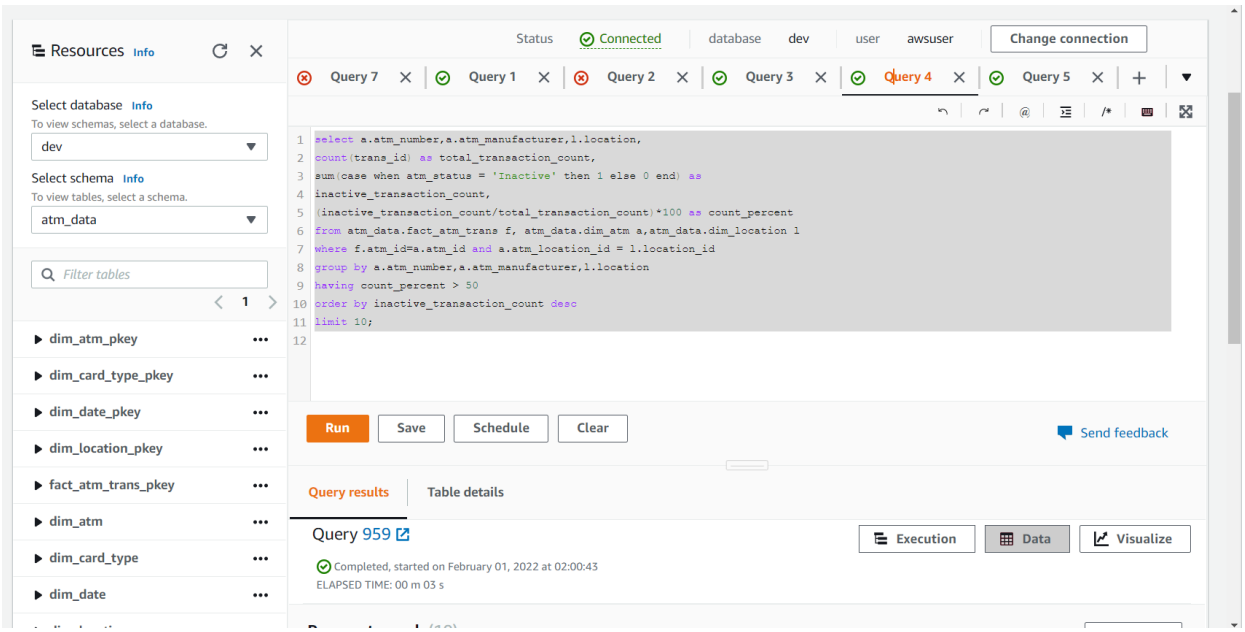
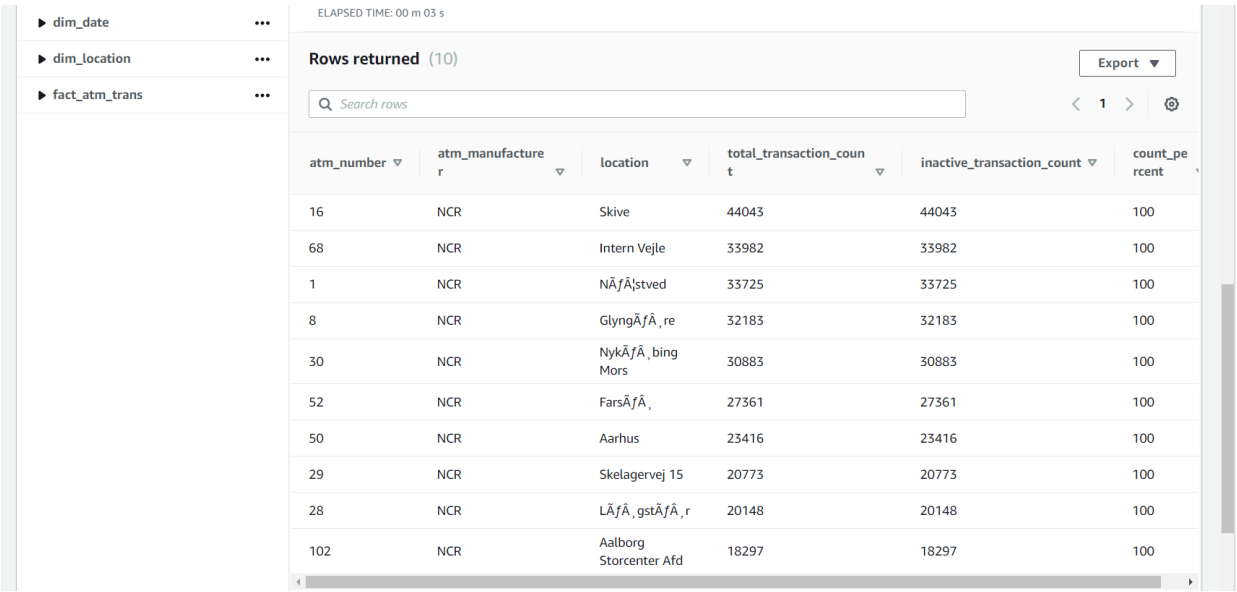
Solving analytical queries on Redshift Cluster

Here, you have to write the query used for solving the question and the screenshots of the table which is outputted after the query is run on the AWS Redshift Query editor UI.

1. **Top 10 ATMs where most transactions are in the 'inactive' state**

```
select a.atm_number,a.atm_manufacturer,l.location,
count(trans_id) as total_transaction_count,
sum(case when atm_status = 'Inactive' then 1 else 0 end) as
inactive_transaction_count,
(inactive_transaction_count/total_transaction_count)*100 as count_percent
from atm_data.fact_atm_trans f, atm_data.dim_atm a,atm_data.dim_location l
where f.atm_id=a.atm_id and a.atm_location_id = l.location_id
group by a.atm_number,a.atm_manufacturer,l.location
having count_percent > 50
order by inactive_transaction_count desc
limit 10;
```

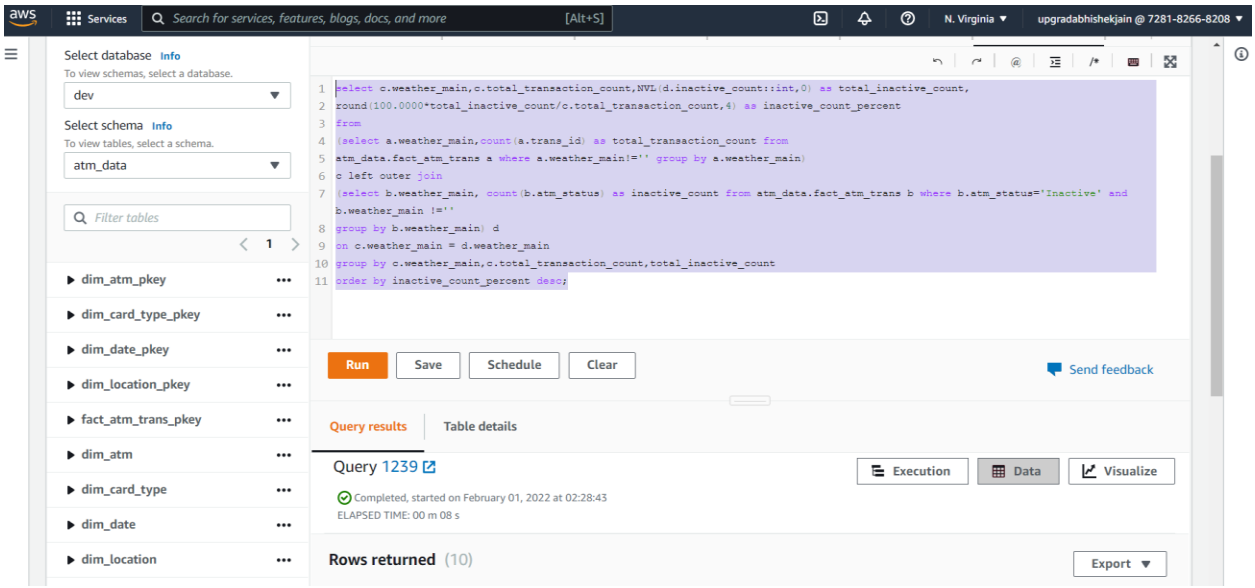


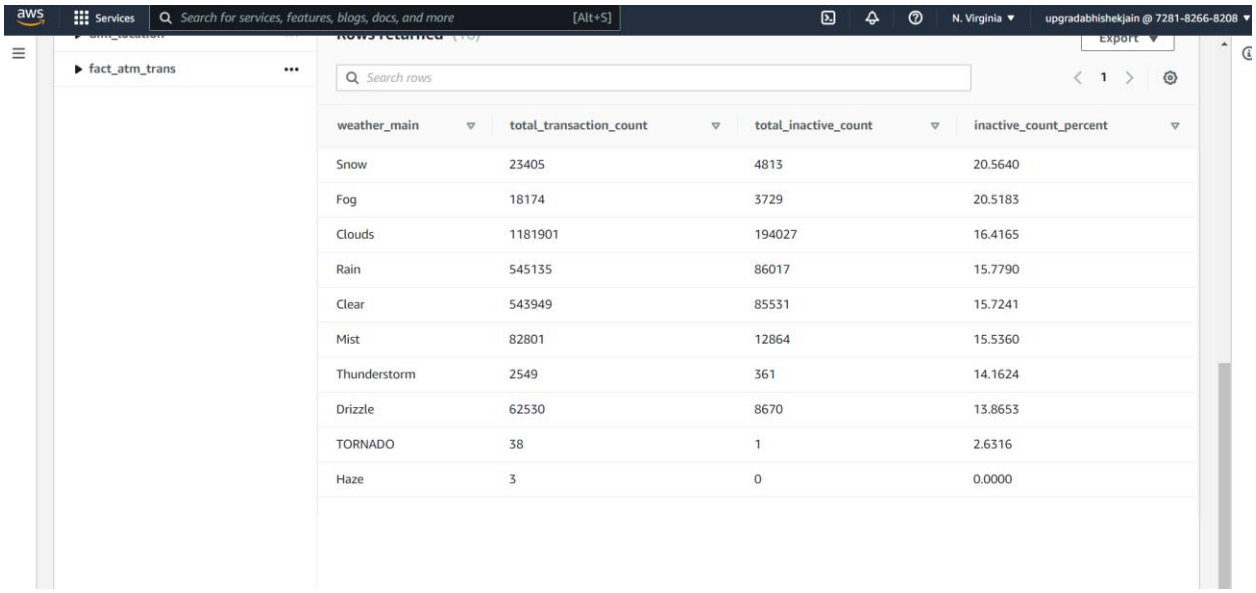


atm_number	atm_manufacturer	location	total_transaction_count	inactive_transaction_count	count_percent
16	NCR	Skive	44043	44043	100
68	NCR	Intern Vejle	33982	33982	100
1	NCR	NÃƒÂ¦stved	33725	33725	100
8	NCR	GlyngÃƒÂ¸re	32183	32183	100
30	NCR	NykÃƒÂ¸bing Mors	30883	30883	100
52	NCR	FarsÃƒÂ¸	27361	27361	100
50	NCR	Aarhus	23416	23416	100
29	NCR	Skelagervej 15	20773	20773	100
28	NCR	LÃƒÂ¸gstÃƒÂ¸r	20148	20148	100
102	NCR	Aalborg Storcenter Afd	18297	18297	100

2. **Number of ATM failures corresponding to the different weather conditions recorded at the time of the transactions**

```
select c.weather_main,c.total_transaction_count,NVL(d.inactive_count::int,0) as
total_inactive_count,
round(100.0000*total_inactive_count/c.total_transaction_count,4) as inactive_count_percent
from
(select a.weather_main,count(a.trans_id) as total_transaction_count from
atm_data.fact_atm_trans a where a.weather_main!='' group by a.weather_main)
c left outer join
(select b.weather_main, count(b.atm_status) as inactive_count from atm_data.fact_atm_trans b
where b.atm_status='Inactive' and b.weather_main !=''
group by b.weather_main) d
on c.weather_main = d.weather_main
group by c.weather_main,c.total_transaction_count,total_inactive_count
order by inactive_count_percent desc;
```

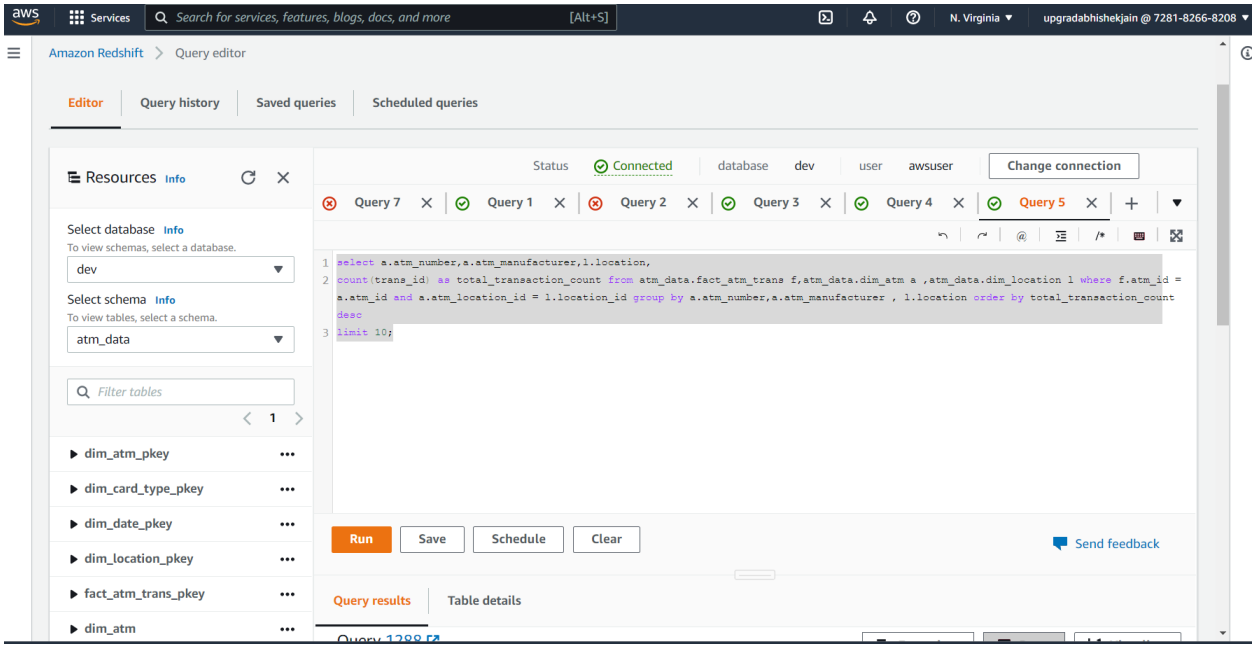


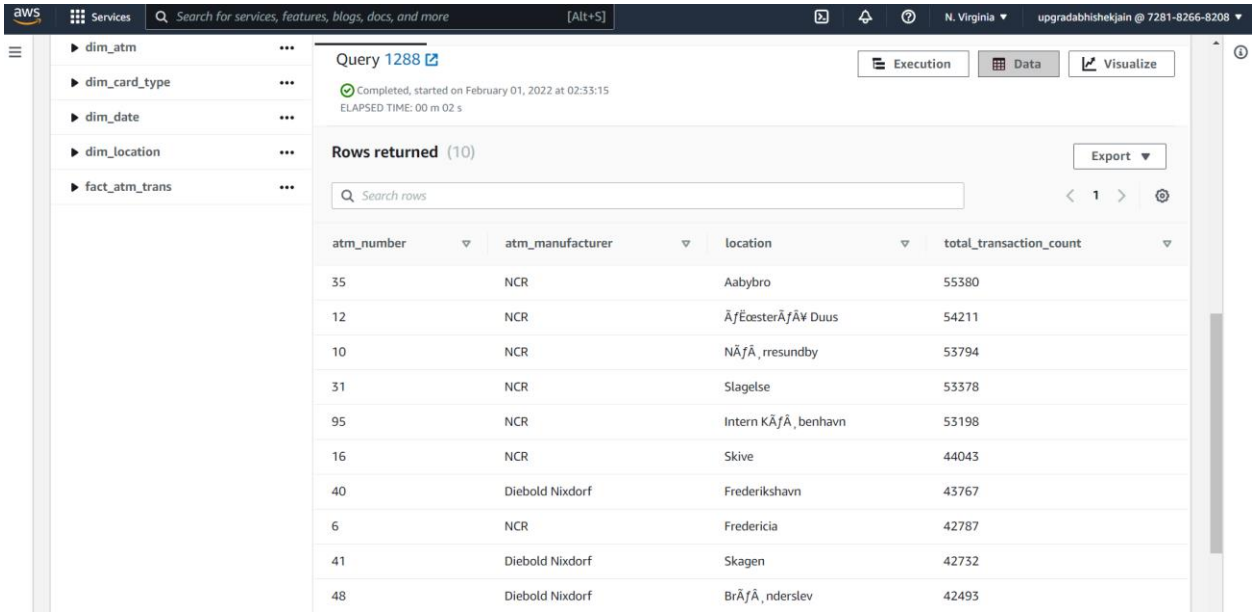


weather_main	total_transaction_count	total_inactive_count	inactive_count_percent
Snow	23405	4813	20.5640
Fog	18174	3729	20.5183
Clouds	1181901	194027	16.4165
Rain	545135	86017	15.7790
Clear	543949	85531	15.7241
Mist	82801	12864	15.5360
Thunderstorm	2549	361	14.1624
Drizzle	62530	8670	13.8653
TORNADO	38	1	2.6316
Haze	3	0	0.0000

3. Top 10 ATMs with the most number of transactions throughout the year

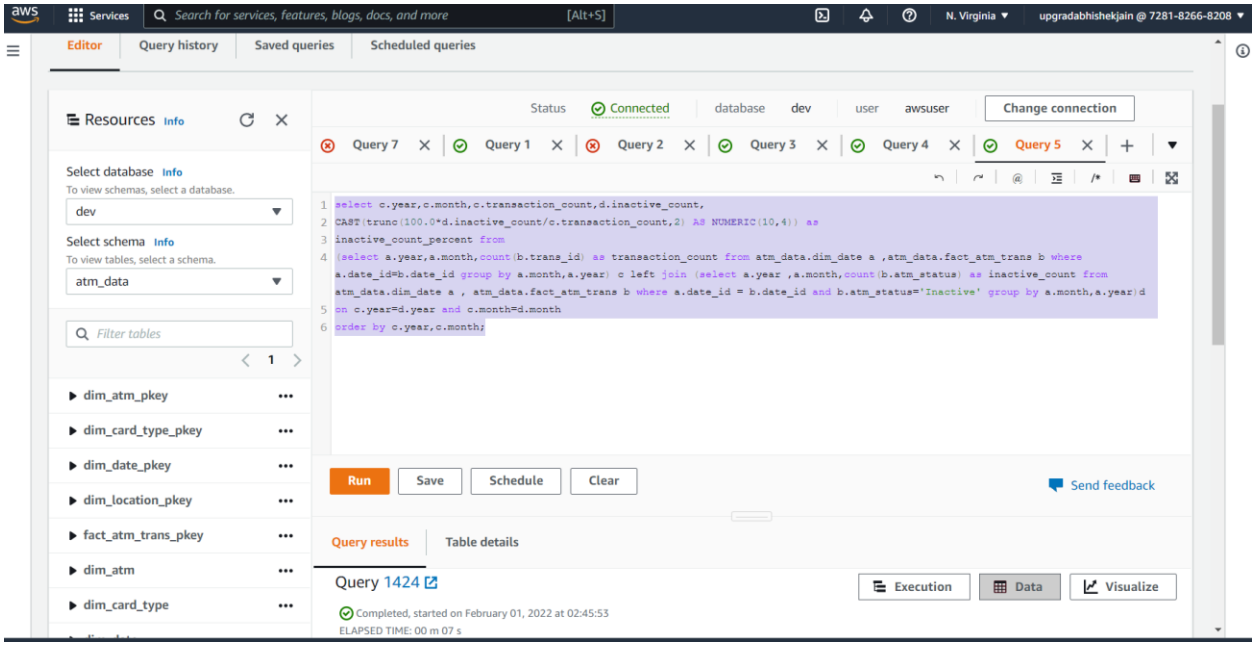
```
select a.atm_number,a.atm_manufacturer,l.location,
count(trans_id) as total_transaction_count from atm_data.fact_atm_trans f,atm_data.dim_atm a
,atm_data.dim_location l where f.atm_id = a.atm_id and a.atm_location_id = l.location_id group
by a.atm_number,a.atm_manufacturer , l.location order by total_transaction_count desc
limit 10;
```

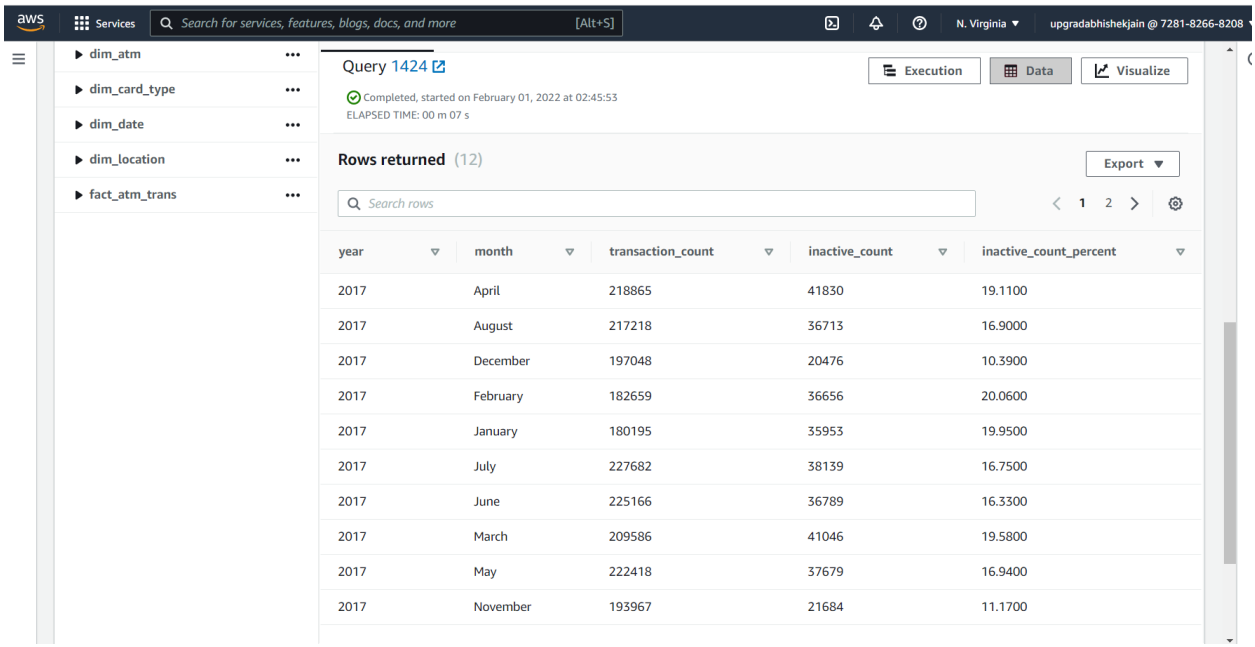




4. **Number of overall ATM transactions going inactive per month for each month**

```
select c.year,c.month,c.transaction_count,d.inactive_count,
CAST(trunc(100.0*d.inactive_count/c.transaction_count,2) AS NUMERIC(10,4)) as
inactive_count_percent from
(select a.year,a.month,count(b.trans_id) as transaction_count from atm_data.dim_date a
,atm_data.fact_atm_trans b where a.date_id=b.date_id group by a.month,a.year) c left join
(select a.year ,a.month,count(b.atm_status) as inactive_count from atm_data.dim_date a ,
atm_data.fact_atm_trans b where a.date_id = b.date_id and b.atm_status='Inactive' group by
a.month,a.year)d
on c.year=d.year and c.month=d.month
order by c.year,c.month;
```

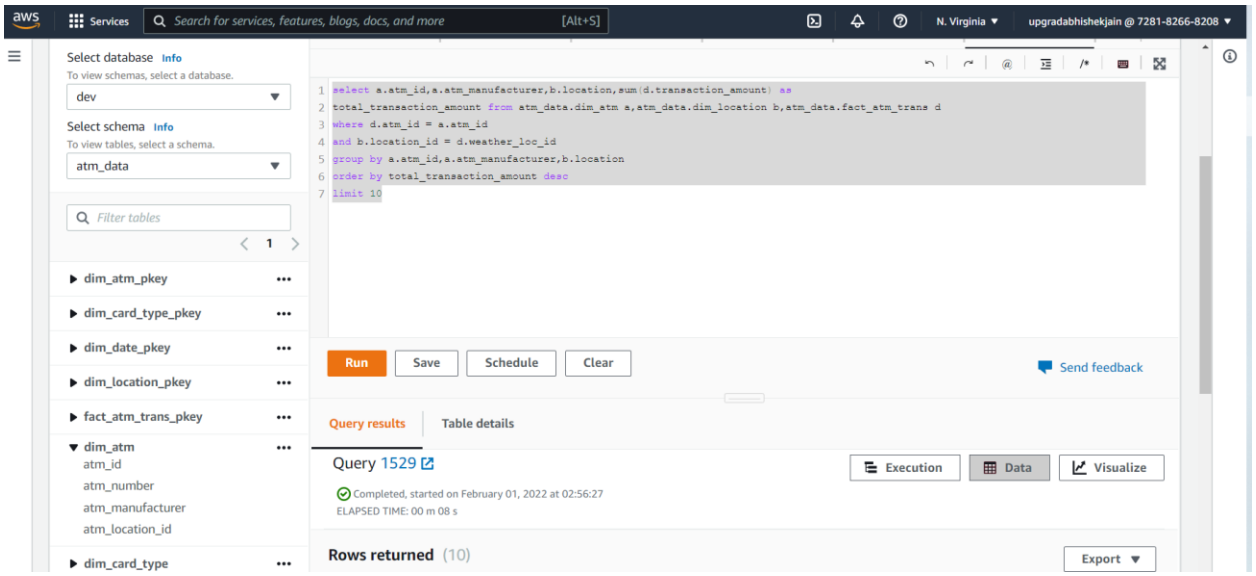


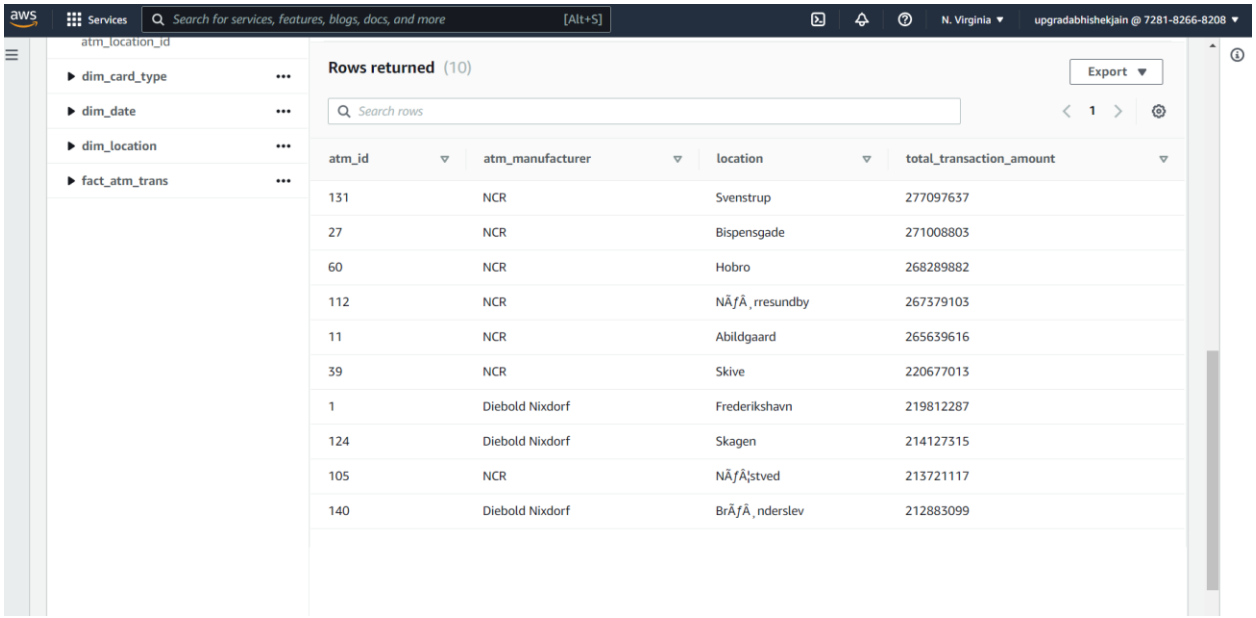


year	month	transaction_count	inactive_count	inactive_count_percent
2017	April	218865	41830	19.1100
2017	August	217218	36713	16.9000
2017	December	197048	20476	10.3900
2017	February	182659	36656	20.0600
2017	January	180195	35953	19.9500
2017	July	227682	38139	16.7500
2017	June	225166	36789	16.3300
2017	March	209586	41046	19.5800
2017	May	222418	37679	16.9400
2017	November	193967	21684	11.1700

5. Top 10 ATMs with the highest total withdrawn amount throughout the year

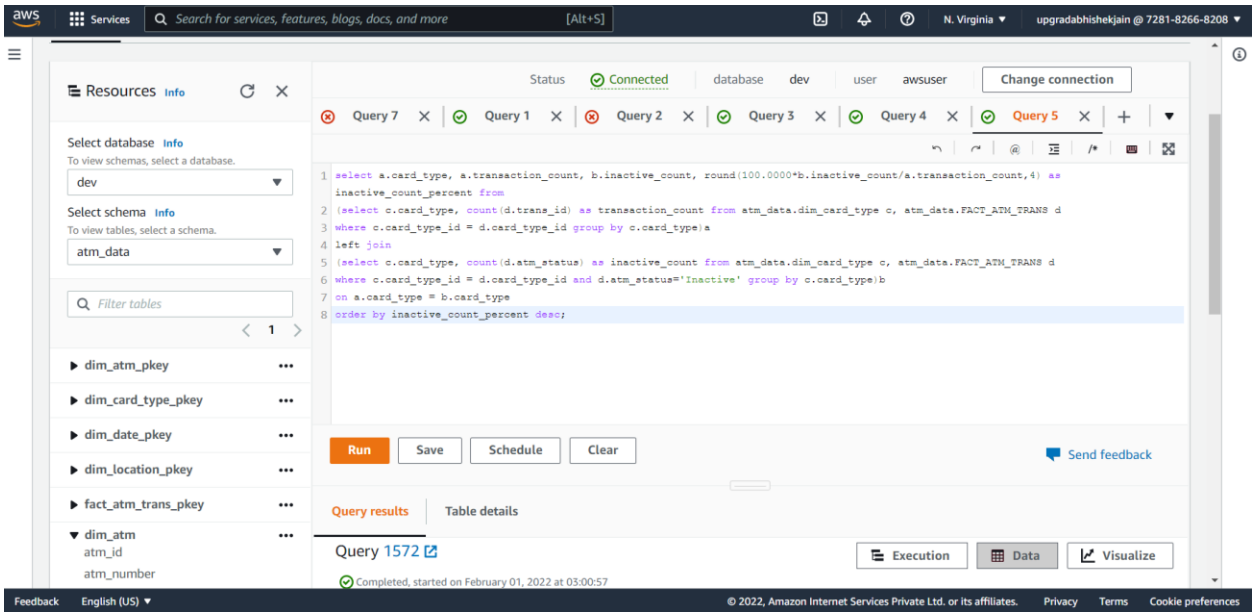
```
select a.atm_id,a.atm_manufacturer,b.location,sum(d.transaction_amount) as
total_transaction_amount from atm_data.dim_atm a,atm_data.dim_location
b,atm_data.fact_atm_trans d
where d.atm_id = a.atm_id
and b.location_id = d.weather_loc_id
group by a.atm_id,a.manufacturer,b.atm_location
order by total_transaction_amount desc
limit 10
```

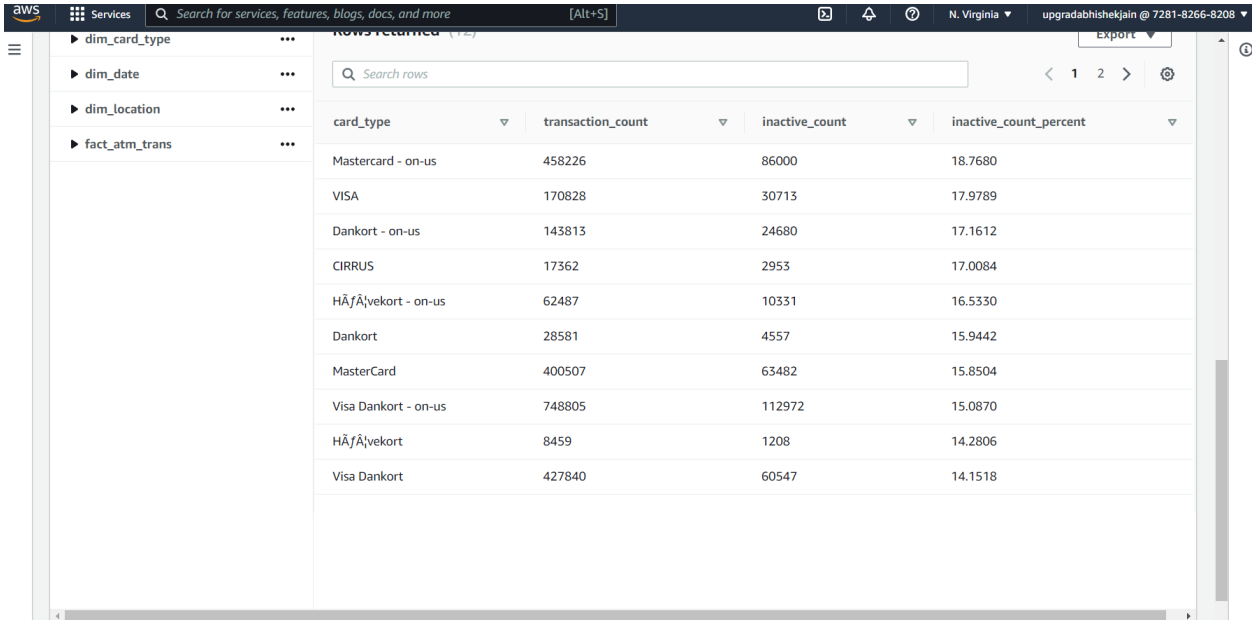




6. Number of failed ATM transactions across various card types

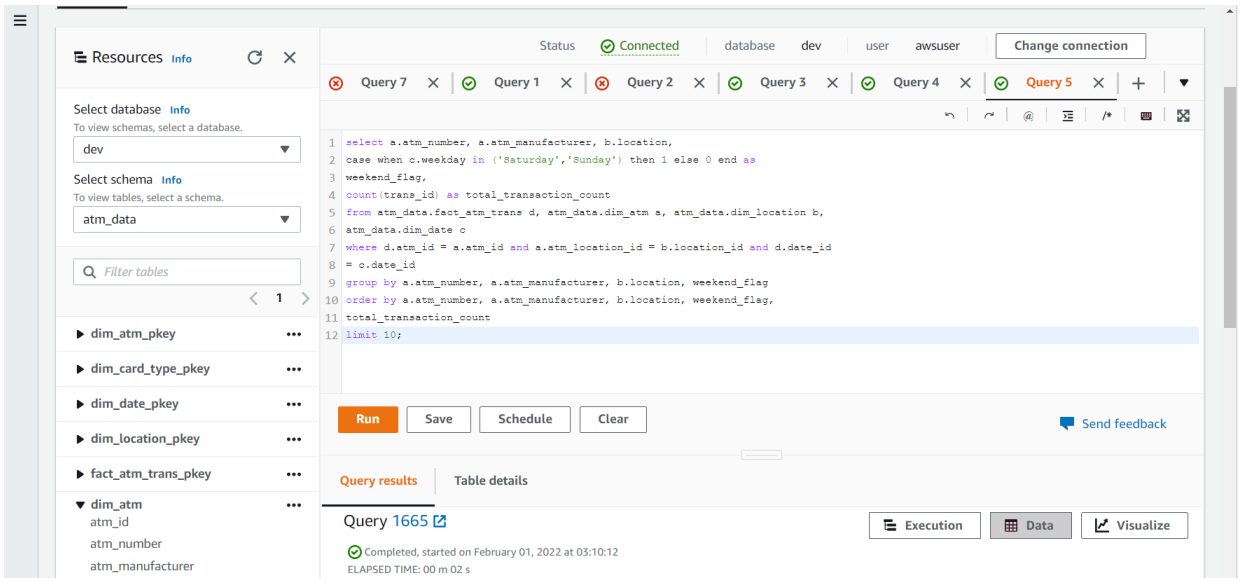
```
select a.card_type, a.transaction_count, b.inactive_count,
round(100.0000*b.inactive_count/a.transaction_count,4) as inactive_count_percent from
(select c.card_type, count(d.trans_id) as transaction_count from atm_data.dim_card_type c,
atm_data.FACT_ATM_TRANS d
where c.card_type_id = d.card_type_id group by c.card_type)a
left join
(select c.card_type, count(d.atm_status) as inactive_count from atm_data.dim_card_type c,
atm_data.FACT_ATM_TRANS d
where c.card_type_id = d.card_type_id and d.atm_status='Inactive' group by c.card_type)b
on a.card_type = b.card_type
order by inactive_count_percent desc;
```

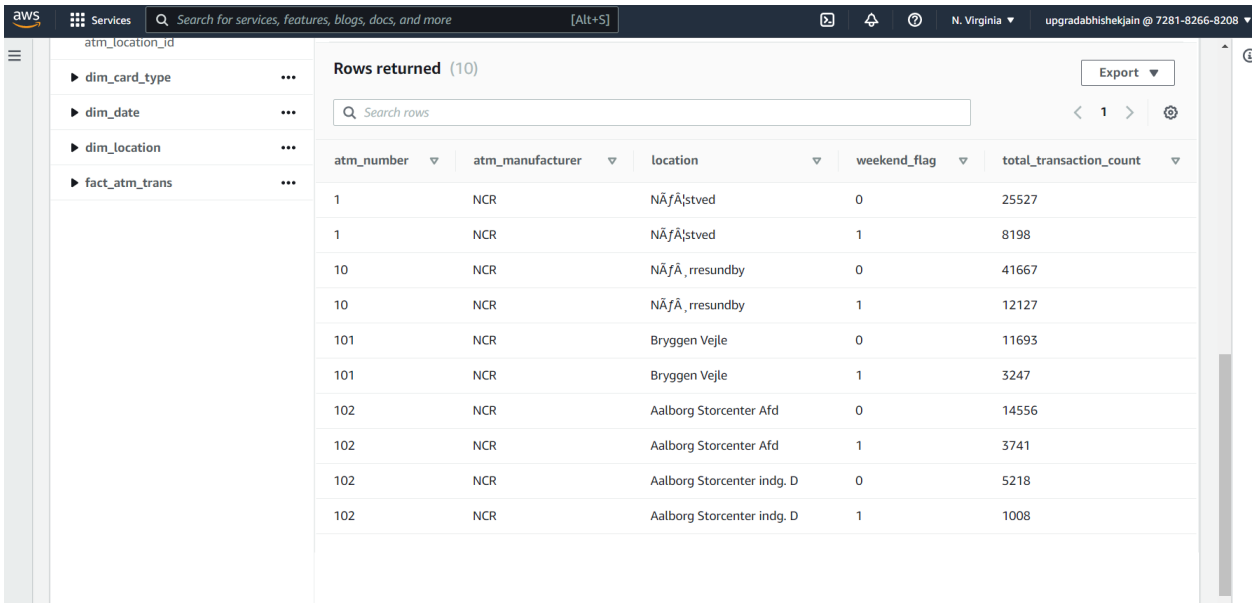




7. **Number of transactions happening on an ATM on weekdays and on weekends throughout the year. Order this by the ATM_number, ATM_manufacturer, location, weekend_flag and then total_transaction_count**

```
select a.atm_number, a.atm_manufacturer, b.location,
case when c.weekday in ('Saturday','Sunday') then 1 else 0 end as
weekend_flag,
count(trans_id) as total_transaction_count
from atm_data.fact_atm_trans d, atm_data.dim_atm a, atm_data.dim_location b,
atm_data.dim_date c
where d.atm_id = a.atm_id and a.atm_location_id = b.location_id and d.date_id
= c.date_id
group by a.atm_number, a.atm_manufacturer, b.location, weekend_flag
order by a.atm_number, a.atm_manufacturer, b.location, weekend_flag,
total_transaction_count
limit 10;
```





8. Most active day in each ATMs from location "Vejgaard"

```
SELECT atm_id,
atm_manufacturer,
location,
weekday,
total_transaction_count
FROM (
select atm_id,
atm_manufacturer,
location,
weekday,
total_transaction_count,
max(total_transaction_count) over (partition by atm_id) as max_version
from (SELECT a.atm_id, a.atm_manufacturer, b.location, c.weekday,
count(d.trans_id) as total_transaction_count
from atm_data.dim_atm a, atm_data.dim_location b, atm_data.dim_date c,
atm_data.FACT_ATM_TRANS d
where d.atm_id = a.atm_id
and b.location_id = d.weather_loc_id
and b.location = 'Vejgaard'
and c.date_id = d.date_id
group by a.atm_id, a.atm_manufacturer, b.location, c.weekday) c
) t
where total_transaction_count = max_version;
```