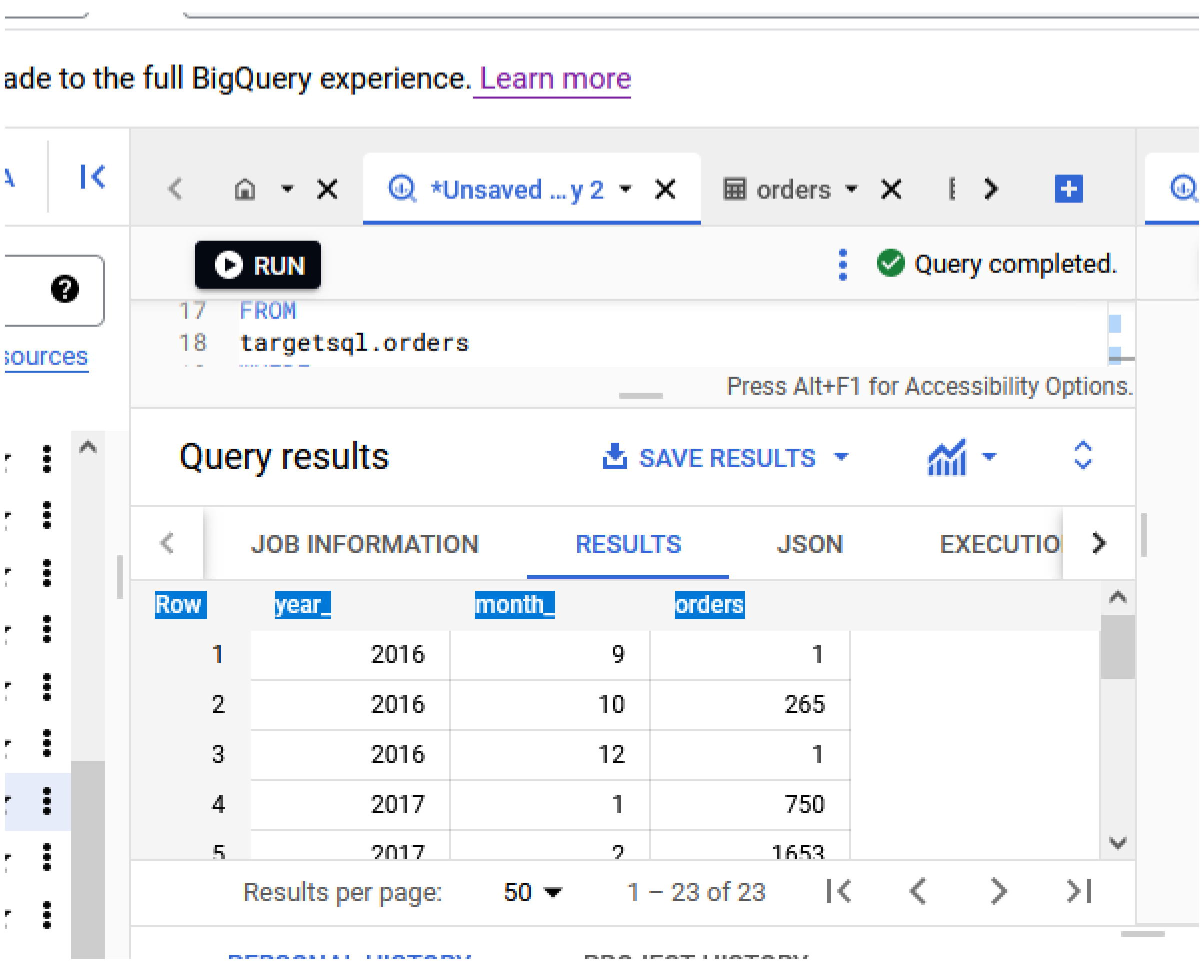
----

2. In-depth Exploration:

a)

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
EXTRACT(year FROM order_purchase_timestamp) AS year_,
EXTRACT(month FROM order_purchase_timestamp) AS month_,
COUNT(DISTINCT order_id) AS orders
FROM
targetsql.orders
WHERE
order_status = 'delivered'
GROUP BY
year_,month_
ORDER BY
year_,month_;
```



ade to the full BigQuery experience. Learn more

```
FROM
targetsql.orders
```

Row	year_	month_	orders
1	2016	9	1
2	2016	10	265
3	2016	12	1
4	2017	1	750
5	2017	2	1653

Results per page: 50 1 – 23 of 23

Row	year_	month_	orders
1	2016	9	1
2	2016	10	265
3	2016	12	1
4	2017	1	750
5	2017	2	1653
6	2017	3	2546
7	2017	4	2303
8	2017	5	3546
9	2017	6	3135
10	2017	7	3872
11	2017	8	4193
12	2017	9	4150
13	2017	10	4478
14	2017	11	7289

15	2017	12	5513
16	2018	1	7069
17	2018	2	6555
18	2018	3	7003
19	2018	4	6798
20	2018	5	6749
21	2018	6	6099
22	2018	7	6159
23	2018	8	6351

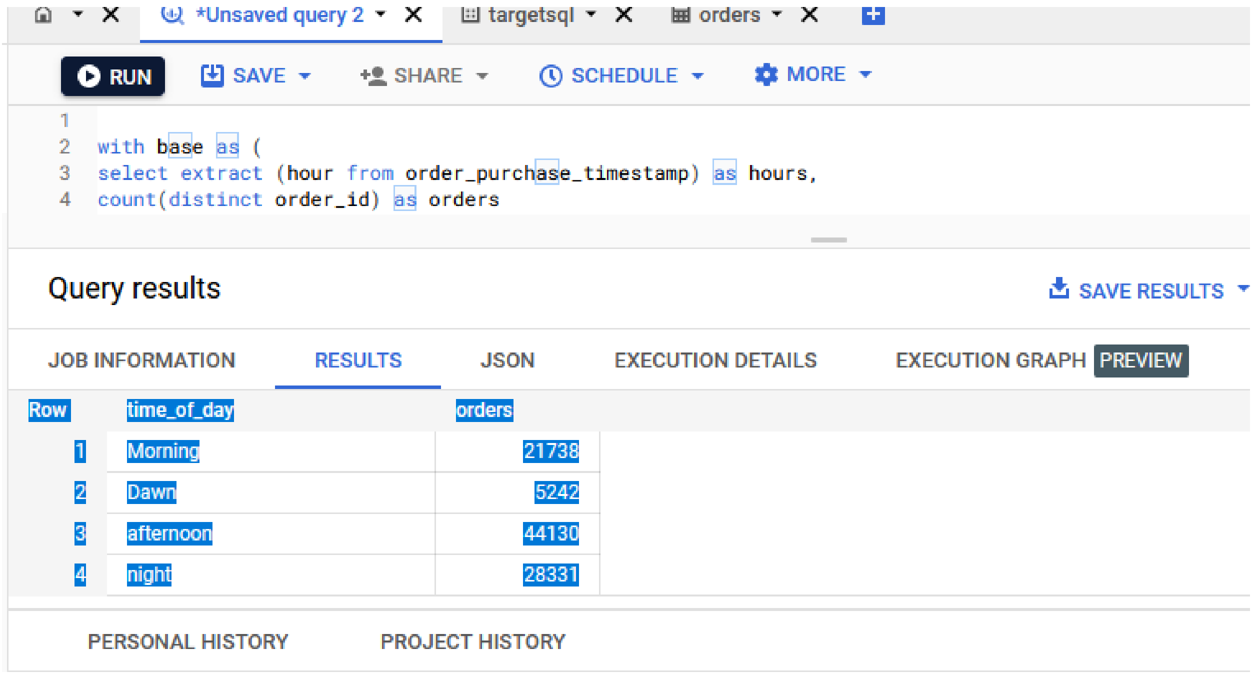
There is a growing trend in e-commerce in brazil.Sales are more in November, December, analysis done based on year 2017 because it has complete months list.

b)

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

```
with base as (
select extract (hour from order_purchase_timestamp) as hours,
count(distinct order_id) as orders
from targetsql.orders
group by 1
),
base_2 as (
select *,case when hours between 0 and 6 then 'Dawn'
              when hours between 7 and 11 then 'Morning'
              when hours between 12 and 18 then 'afternoon'
              when hours between 19 and 23 then 'night'
end as time_of_day
from base
)
select time_of_day,sum(orders) as orders from base_2
group by time_of_day
```

Row	time_of_day	orders
1	Morning	21738
2	Dawn	5242
3	afternoon	44130
4	night	28331


*Unsaved query 2
targetsql
orders
RUN
SAVE
SHARE
SCHEDULE
MORE

```
with base as (
select extract (hour from order_purchase_timestamp) as hours,
count(distinct order_id) as orders
```

Query results

SAVE RESULTS

JOB INFORMATION RESULTS JSON EXECUTION DETAILS EXECUTION GRAPH PREVIEW

Row	time_of_day	orders
1	Morning	21738
2	Dawn	5242
3	afternoon	44130
4	night	28331

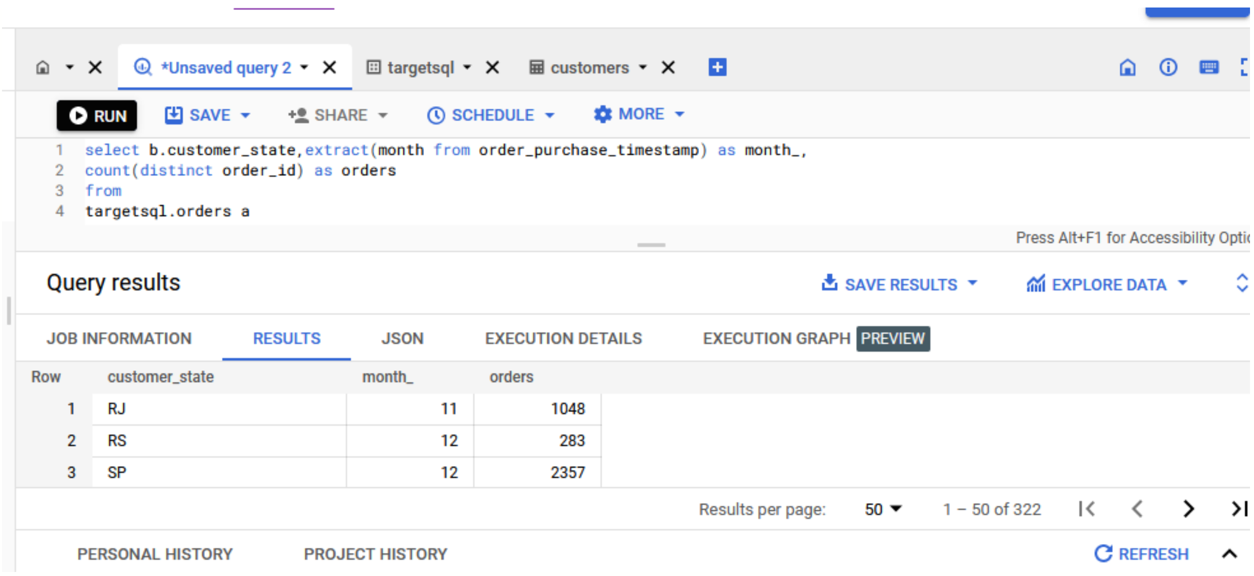
PERSONAL HISTORY PROJECT HISTORY

```
Based on above data, customers buy mostly in afternoon , which is between 12 and 18
```

---

1. Evolution of E-commerce orders in the Brazil region:
    a. Get month on month orders by states

```
select b.customer_state,extract(month from order_purchase_timestamp) as month_,
count(distinct order_id) as orders
from
targetsql.orders a
inner join targetsql.customers b
on a.customer_id=b.customer_id
group by 1
```



```
select b.customer_state,extract(month from order_purchase_timestamp) as month_,
count(distinct order_id) as orders
from
targetsql.orders a
```

Query results

Row	customer_state	month_	orders
1	RJ	11	1048
2	RS	12	283
3	SP	12	2357

Results per page: 50 1 – 50 of 322

customer_state	month_	orders
RJ	11	1048
RS	12	283
SP	12	2357
DF	2	196
PR	11	378
MT	4	92
MA	7	79
AL	7	40
SP	7	4381
MT	7	85
MG	7	1111
MG	5	1190
SP	5	4632
PE	5	174
SP	10	1908
RJ	1	990
SP	1	3351
DF	1	151
RS	1	427
PE	6	140
DF	9	97
SP	2	3357
SE	7	42
RJ	12	783

PR	12	271
RS	3	569
PA	2	83
RJ	3	1302
MG	3	1237
PE	10	87
SP	4	3967
RJ	4	1172
RS	4	488
BA	4	318
CE	1	99
PE	1	113
DF	5	208
GO	5	226
BA	5	368
RJ	5	1321
MG	6	1080
RJ	6	1128
SP	6	4104
CE	4	143
PA	3	109
MT	3	71
PR	1	443
CE	6	121
DF	6	220
SE	6	37
ES	11	170
SC	11	303
BA	3	340
RJ	7	1288
BA	7	405
RJ	8	1307
BA	8	323
MG	2	1063
BA	1	264
RS	9	279
PI	2	46
SP	3	4047
PB	5	47
RS	10	276
RJ	10	725
MG	4	1061
AL	3	40

PR	3	504
RN	6	49
SP	11	3012
PR	5	524
SP	8	4982
PB	6	51
SP	9	1648
GO	7	192
GO	3	199
MG	11	943
PB	11	30
CE	3	126
PI	5	56
CE	11	108
PR	7	523
ES	7	206
PE	4	154
PA	4	107
PI	4	50
CE	9	77
MG	9	511
MA	9	42
MG	1	971
BA	6	307
DF	4	183
RJ	2	1176
RS	2	473
PE	2	146
BA	10	170
GO	6	184
AM	3	14
SC	5	379
DF	8	232
SC	8	365
DF	10	104
MG	12	691
ES	5	228
DF	3	207
RN	1	51
SC	7	356
MG	8	1177
RJ	9	612
MA	4	73

RR	2	7
MA	3	77
RS	8	599
SE	8	43
ES	4	188
PR	10	225
MG	10	600
CE	7	140
ES	12	113
SE	3	43
GO	4	177
RR	9	2
CE	2	101
PR	6	478
AL	4	51
PA	7	96
ES	8	200
PE	8	170
SC	6	321
MT	10	55
SC	3	362
CE	10	74
MS	7	74
GO	10	117
PA	1	82
MA	11	56
MT	11	74
MA	6	59
PR	4	500
BA	12	192
BA	9	170
PB	9	29
MS	12	36
MA	12	41
RS	11	422
PR	2	460
BA	2	273
PE	3	153
DF	7	243
BA	11	250
MA	1	66
SC	2	316
GO	2	176

RS	7	565
SC	1	345
ES	6	204
MT	8	78
DF	12	131
RS	6	526
CE	5	136
PE	7	210
RN	3	52
TO	1	19
PR	9	183
PI	8	43
RR	3	8
PA	11	70
RN	11	44
GO	11	157
AM	5	19
AL	8	34
DF	11	168
MA	8	70
GO	1	164
MT	9	35
PB	7	79
MS	5	74
ES	3	182
SC	4	351
GO	9	88
SE	4	27
MT	1	96
AC	10	6
MT	6	83
PE	12	103
PA	10	58
PB	8	46
PB	4	51
AL	9	20
TO	5	34
PA	6	92
PE	9	76
TO	4	33
GO	8	213
ES	1	159
MA	5	65

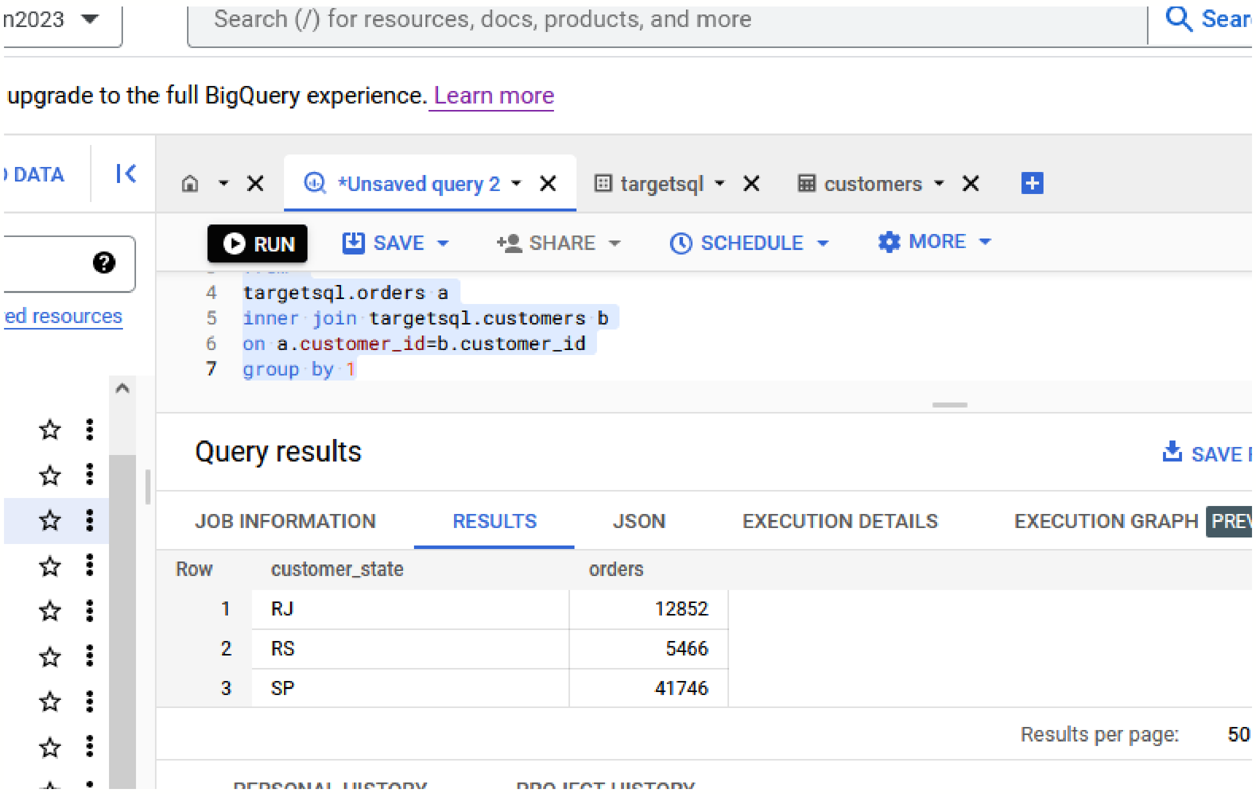
PR	8	556
AL	12	14
PA	8	104
MS	3	79
CE	8	130
SC	10	189
RO	4	20
ES	2	186
SE	11	27
SC	9	157
PI	3	48
RS	5	559
RR	1	2
RO	8	23
PI	10	25
TO	2	28
RO	6	22
AL	10	30
GO	12	127
PB	10	31
PI	9	23
SE	1	24
ES	10	104
MS	4	58
RN	4	42
RO	7	27
RN	7	56
AM	7	23
PI	7	52
RN	5	39
MT	5	104
SE	5	19
AL	5	46
PA	5	75
MA	10	52
RN	10	27
SE	10	25
RO	10	14
TO	10	13
MS	10	34
SC	12	193
SE	12	20
PB	1	33

MS	1	71
PA	12	58
MT	12	50
PI	1	55
CE	12	81
PB	12	37
MS	6	76
AP	6	4
TO	6	26
MS	9	33
TO	9	17
ES	9	93
RN	9	24
SE	9	16
SE	2	27
RO	2	25
AC	1	8
AM	2	16
AL	6	34
PI	6	43
AM	6	8
PI	12	23
AC	11	5
RN	12	30
RO	11	17
PI	11	31
RR	11	2
PB	3	55
TO	3	28
RR	10	4
MS	11	46
AL	11	26
PE	11	126
AM	4	19
AC	8	7
AM	8	9
AL	1	39
RO	1	23
AM	1	12
AP	1	11
AP	4	5
TO	12	14
RO	5	26

AL	2	39
RO	3	29
MS	2	75
TO	11	17
MA	2	67
AP	2	4
PB	2	47
MS	8	59
AP	3	8
AC	4	9
PA	9	41
AM	9	9
RO	9	16
MT	2	84
AC	2	6
TO	7	23
AP	11	4
RR	7	6
AC	12	5
AP	12	4
AC	6	7
AC	9	5
AC	5	10
AC	3	4
TO	8	28
RN	8	40
AM	10	3
RN	2	31
AP	5	11
AP	7	7
AM	11	10
AP	8	5
AC	7	9
RR	4	4
RO	12	11
AM	12	6
AP	9	2
RR	6	8
AP	10	3
RR	5	3

Distribution of customers across the states in Brazil

```
select b.customer_state,
count(distinct order_id) as orders
from
targetsql.orders a
inner join targetsql.customers b
on a.customer_id=b.customer_id
group by 1
```



```
targetsql.orders a
inner join targetsql.customers b
on a.customer_id=b.customer_id
group by 1
```

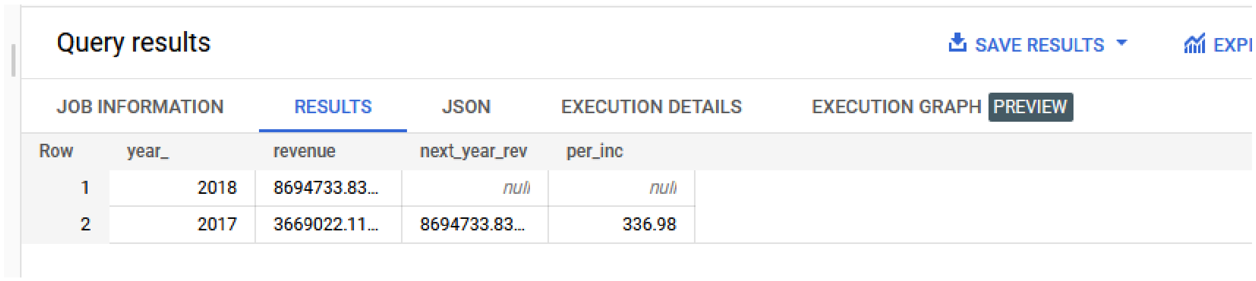
Row	customer_state	orders
1	RJ	12852
2	RS	5466
3	SP	41746

customer_state	orders
RJ	12852
RS	5466
SP	41746
DF	2140
PR	5045
MT	907

MA	747
AL	413
MG	11635
PE	1652
SE	350
PA	975
BA	3380
CE	1336
GO	2020
ES	2033
SC	3637
PI	495
PB	536
RN	485
AM	148
RR	46
MS	715
TO	280
AC	81
RO	253
AP	68

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

```
with base as (
select extract (year from order_purchase_timestamp) year_,sum(payment_value) as revenue
from
targetsql.orders a
inner join
targetsql.payments b
on a.order_id=b.order_id
where extract (month from order_purchase_timestamp) between 1 and 8
group by 1
order by 1
),
base_2 as (
  select *,lead(revenue,1) over (order by year_ asc) as next_year_rev from base
)
select *, round((next_year_rev + revenue)/revenue*100,2) as per_inc from base_2
```

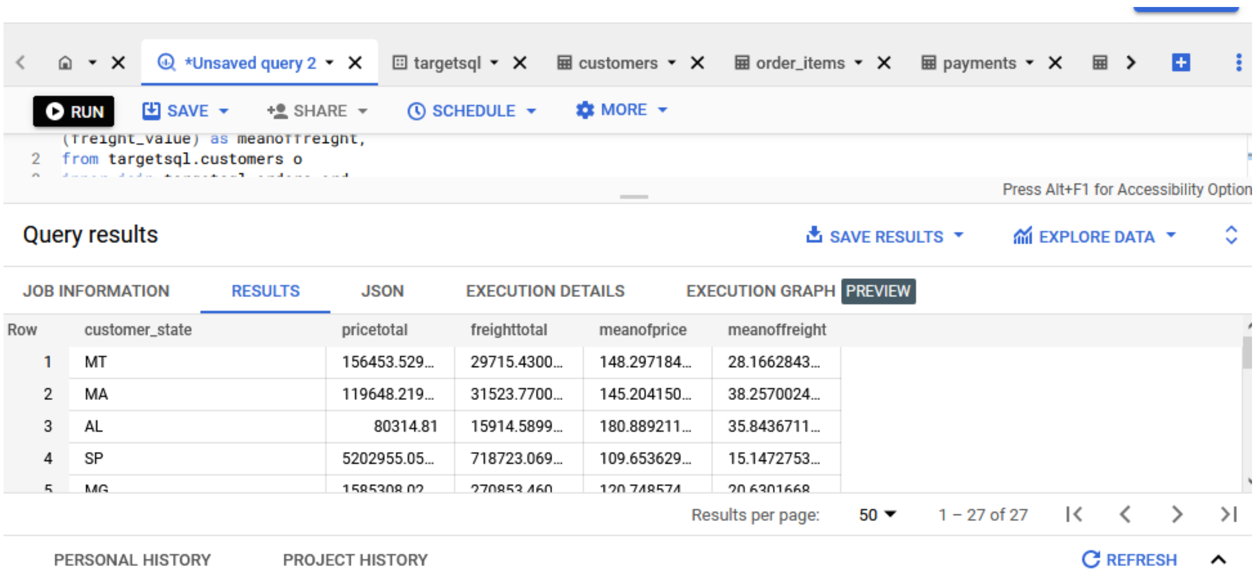


Query results

Row	year_	revenue	next_year_rev	per_inc
1	2018	8694733.83...	null	null
2	2017	3669022.11...	8694733.83...	336.98

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

```
select o.customer_state, sum(price) as pricetotal,sum(freight_value) as freighttotal,avg(price
) as meanofprice,avg(freight_value) as meanoffreight,
from targetsql.customers o
inner join targetsql.orders ord
on o.customer_id=ord.customer_id
inner join targetsql.order_items ot
on ord.order_id=ot.order_id
group by 1
```



```
(freight_value) as meanoffreight,
from targetsql.customers o
```

Query results

Row	customer_state	pricetotal	freighttotal	meanofprice	meanoffreight
1	MT	156453.529...	29715.4300...	148.297184...	28.1662843...
2	MA	119648.219...	31523.7700...	145.204150...	38.2570024...
3	AL	80314.81	15914.5899...	180.889211...	35.8436711...
4	SP	5202955.05...	718723.069...	109.653629...	15.1472753...
5	MG	1585308.02	270853.460	120.748574	20.6301668

Results per page: 50 1 – 27 of 27

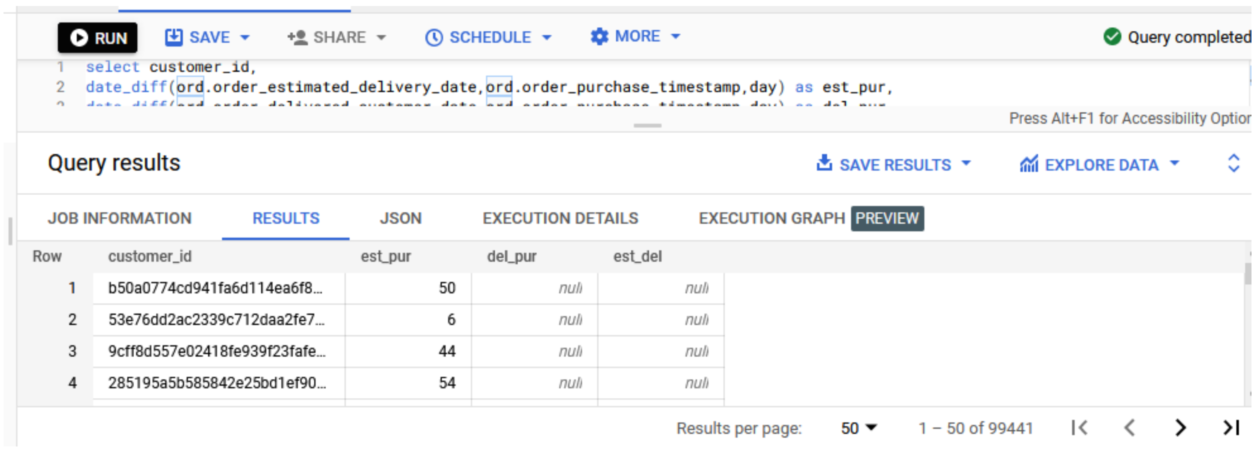
5. Analysis on sales, freight and delivery time

a.Calculate days between purchasing, delivering and estimated delivery

b.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

```
select customer_id,
date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day) as est_pur,
date_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp,day) as del_pur,
date_diff(ord.order_estimated_delivery_date,ord.order_delivered_customer_date,day) as est_del
from targetsql.orders as ord
```



```
select customer_id,
date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day) as est_pur,
```

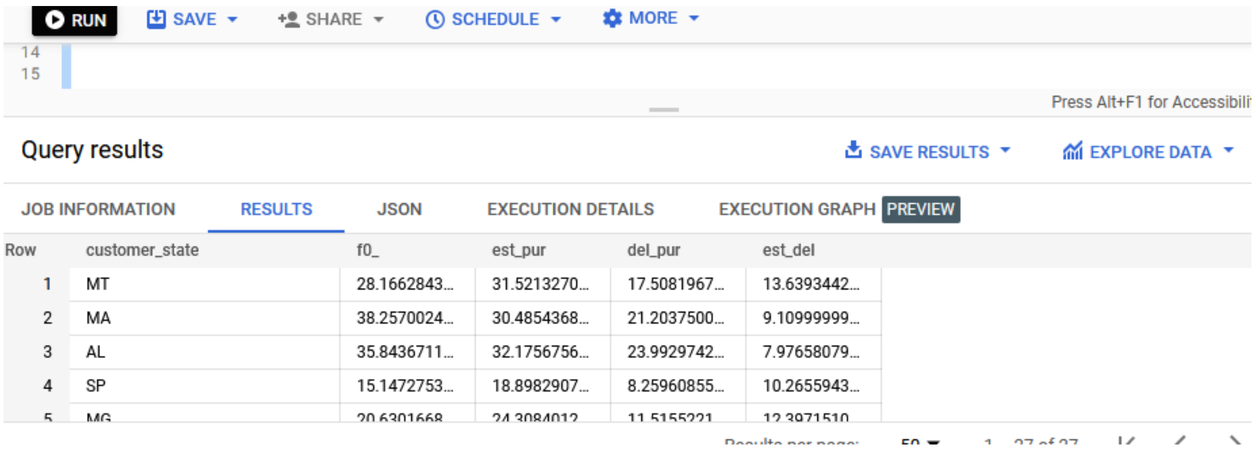
Query results

Row	customer_id	est_pur	del_pur	est_del
1	b50a0774cd941fa6d114ea6f8...	50	null	null
2	53e76dd2ac2339c712daa2fe7...	6	null	null
3	9cff8d557e02418fe939f23fafe...	44	null	null
4	285195a5b585842e25bd1ef90...	54	null	null

Results per page: 50 1 – 50 of 99441

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

```
select customer_state,avg(freight_value),
avg(date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day)) as est_pur,
avg(date_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp,day)) as del_pur,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_delivered_customer_date,day)) as est
_del
from targetsql.orders as ord
inner join targetsql.customers as cs
on ord.customer_id=cs.customer_id
inner join targetsql.order_items as ot
on ot.order_id=ord.order_id
group by customer_state
```



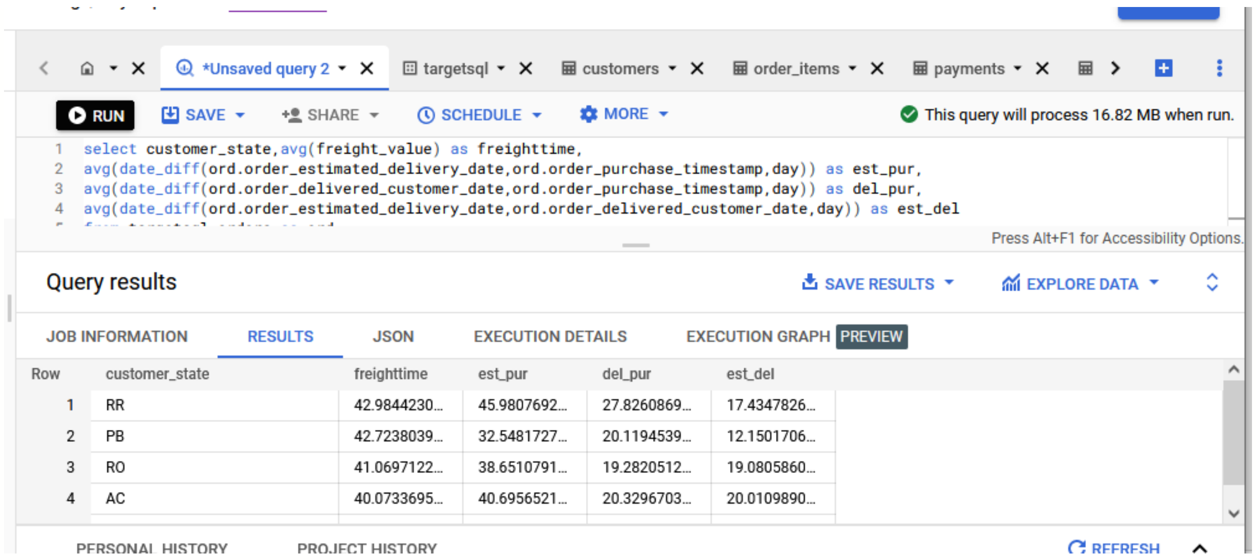
Query results

Row	customer_state	f0_	est_pur	del_pur	est_del
1	MT	28.1662843...	31.5213270...	17.5081967...	13.6393442...
2	MA	38.2570024...	30.4854368...	21.2037500...	9.10999999...
3	AL	35.8436711...	32.1756756...	23.9929742...	7.97658079...
4	SP	15.1472753...	18.8982907...	8.25960855...	10.2655943...
5	MG	20.6301668	24.3084012	11.5155221	12.3971510

Results per page: 50 1 – 27 of 27

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

```
select customer_state,avg(freight_value) as freighttime,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day)) as est_pur,
avg(date_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp,day)) as del_pur,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_delivered_customer_date,day)) as est
_del
from targetsql.orders as ord
inner join targetsql.customers as cs
on ord.customer_id=cs.customer_id
inner join targetsql.order_items as ot
on ot.order_id=ord.order_id
group by customer_state
order by freighttime desc limit 5
```



This query will process 16.82 MB when run.

```
select customer_state,avg(freight_value) as freighttime,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day)) as est_pur,
avg(date_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp,day)) as del_pur,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_delivered_customer_date,day)) as est_del
```

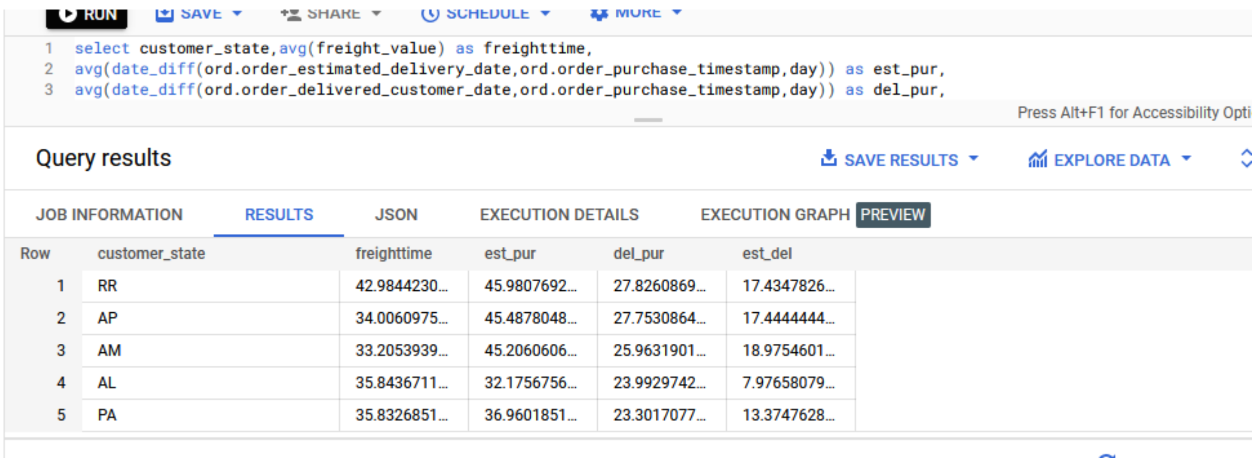
Press Alt+F1 for Accessibility Options.

Query results

Row	customer_state	freighttime	est_pur	del_pur	est_del
1	RR	42.9844230...	45.9807692...	27.8260869...	17.4347826...
2	PB	42.7238039...	32.5481727...	20.1194539...	12.1501706...
3	RO	41.0697122...	38.6510791...	19.2820512...	19.0805860...
4	AC	40.0733695...	40.6956521...	20.3296703...	20.0109890...

e.Top 5 states with highest/lowest average time to delivery

```
select customer_state,avg(freight_value) as freighttime,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day)) as est_pur,
avg(date_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp,day)) as del_pur,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_delivered_customer_date,day)) as est
_del
from targetsql.orders as ord
inner join targetsql.customers as cs
on ord.customer_id=cs.customer_id
inner join targetsql.order_items as ot
on ot.order_id=ord.order_id
group by customer_state
order by del_pur desc limit 5
```



```
select customer_state,avg(freight_value) as freighttime,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day)) as est_pur,
avg(date_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp,day)) as del_pur,
```

Press Alt+F1 for Accessibility Opti

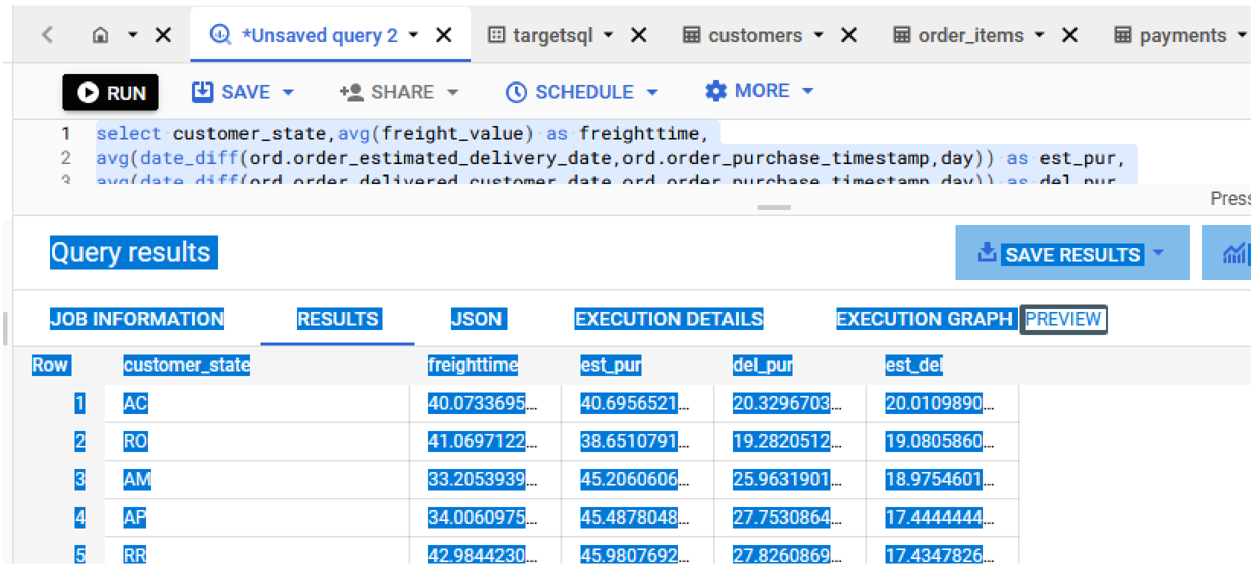
Query results

Row	customer_state	freighttime	est_pur	del_pur	est_del
1	RR	42.9844230...	45.9807692...	27.8260869...	17.4347826...
2	AP	34.0060975...	45.4878048...	27.7530864...	17.4444444...
3	AM	33.2053939...	45.2060606...	25.9631901...	18.9754601...
4	AL	35.8436711...	32.1756756...	23.9929742...	7.97658079...
5	PA	35.8326851...	36.9601851...	23.3017077...	13.3747628...

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

```
select customer_state,avg(freight_value) as freighttime,

avg(date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day)) as est_pur,
avg(date_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp,day)) as del_pur,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_delivered_customer_date,day)) as est
_del
from targetsql.orders as ord
inner join targetsql.customers as cs
on ord.customer_id=cs.customer_id
inner join targetsql.order_items as ot
on ot.order_id=ord.order_id
group by customer_state
order by est_del desc limit 5
```



```
select customer_state,avg(freight_value) as freighttime,
avg(date_diff(ord.order_estimated_delivery_date,ord.order_purchase_timestamp,day)) as est_pur,
```

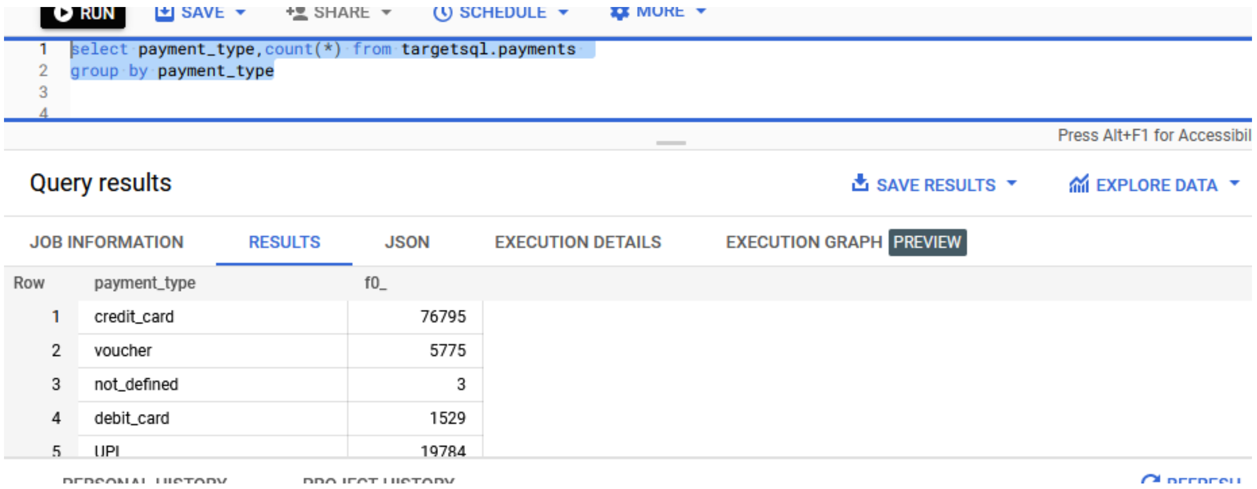
Query results

Row	customer_state	freighttime	est_pur	del_pur	est_del
1	AC	40.0733695...	40.6956521...	20.3296703...	20.0109890...
2	RO	41.0697122...	38.6510791...	19.2820512...	19.0805860...
3	AM	33.2053939...	45.2060606...	25.9631901...	18.9754601...
4	AP	34.0060975...	45.4878048...	27.7530864...	17.4444444...
5	RR	42.9844230...	45.9807692...	27.8260869...	17.4347826...

6. Payment type analysis:

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

```
select payment_type,count(*) from targetsql.payments
group by payment_type
```

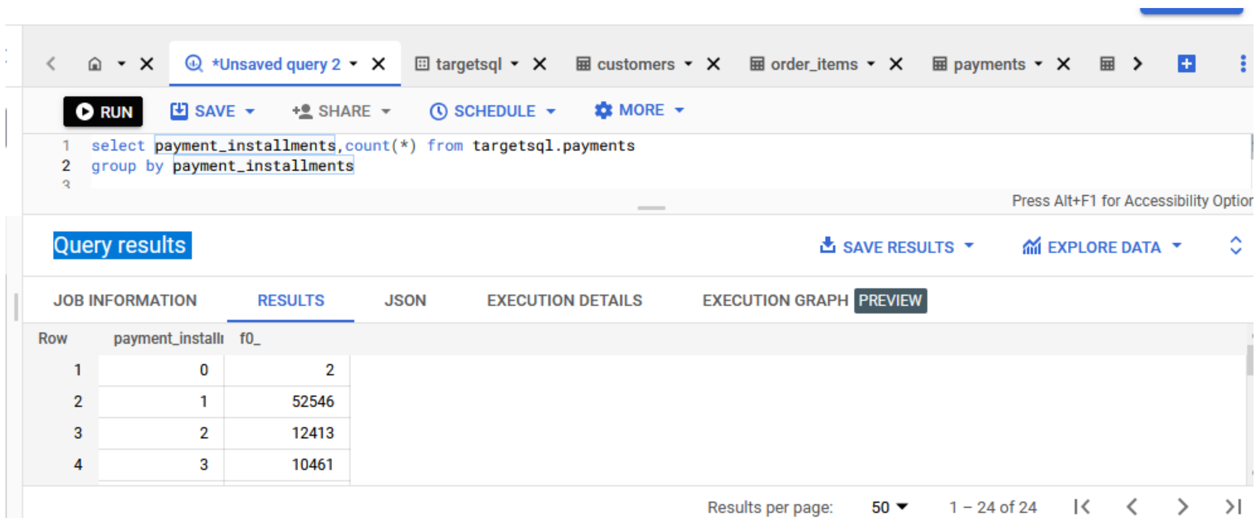


```
select payment_type,count(*) from targetsql.payments
group by payment_type
```

Query results

Row	payment_type	f0_
1	credit_card	76795
2	voucher	5775
3	not_defined	3
4	debit_card	1529
5	UPI	19784

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



```
select payment_installments,count(*) from targetsql.payments
group by payment_installments
```

Query results

Row	payment_installı	f0_
1	0	2
2	1	52546
3	2	12413
4	3	10461

```
select payment_installments,count(*) from targetsql.payments
group by payment_installments
```

7. Actionable Insights (10 points)

a) It is observed that for all the states, the estimated delivery date is exceeding the actual delivery date by 20 to 7 days. Which means that orders are getting delivered quicker than expected.

b) Most of the sales, which is 69 percent of sales, are happening when there is a 1 installment option provided on the item.

c) 67 percent of sales are happening from states 'SP','RJ','MG', which means many customers from these states are showing interest to buy from our website, we are delivering to these customers between 8 to 14 days. These three states are among the top 6 states whose delivery times is lower. Which means customers prefer to buy more if the delivery time is lower.
d) 44 percent of sales are happening in afternoon
e) 13 percent of sales happened in November, December months together

8. Recommendations:

a) We need to reset the estimated delivery dates to the lesser timelines based on the recent actual delivery dates data which is available in our database, so that more customers will be willing to buy items from the website due to the lesser delivery dates.
b) We need to cover more items on the website which can be bought using 1 installment
c) These are the three states 'BA','RS','SC' in which the freight value is high even though delivery date is more than expected. If we can decrease the delivery time in these states, there is more scope for increasing the sales in these states
d) Because more sales are happening in afternoon, we can reduce our support staff, warehouse staff in remaining part of day and increase them in afternoon shift, so that we can cut down unnecessary costs
e) Because more sales are happening in November, December months, we can reduce our support staff, warehouse staff in remaining part of year and hire temporary associates for these two months, so that we can cut down costs