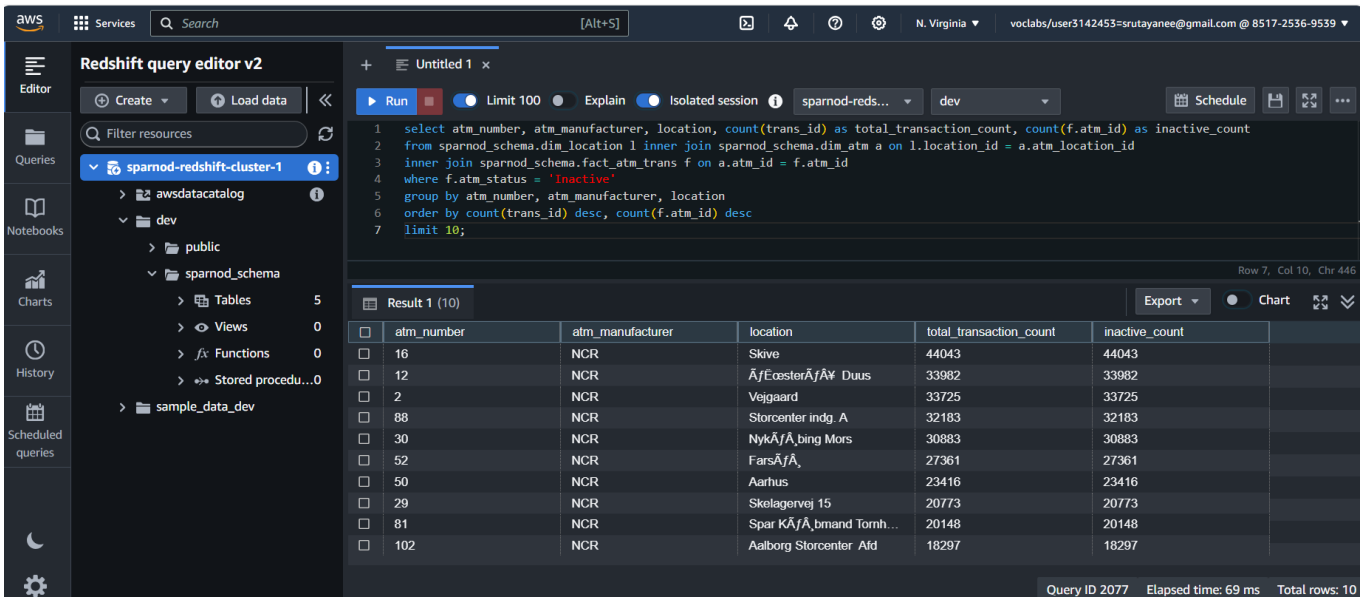
Queries used for solving the Analytical queries in the Redshift cluster:

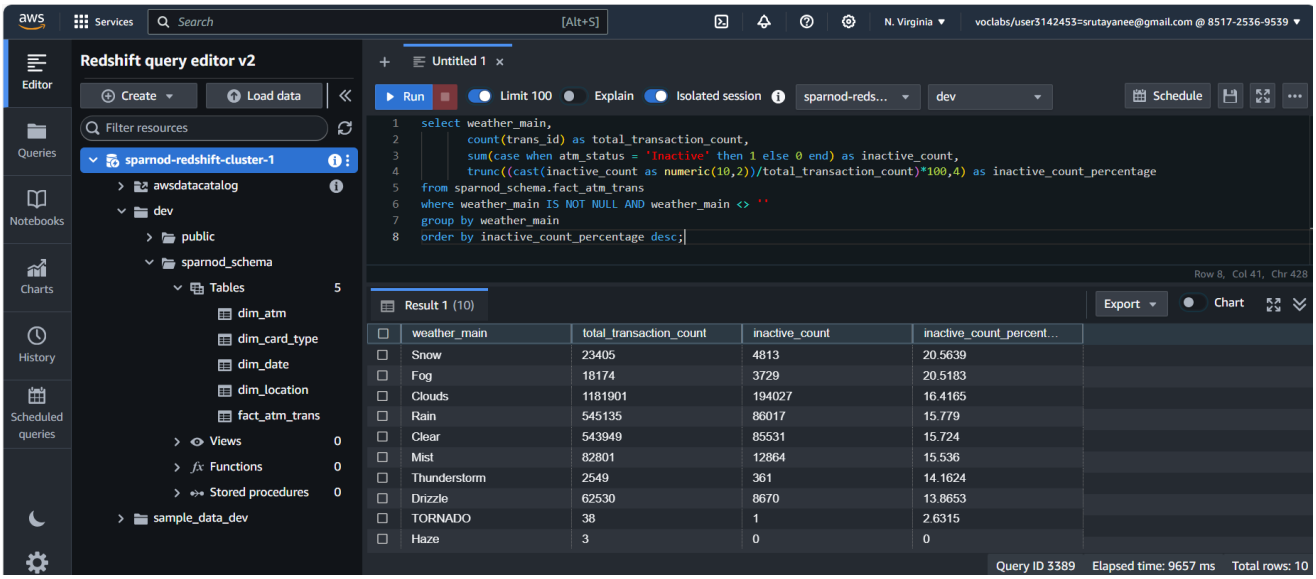
1. Top 10 ATMs where most transactions are in the ’inactive’ state.

```
select atm_number, atm_manufacturer, location, count(trans_id) as total_transaction_count,
count(f.atm_id) as inactive_count

from sparnod_schema.dim_location l inner join sparnod_schema.dim_atm a on l.location_id
= a.atm_location_id

inner join sparnod_schema.fact_atm_trans f on a.atm_id = f.atm_id

where f.atm_status = 'Inactive'

group by atm_number, atm_manufacturer, location

order by count(trans_id) desc, count(f.atm_id) desc

limit 10;
```



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

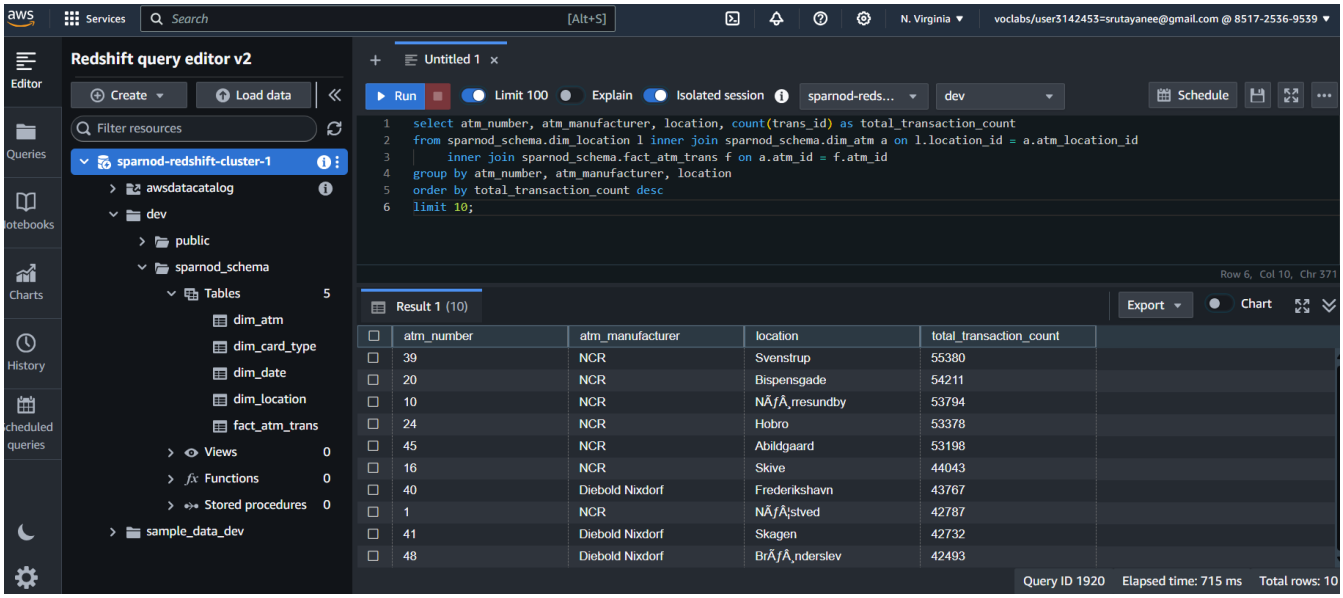
```
select weather_main,
    count(trans_id) as total_transaction_count,
    sum(case when atm_status = 'Inactive' then 1 else 0 end) as inactive_count,
    trunc((cast(inactive_count as numeric(10,2))/total_transaction_count)*100,4) as inactive_count_percentage
from sparnod_schema.fact_atm_trans
where weather_main IS NOT NULL AND weather_main <> ''
group by weather_main
order by inactive_count_percentage desc;
```



weather_main	total_transaction_count	inactive_count	inactive_count_percent...
Snow	23405	4813	20.5639
Fog	18174	3729	20.5183
Clouds	1181901	194027	16.4165
Rain	545135	86017	15.779
Clear	543949	85531	15.724
Mist	82801	12864	15.536
Thunderstorm	2549	361	14.1624
Drizzle	62530	8670	13.8653
TORNADO	38	1	2.6315
Haze	3	0	0

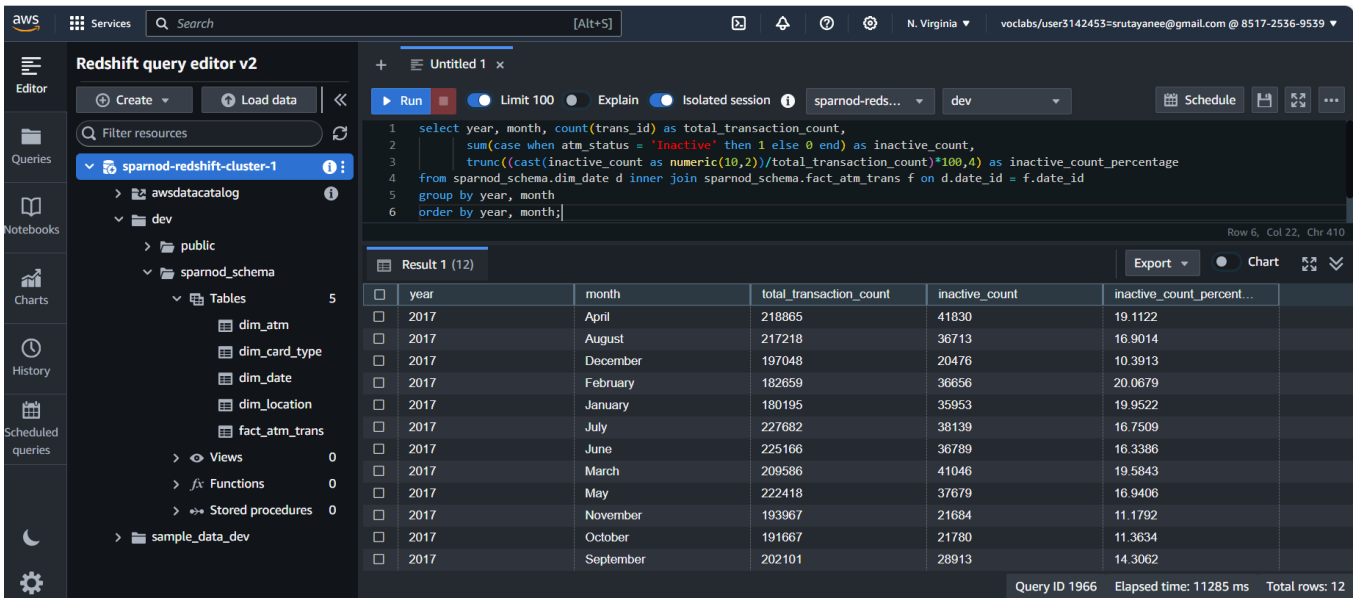
3. Top 10 ATMs with the most number of transactions throughout the year

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



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

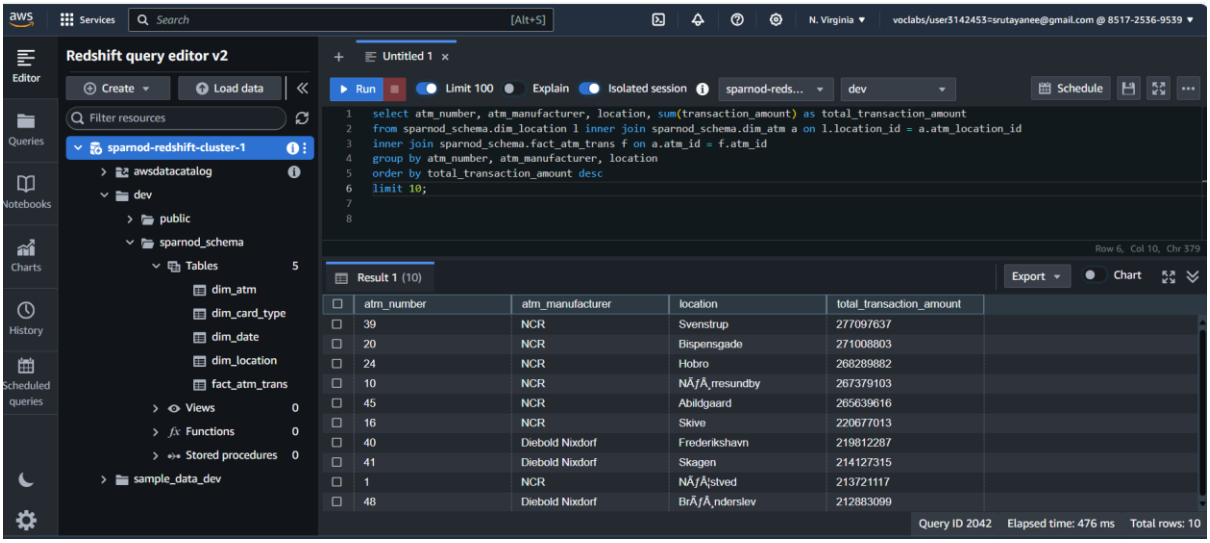
```
select year, month, count(trans_id) as total_transaction_count,
    sum(case when atm_status = 'Inactive' then 1 else 0 end) as inactive_count,
    trunc((cast(inactive_count as numeric(10,2))/total_transaction_count)*100,4) as inactive_count_percentage
from sparnod_schema.dim_date d inner join sparnod_schema.fact_atm_trans f on d.date_id = f.date_id
group by year, month
order by year, month;
```



year	month	total_transaction_count	inactive_count	inactive_count_percent...
2017	April	218865	41830	19.1122
2017	August	217218	36713	16.9014
2017	December	197048	20476	10.3913
2017	February	182659	36656	20.0679
2017	January	180195	35953	19.9522
2017	July	227682	38139	16.7509
2017	June	225166	36789	16.3386
2017	March	209586	41046	19.5843
2017	May	222418	37679	16.9406
2017	November	193967	21684	11.1792
2017	October	191667	21780	11.3634
2017	September	202101	28913	14.3062

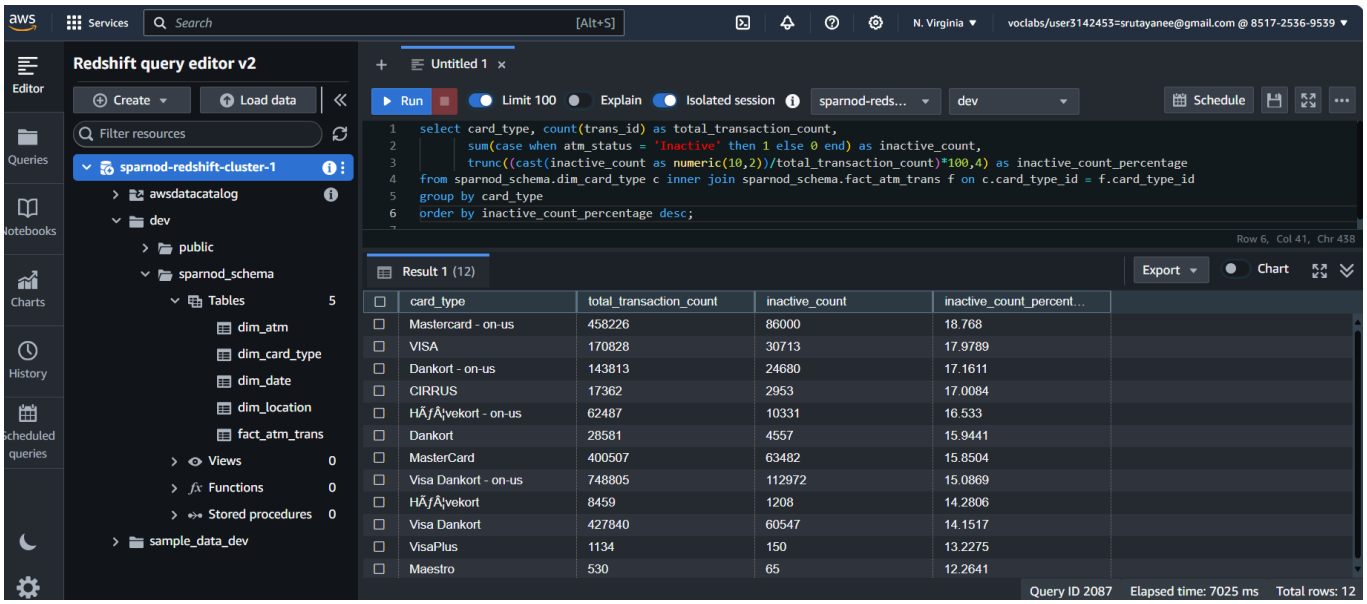
5. Top 10 ATMs with the highest total amount withdrawn throughout the year

```
select atm_number, atm_manufacturer, location, sum(transaction_amount) as
total_transaction_amount

from sparnod_schema.dim_location l inner join sparnod_schema.dim_atm a on l.location_id
= a.atm_location_id

inner join sparnod_schema.fact_atm_trans f on a.atm_id = f.atm_id

group by atm_number, atm_manufacturer, location

order by total_transaction_amount desc

limit 10;
```



6. Number of failed ATM transactions across various card types

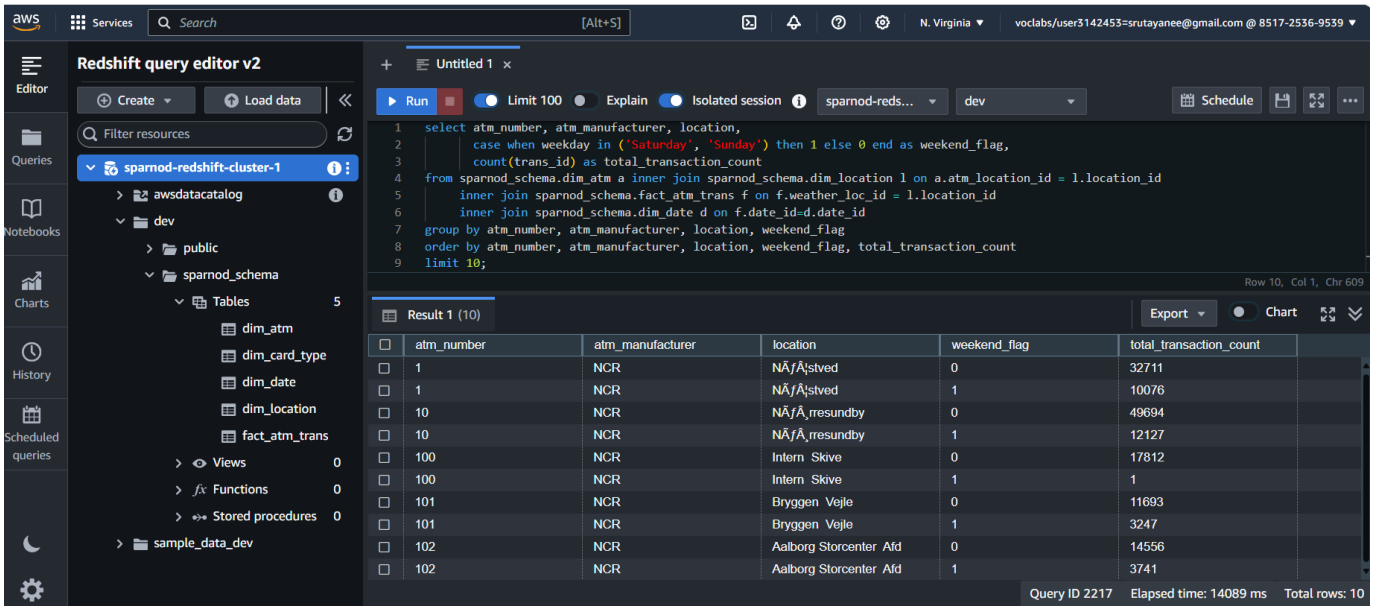
```
select card_type, count(trans_id) as total_transaction_count,
    sum(case when atm_status = 'Inactive' then 1 else 0 end) as inactive_count,
    trunc((cast(inactive_count as numeric(10,2))/total_transaction_count)*100,4) as inactive_count_percentage
from sparnod_schema.dim_card_type c inner join sparnod_schema.fact_atm_trans f on c.card_type_id = f.card_type_id
group by card_type
order by inactive_count_percentage desc;
```



card_type	total_transaction_count	inactive_count	inactive_count_percent...
Mastercard - on-us	458226	86000	18.768
VISA	170828	30713	17.9789
Dankort - on-us	143813	24680	17.1611
CIRRUS	17362	2953	17.0084
HÃƒÂ¦vekort - on-us	62487	10331	16.533
Dankort	28581	4557	15.9441
MasterCard	400507	63482	15.8504
Visa Dankort - on-us	748805	112972	15.0869
HÃƒÂ¦vekort	8459	1208	14.2806
Visa Dankort	427840	60547	14.1517
VisaPlus	1134	150	13.2275
Maestro	530	65	12.2641

7. Top 10 records with the number of transactions ordered by the ATM_number, ATM_manufacturer, location, weekend_flag and then total_transaction_count, on weekdays and on weekends throughout the year

```
select atm_number, atm_manufacturer, location,
    case when weekday in ('Saturday', 'Sunday') then 1 else 0 end as weekend_flag,
    count(trans_id) as total_transaction_count
from sparnod_schema.dim_atm a inner join sparnod_schema.dim_location l on a.atm_location_id = l.location_id
   inner join sparnod_schema.fact_atm_trans f on f.weather_loc_id = l.location_id
   inner join sparnod_schema.dim_date d on f.date_id=d.date_id
group by atm_number, atm_manufacturer, location, weekend_flag
order by atm_number, atm_manufacturer, location, weekend_flag, total_transaction_count
limit 10;
```



atm_number	atm_manufacturer	location	weekend_flag	total_transaction_count
1	NCR	NÃƒÂ¦stved	0	32711
1	NCR	NÃƒÂ¦stved	1	10076
10	NCR	NÃƒÂ¸rresundby	0	49694
10	NCR	NÃƒÂ¸rresundby	1	12127
100	NCR	Intern Skive	0	17812
100	NCR	Intern Skive	1	1
101	NCR	Bryggen Vejle	0	11693
101	NCR	Bryggen Vejle	1	3247
102	NCR	Aalborg Storcenter Afd	0	14556
102	NCR	Aalborg Storcenter Afd	1	3741

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

```
select atm_number, atm_manufacturer, location, weekday, count(trans_id) as total_transaction_count
from sparnod_schema.fact_atm_trans f inner join sparnod_schema.dim_atm a on f.atm_id=a.atm_id
    inner join sparnod_schema.dim_location l on a.atm_location_id = l.location_id
    inner join sparnod_schema.dim_date d on f.date_id=d.date_id
where location = 'Vejgaard' AND weekday IN (
    select weekday
    from sparnod_schema.dim_date d inner join sparnod_schema.fact_atm_trans f
    on d.date_id=f.date_id
    group by weekday
    order by count(trans_id) desc
    limit 1
)
group by atm_number, atm_manufacturer, location, weekday
order by total_transaction_count;
```