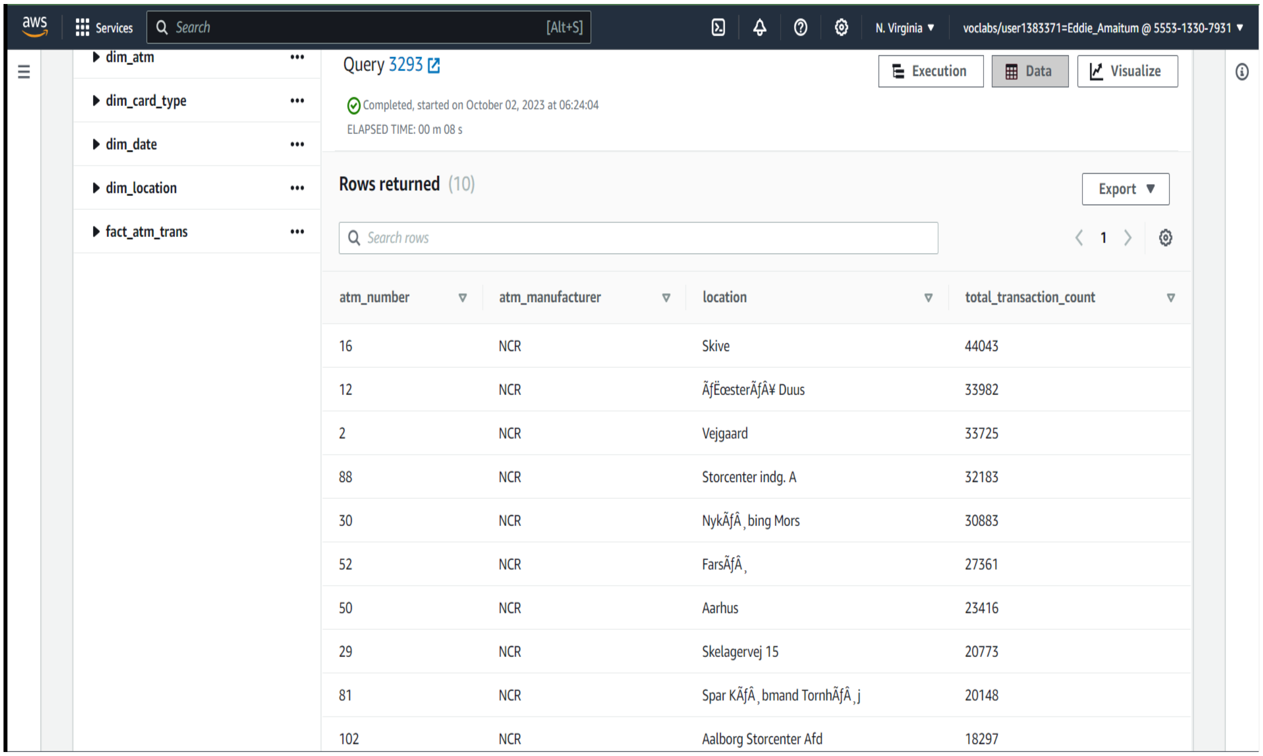
Solving analytical queries on Redshift Cluster

Here, you will find the queries I wrote to solve business questions 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
   atm_number,
   atm_manufacturer,
   location,
   COUNT(trans_id) AS total_transaction_count
FROM
   atm_data.fact_atm_trans T
INNER JOIN
   atm_data.dim_atm A ON T.atm_id = A.atm_id
INNER JOIN
   atm_data.dim_location L ON A.atm_location_id = L.location_id
WHERE
   atm_status = 'Inactive'
GROUP BY
   atm_number,
   atm_manufacturer,
   location
ORDER BY
   total_transaction_count DESC
LIMIT 10;
```



Query 3293

Completed, started on October 02, 2023 at 06:24:04

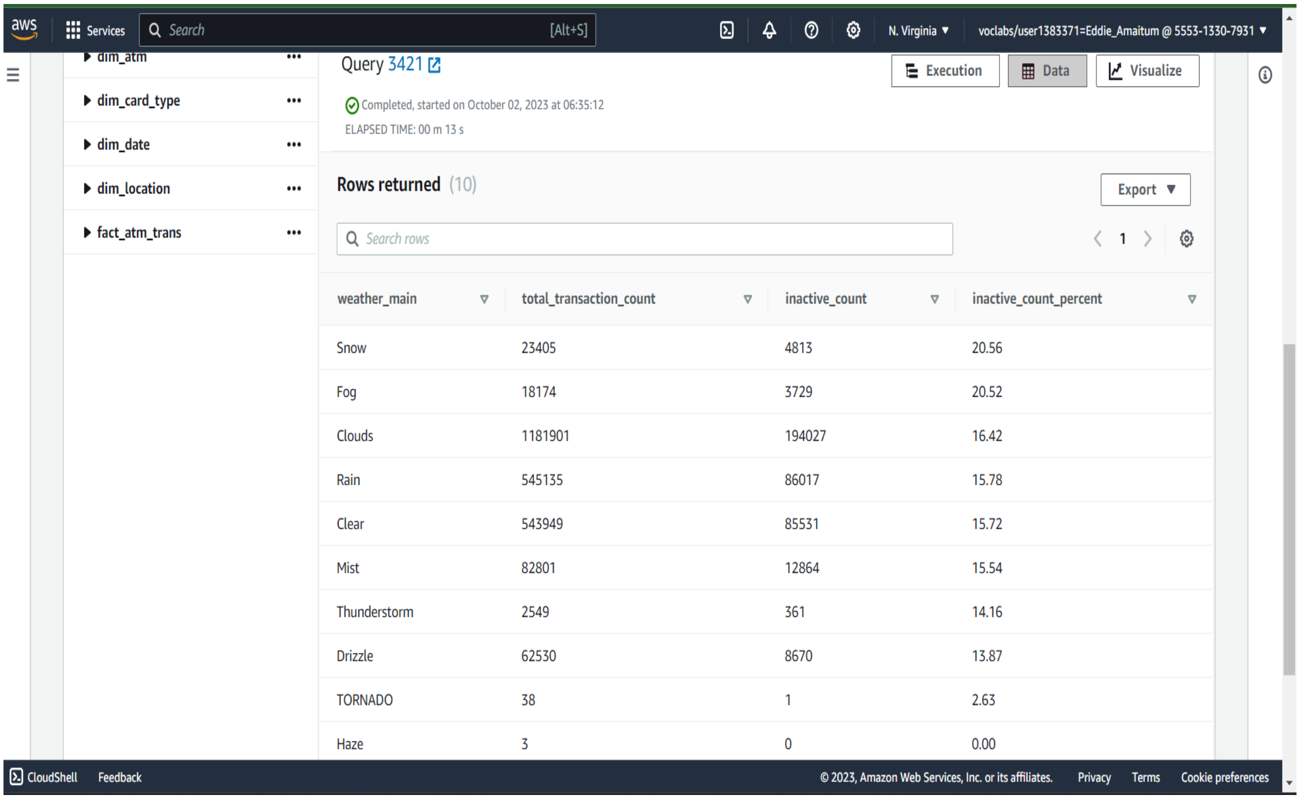
ELAPSED TIME: 00 m 08 s

Rows returned (10)

atm_number	atm_manufacturer	location	total_transaction_count
16	NCR	Skive	44043
12	NCR	ÃƒËœsterÃƒÂ¥ Duus	33982
2	NCR	Vejgaard	33725
88	NCR	Storcenter indg. A	32183
30	NCR	NykÃƒÂ¸bing Mors	30883
52	NCR	FarsÃƒÂ¸	27361
50	NCR	Aarhus	23416
29	NCR	Skelagervej 15	20773
81	NCR	Spar KÃƒÂ¸bmand TornhÃƒÂ¸j	20148
102	NCR	Aalborg Storcenter Afd	18297

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

```
WITH weather_atm_failure AS
 (
   SELECT
      weather_main,
      COUNT(trans_id) AS total_transaction_count,
      SUM(CASE WHEN atm_status = 'Inactive' THEN 1 ELSE 0 END) AS
inactive_count
   FROM
      Atm_data.fact_atm_trans
   WHERE
      weather_main != ''
   GROUP BY
      weather_main
)
SELECT
   *,
   ROUND(CAST(inactive_count AS NUMERIC(10,2)) / total_transaction_count * 100, 2) AS inactive_count_percent
FROM
   weather_atm_failure
ORDER BY
   inactive_count_percent DESC;
```



Query 3421

Completed, started on October 02, 2023 at 06:35:12

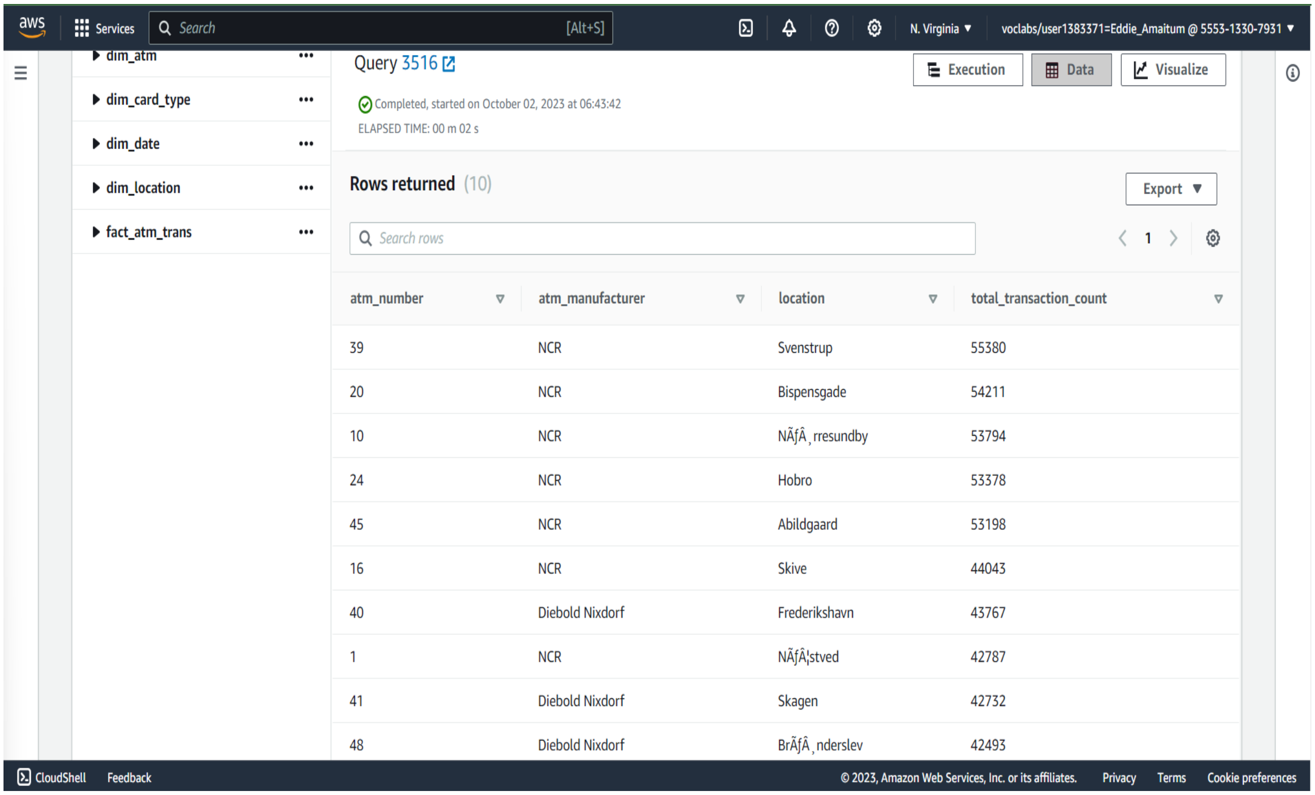
ELAPSED TIME: 00 m 13 s

Rows returned (10)

weather_main	total_transaction_count	inactive_count	inactive_count_percent
Snow	23405	4813	20.56
Fog	18174	3729	20.52
Clouds	1181901	194027	16.42
Rain	545135	86017	15.78
Clear	543949	85531	15.72
Mist	82801	12864	15.54
Thunderstorm	2549	361	14.16
Drizzle	62530	8670	13.87
TORNADO	38	1	2.63
Haze	3	0	0.00

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
   atm_data.fact_atm_trans T
INNER JOIN
   atm_data.dim_atm A ON T.atm_id = A.atm_id
INNER JOIN
   atm_data.dim_location L ON A.atm_location_id = L.location_id
GROUP BY
   atm_number,
   atm_manufacturer,
   location
ORDER BY
   total_transaction_count DESC
LIMIT 10;
```



Query 3516

Completed, started on October 02, 2023 at 06:43:42

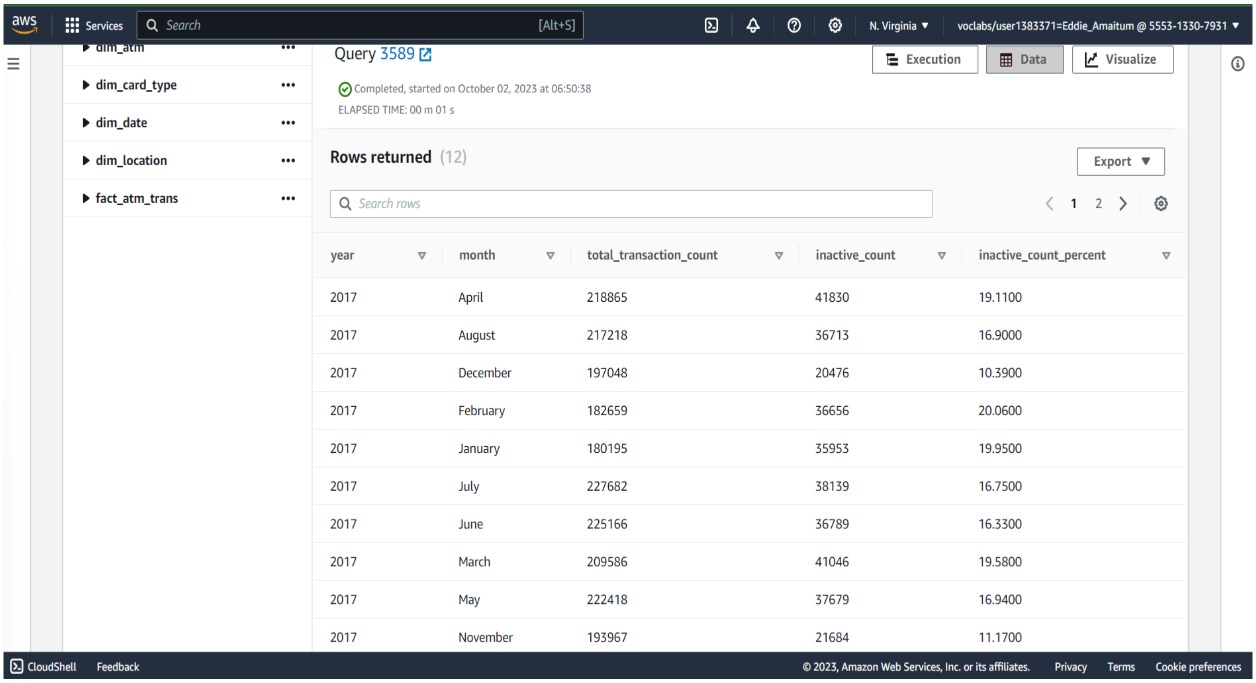
ELAPSED TIME: 00 m 02 s

Rows returned (10)

atm_number	atm_manufacturer	location	total_transaction_count
39	NCR	Svenstrup	55380
20	NCR	Bispensgade	54211
10	NCR	NÃƒÂ¸rresundby	53794
24	NCR	Hobro	53378
45	NCR	Abildgaard	53198
16	NCR	Skive	44043
40	Diebold Nixdorf	Frederikshavn	43767
1	NCR	NÃƒÂ¦stved	42787
41	Diebold Nixdorf	Skagen	42732
48	Diebold Nixdorf	BrÃƒÂ¸nderslev	42493

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

```
SELECT
    d.year,
    d.month,
    COUNT(trans_id) AS total_transaction_count,
    SUM(CASE WHEN atm_status = 'Inactive' THEN 1 ELSE 0 END) AS inactive_count,
    CASE
        WHEN COALESCE(inactive_count, 0) = 0 THEN 0.0000
        ELSE TRUNC((CAST(inactive_count AS NUMERIC(10,4)) /
total_transaction_count) * 100, 2)
    END AS inactive_count_percent
FROM
    atm_data.fact_atm_trans f
INNER JOIN
    atm_data.dim_date d ON f.date_id = d.date_id
GROUP BY
    d.year,
    d.month
ORDER BY
    d.year,
    d.month;
```



Query 3589

Completed, started on October 02, 2023 at 06:50:38

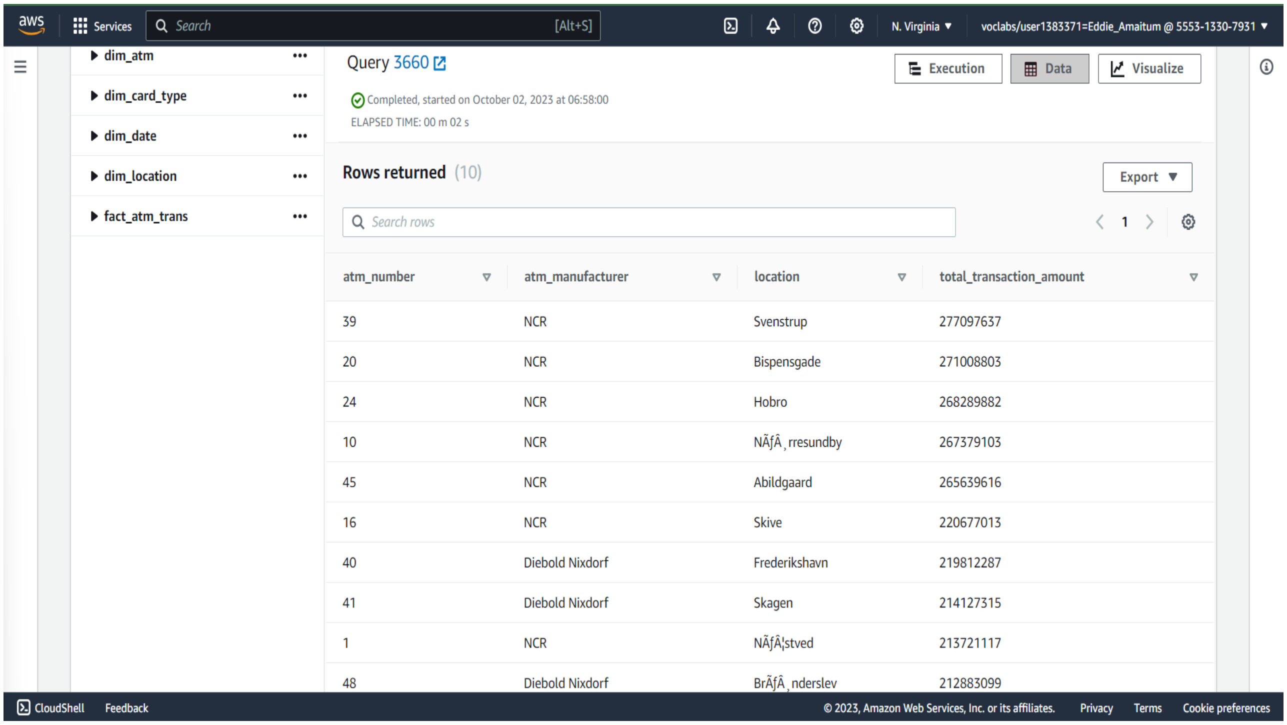
ELAPSED TIME: 00 m 01 s

Rows returned (12)

year	month	total_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
    atm_number,
    atm_manufacturer,
    location,
    SUM(transaction_amount) AS total_transaction_amount
FROM
    atm_data.fact_atm_trans T
INNER JOIN
    atm_data.dim_atm A ON T.atm_id = A.atm_id
INNER JOIN
    atm_data.dim_location L ON A.atm_location_id = L.location_id
GROUP BY
    atm_number,
    atm_manufacturer,
    location
ORDER BY
    total_transaction_amount DESC
LIMIT 10;
```



Query 3660

Completed, started on October 02, 2023 at 06:58:00

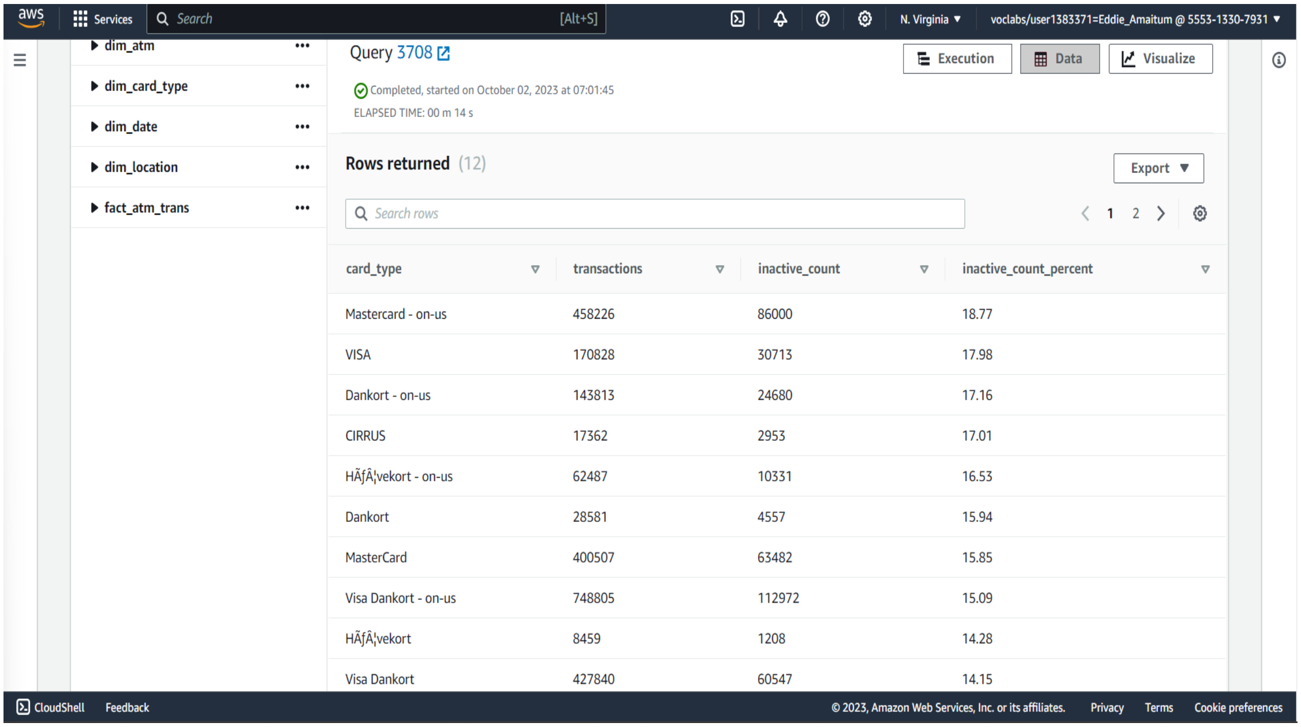
ELAPSED TIME: 00 m 02 s

Rows returned (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

```
WITH card_type_failure AS
(
   SELECT
      card_type,
      COUNT(trans_id) AS transactions,
      SUM(CASE WHEN atm_status = 'Inactive' THEN 1 ELSE 0 END) AS
inactive_count
   FROM
      atm_data.fact_atm_trans F
   INNER JOIN
      atm_data.dim_card_type C ON F.card_type_id = C.card_type_id
   GROUP BY
      card_type
)
SELECT *,
   ROUND(CAST(inactive_count AS NUMERIC(10,2)) / transactions * 100, 2) AS
inactive_count_percent
FROM
   card_type_failure
ORDER BY
   inactive_count_percent DESC;
```



Query 3708

Completed, started on October 02, 2023 at 07:01:45

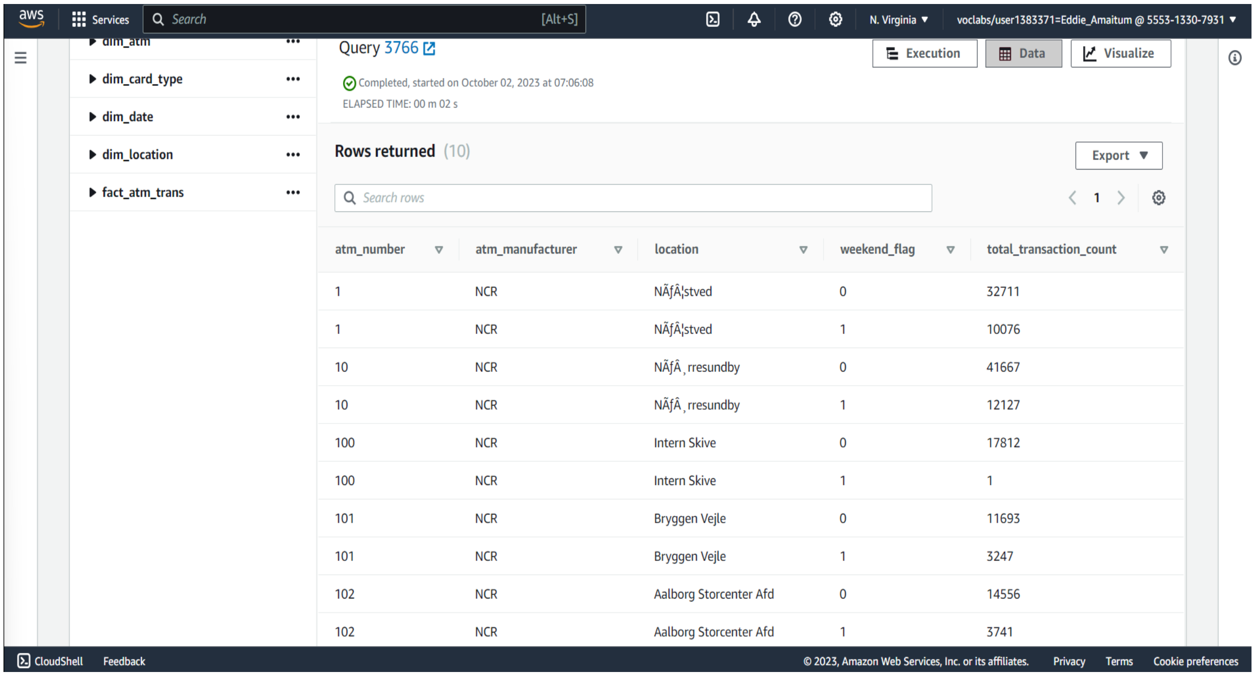
ELAPSED TIME: 00 m 14 s

Rows returned (12)

card_type	transactions	inactive_count	inactive_count_percent
Mastercard - on-us	458226	86000	18.77
VISA	170828	30713	17.98
Dankort - on-us	143813	24680	17.16
CIRRUS	17362	2953	17.01
HÃƒÂ¦vekort - on-us	62487	10331	16.53
Dankort	28581	4557	15.94
MasterCard	400507	63482	15.85
Visa Dankort - on-us	748805	112972	15.09
HÃƒÂ¦vekort	8459	1208	14.28
Visa Dankort	427840	60547	14.15

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
   atm_number,
   atm_manufacturer,
   location,
   CASE WHEN weekday IN ('Sunday', 'Saturday') THEN 1 ELSE 0 END AS
weekend_flag,
   COUNT(trans_id) AS total_transaction_count
FROM
   atm_data.fact_atm_trans T
INNER JOIN
   atm_data.dim_atm A ON T.atm_id = A.atm_id
INNER JOIN
   atm_data.dim_location L ON A.atm_location_id = L.location_id
INNER JOIN
   atm_data.dim_date D ON T.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;
```



Query 3766

Completed, started on October 02, 2023 at 07:06:08

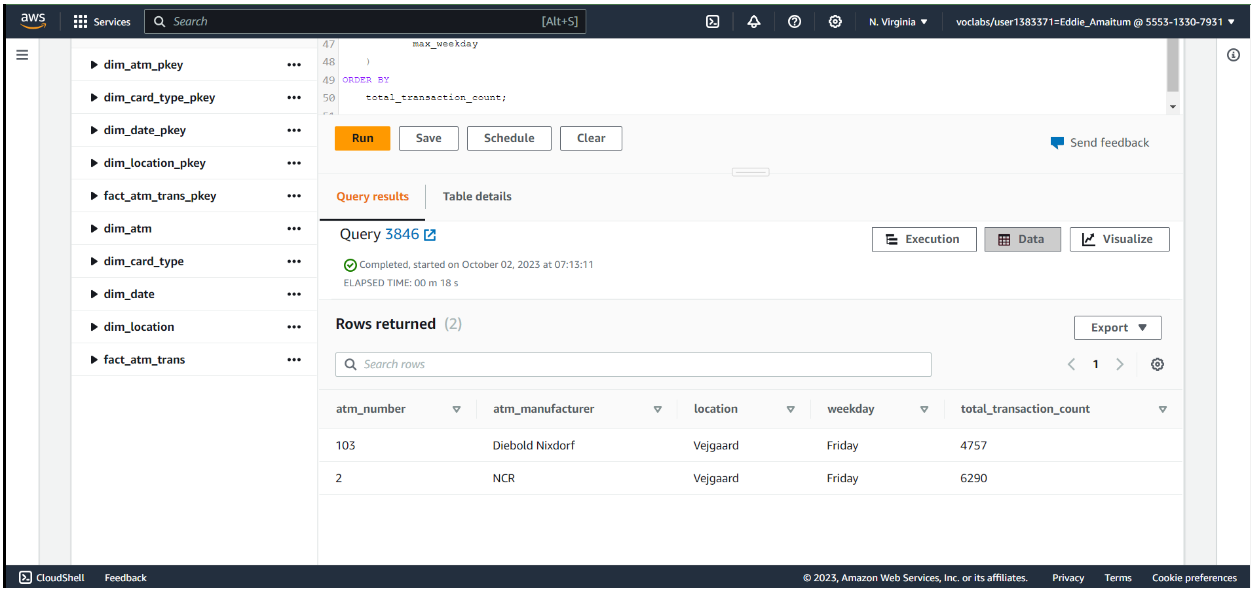
ELAPSED TIME: 00 m 02 s

Rows returned (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	41667
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"

```
WITH atm_weekday_details AS (
   SELECT
      atm_number,
      atm_manufacturer,
      location,
      weekday,
      COUNT(trans_id) AS total_transaction_count
   FROM
      atm_data.fact_atm_trans T
   INNER JOIN
      atm_data.dim_atm A ON T.atm_id = A.atm_id
   INNER JOIN
      atm_data.dim_location L ON A.atm_location_id = L.location_id
   INNER JOIN
      atm_data.dim_date D ON T.date_id = D.date_id
   WHERE
      location = 'Vejgaard'
   GROUP BY
      atm_number,
        atm_manufacturer,
        location,
        weekday
),
max_weekday AS (
    SELECT
        weekday
    FROM
        atm_weekday_details
    WHERE
        total_transaction_count = (
            SELECT
                MAX(total_transaction_count)
            FROM
                atm_weekday_details
        )
    LIMIT 1
)
SELECT
    *
FROM
    atm_weekday_details
WHERE
    weekday = (
        SELECT
            weekday
        FROM
            max_weekday
    )
ORDER BY
    total_transaction_count;
```



```
            max_weekday
    )
ORDER BY
    total_transaction_count;
```

Query 3846

Completed, started on October 02, 2023 at 07:13:11

ELAPSED TIME: 00 m 18 s

Rows returned (2)

atm_number	atm_manufacturer	location	weekday	total_transaction_count
103	Diebold Nixdorf	Vejgaard	Friday	4757
2	NCR	Vejgaard	Friday	6290