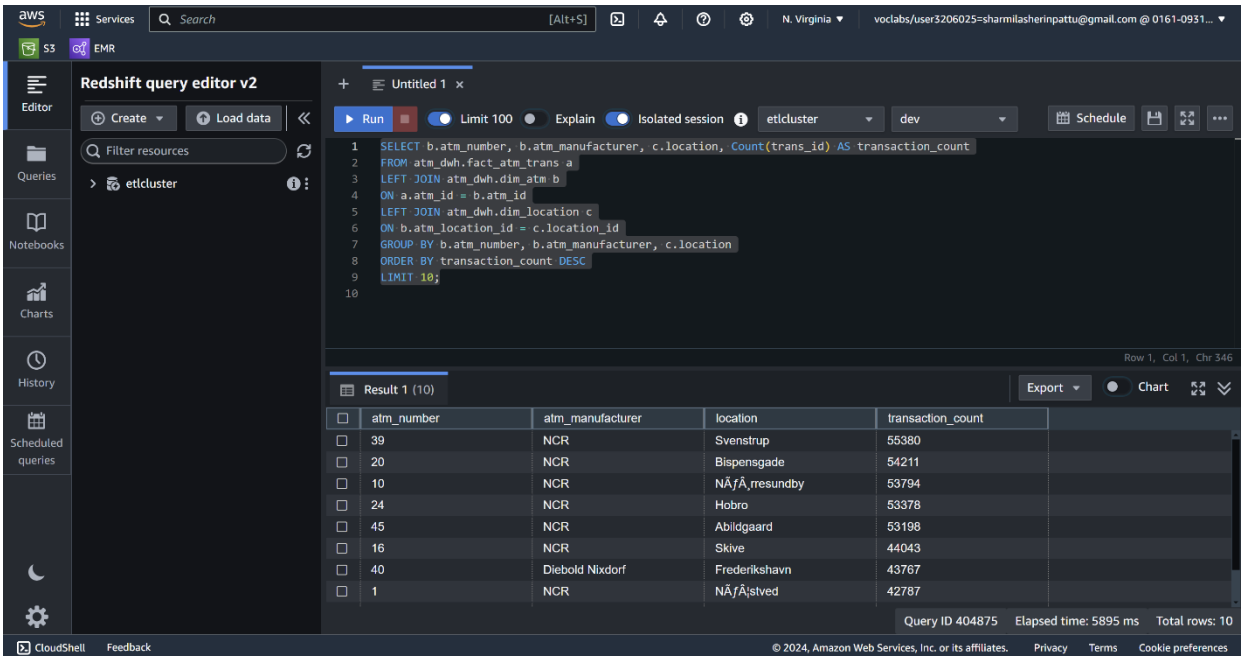
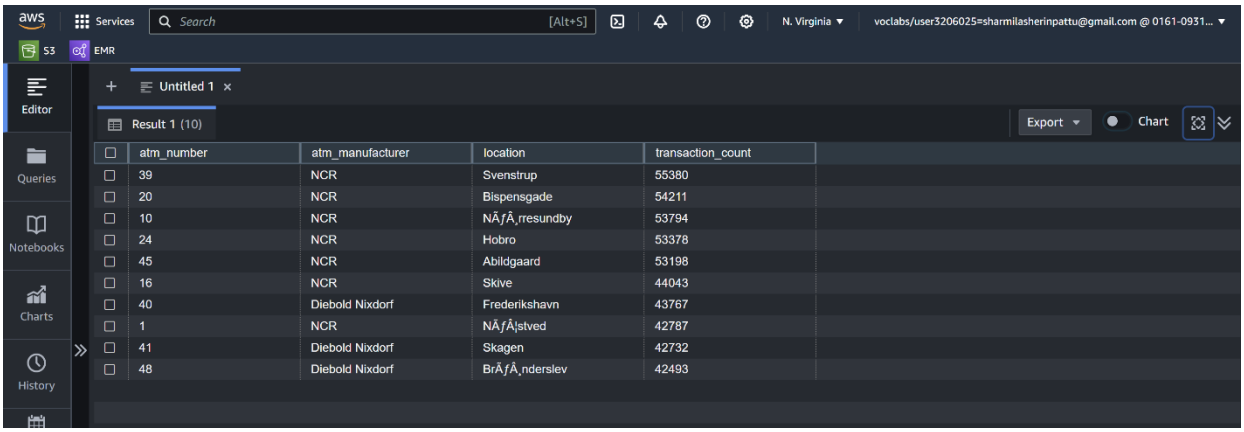
Solving analytical queries on Redshift Cluster

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

```
SELECT b.atm_number,
       b.atm_manufacturer,
       c.location,
       Sum(CASE
             WHEN atm_status = 'Inactive' THEN 1
             ELSE 0
           end) AS inactive_count
FROM atm_dwh.fact_atm_trans a
    LEFT JOIN atm_dwh.dim_atm b
           ON a.atm_id = b.atm_id
    LEFT JOIN atm_dwh.dim_location c
           ON b.atm_location_id = c.location_id
GROUP BY b.atm_number, b.atm_manufacturer, c.location
ORDER BY inactive_count DESC
LIMIT 10;
```

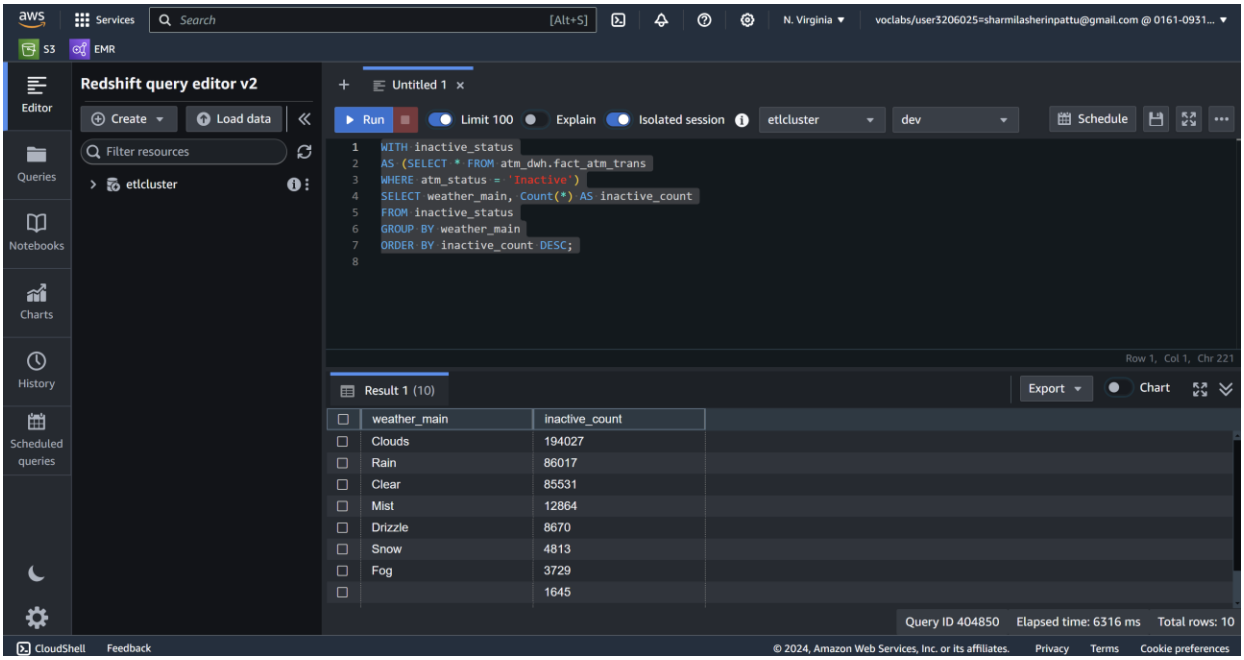




2. Number of ATM failures corresponding to the different weather conditions recorded at the time of the transaction

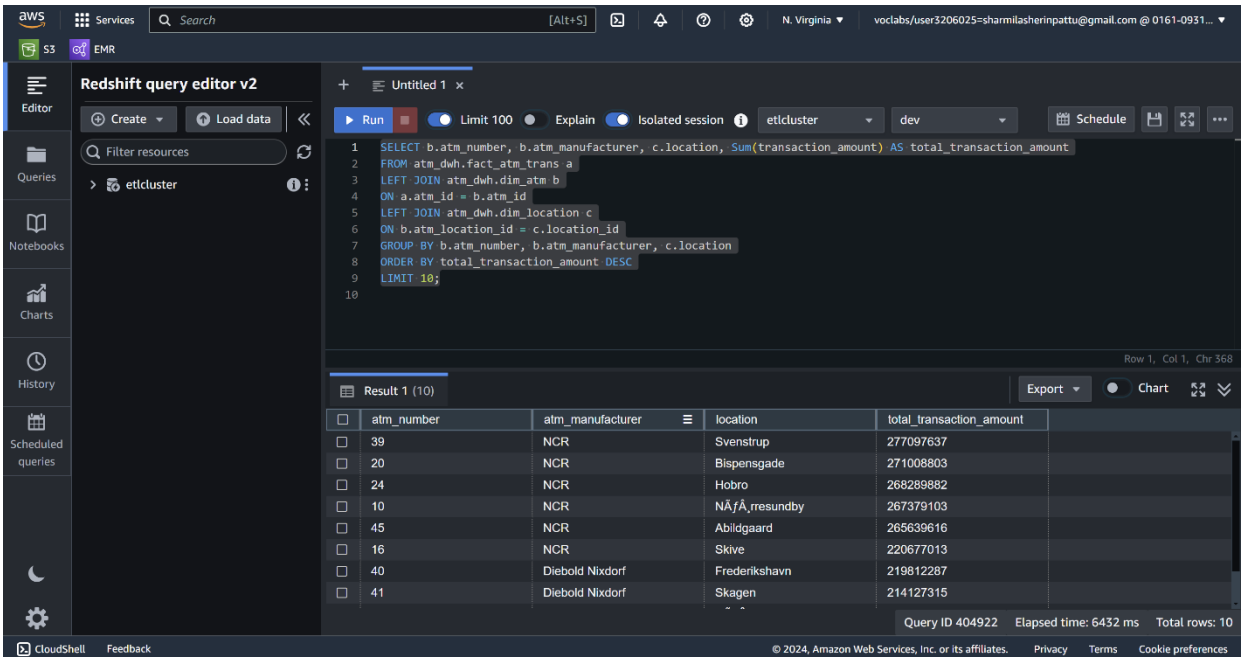
```
WITH inactive_status
AS (SELECT *
      FROM atm_dwh.fact_atm_trans
      WHERE atm_status = 'Inactive')
SELECT weather_main, Count(*) AS inactive_count
FROM inactive_status
```

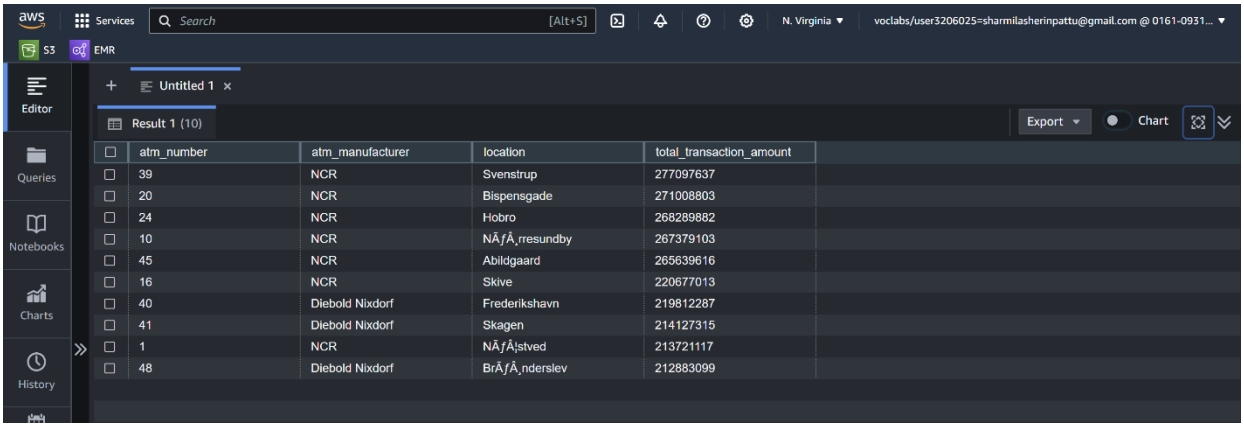
```
GROUP BY weather_main
ORDER BY inactive_count DESC;
```



3. Top 10 ATMs with the most number of transactions

```
SELECT b.atm_number,
    b.atm_manufacturer,
    c.location, Count(trans_id) AS transaction_count
FROM atm_dwh.fact_atm_trans a
    LEFT JOIN atm_dwh.dim_atm b
        ON a.atm_id = b.atm_id
    LEFT JOIN atm_dwh.dim_location c
        ON b.atm_location_id = c.location_id
GROUP BY b.atm_number, b.atm_manufacturer, c.location
ORDER BY transaction_count DESC
LIMIT 10;
```

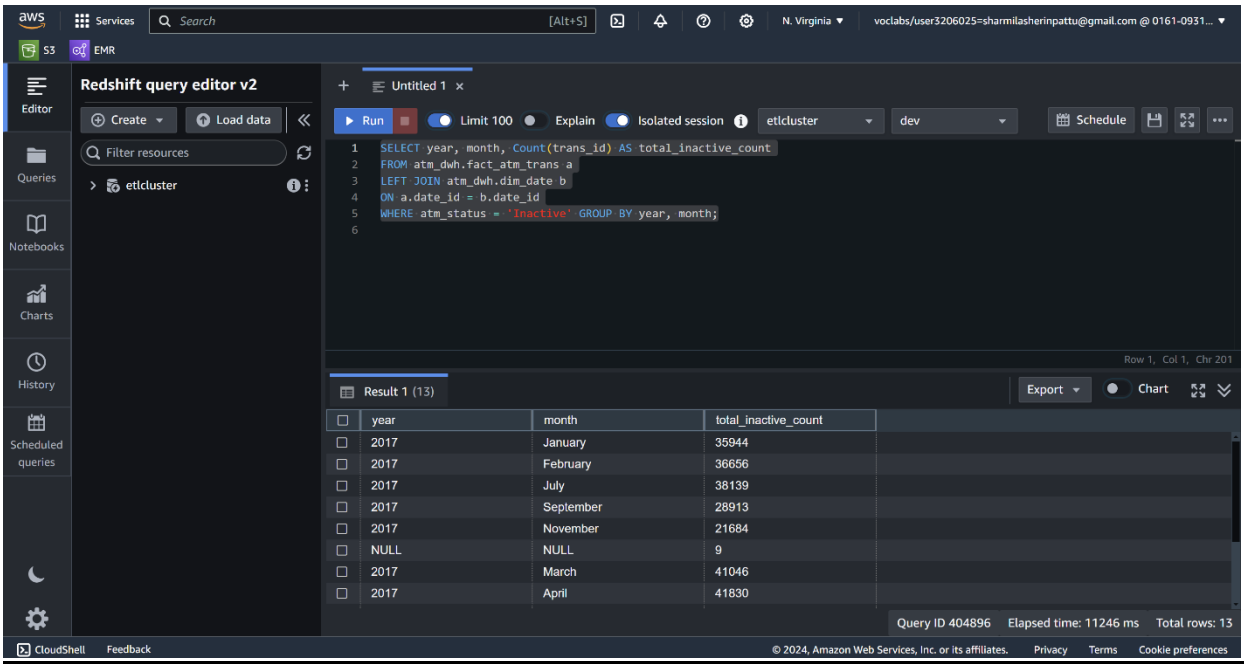


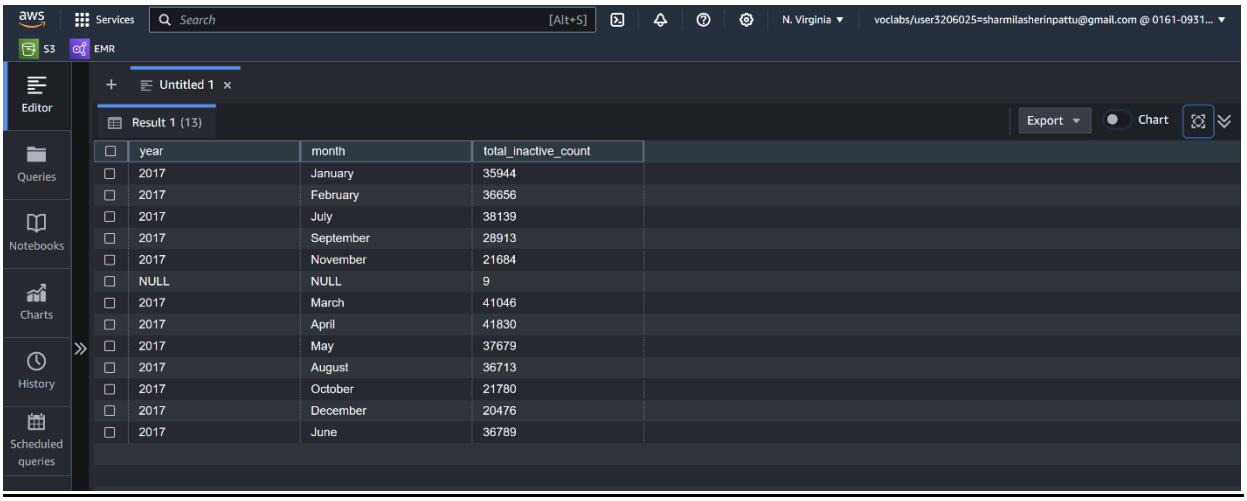


4. Number of overall ATM transactions going inactive per month for each month

```
SELECT year,
    month,
    Count(trans_id) AS total_inactive_count
FROM atm_dwh.fact_atm_trans a
```

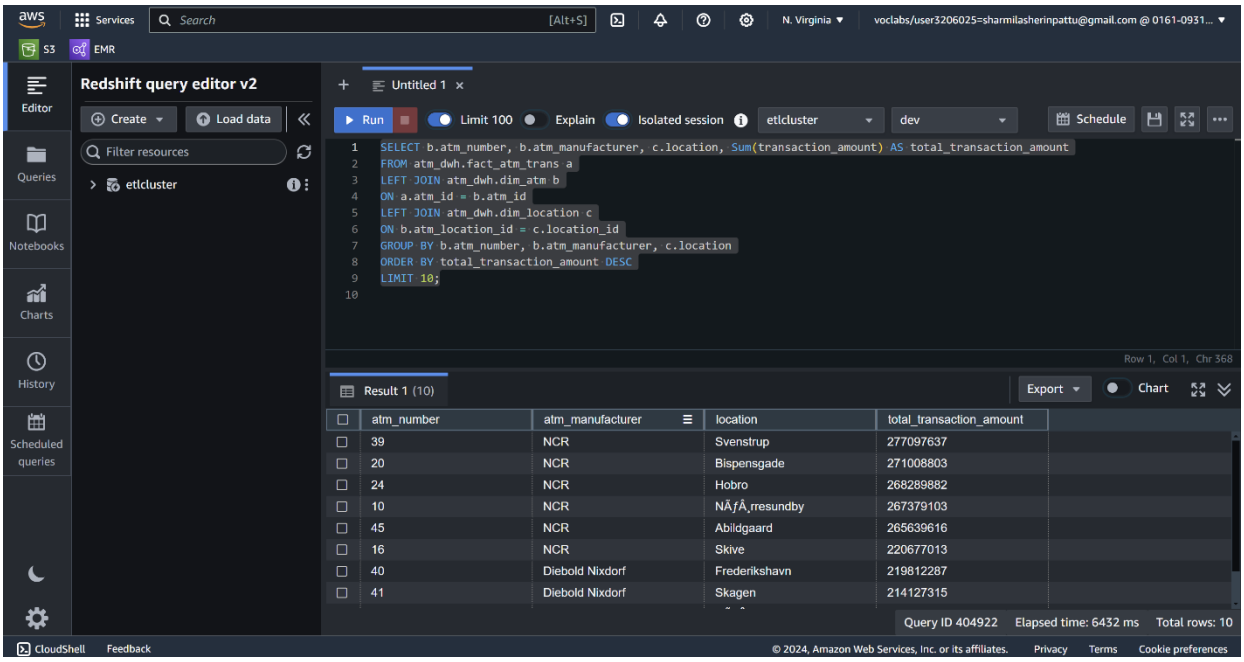
```
    LEFT JOIN atm_dwh.dim_date b
        ON a.date_id = b.date_id
WHERE atm_status = 'Inactive'
GROUP BY year, month;
```

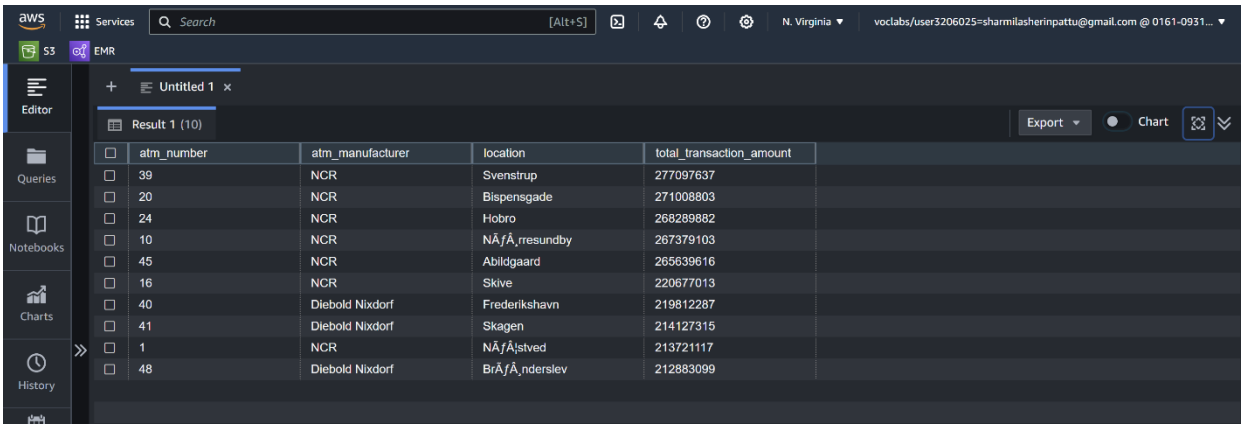




5. Top 10 ATMs with the highest total withdrawn amount throughout the year

```
SELECT b.atm_number,
    b.atm_manufacturer,
    c.location,
    Sum(transaction_amount) AS total_transaction_amount
FROM atm_dwh.fact_atm_trans a
   LEFT JOIN atm_dwh.dim_atm b
       ON a.atm_id = b.atm_id
   LEFT JOIN atm_dwh.dim_location c
       ON b.atm_location_id = c.location_id
GROUP BY b.atm_number, b.atm_manufacturer, c.location
ORDER BY total_transaction_amount DESC
LIMIT 10;
```

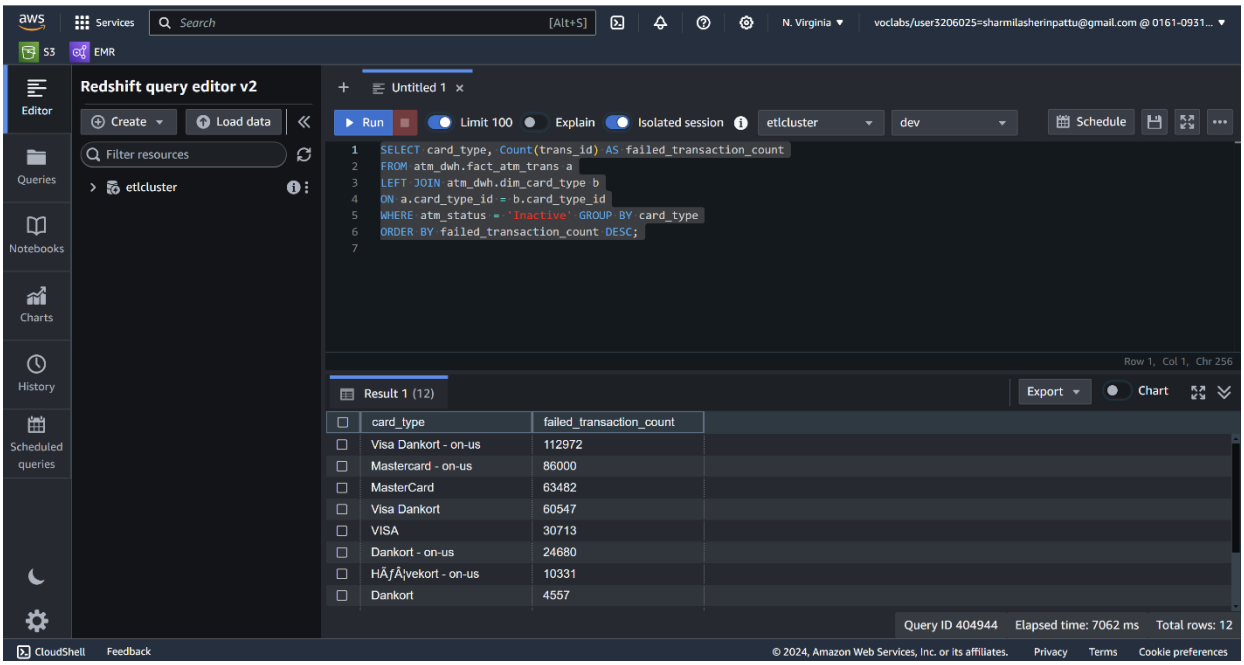




atm_number	atm_manufacturer	location	total_transaction_amount
39	NCR	Svenstrup	277097637
20	NCR	Bispensgade	271008803
24	NCR	Hobro	268289882
10	NCR	NÃƒÂ¸rresundby	267379103
45	NCR	Abildgaard	265639616
16	NCR	Skive	220677013
40	Diebold Nixdorf	Frederikshavn	219812287
41	Diebold Nixdorf	Skagen	214127315
1	NCR	NÃƒÂ¦stved	213721117
48	Diebold Nixdorf	BrÃƒÂ¸nderslev	212883099

6. Number of failed ATM transactions across various card types

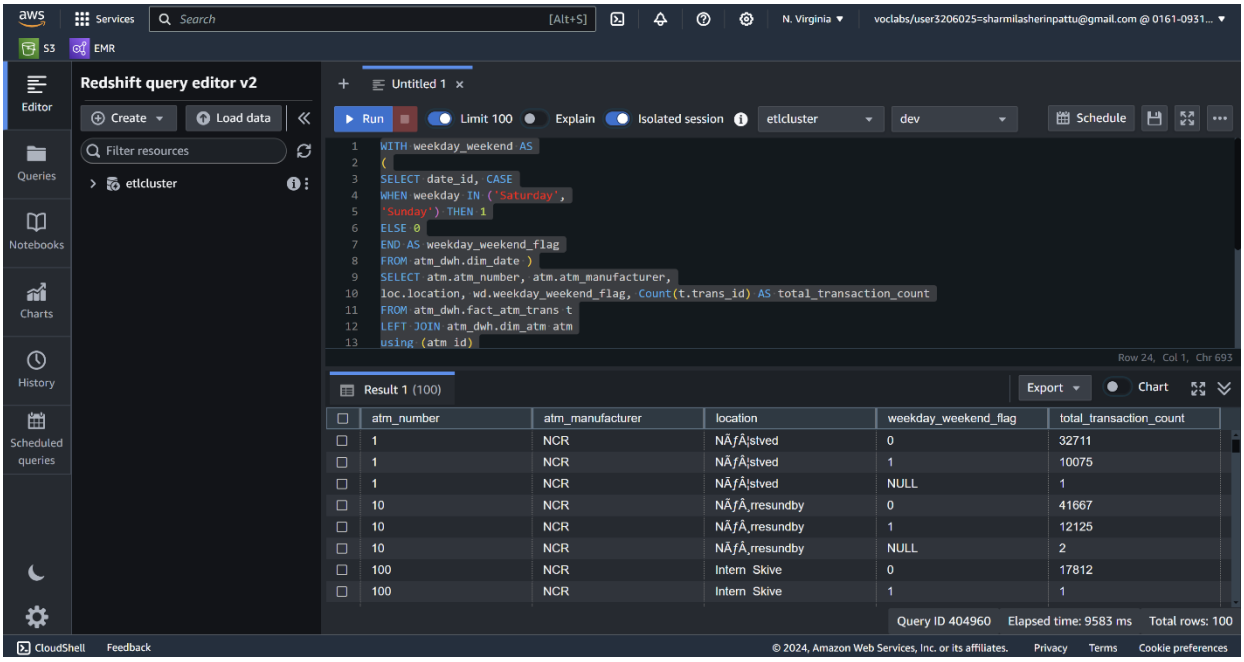
```
SELECT card_type,
    Count(trans_id) AS failed_transaction_count
FROM atm_dwh.fact_atm_trans a
   LEFT JOIN atm_dwh.dim_card_type b
      ON a.card_type_id = b.card_type_id
WHERE atm_status = 'Inactive'
GROUP BY card_type
ORDER BY failed_transaction_count DES;
```

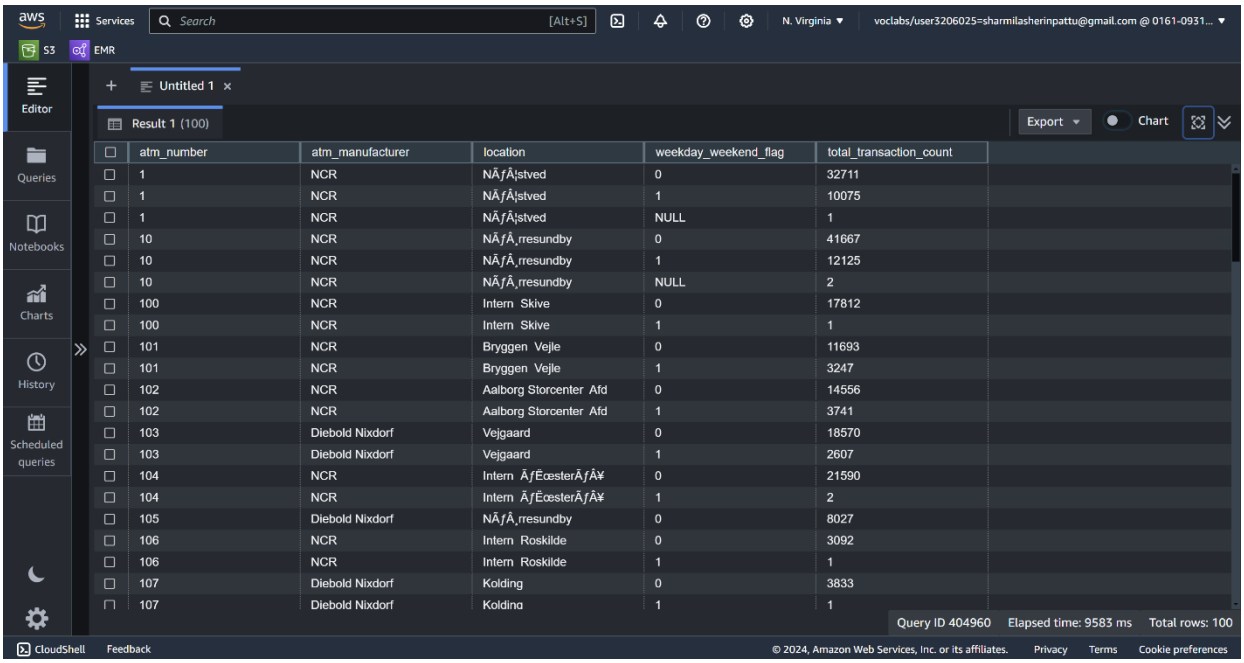


7. Number of transactions happening on an ATM on weekdays and on weekends throughout the year. Order this by the ATM_number, ATM_manufacturer, location, weekend_flag and then total_transaction_count

```
WITH weekday_weekend AS
(
    SELECT date_id,
        CASE
            WHEN weekday IN ('Saturday',
                             'Sunday') THEN 1
            ELSE 0
        END AS weekday_weekend_flag
    FROM atm_dwh.dim_date )
SELECT atm.atm_number,
    atm.atm_manufacturer,
    loc.location,
```

```
  wd.weekday_weekend_flag,
  Count(t.trans_id) AS total_transaction_count
FROM atm_dwh.fact_atm_trans t
  LEFT JOIN atm_dwh.dim_atm atm
      using (atm_id)
  LEFT JOIN weekday_weekend wd
      using (date_id)
  LEFT JOIN atm_dwh.dim_location loc
      ON loc.location_id= t.weather_loc_id
GROUP BY atm.atm_number, atm.atm_manufacturer, loc.location, wd.weekday_weekend_flag
ORDER BY atm.atm_number, atm.atm_manufacturer, loc.location, wd.weekday_weekend_flag,
total_transaction_count;
```





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

```
WITH rank AS
(
   SELECT DISTINCT atm_id,
    atm_number, location,
    weekday,
    count(trans_id)                                AS
 total_transaction_count,
                  row_number() OVER (partition BY atm_id ORDER BY count(trans_id)
 DESC) AS high
FROM atm_dwh.fact_atm_trans f
    LEFT JOIN atm_dwh.dim_atm
        using (atm_id)
  LEFT JOIN atm_dwh.dim_location al
```

```
    ON f.weather_loc_id = al.location_id
  LEFT JOIN atm_dwh.dim_date
    using (date_id) WHERE location = 'Vejgaard'
GROUP BY weekday, atm_id, atm_number, location)
SELECT * FROM rank WHERE high=1;
```