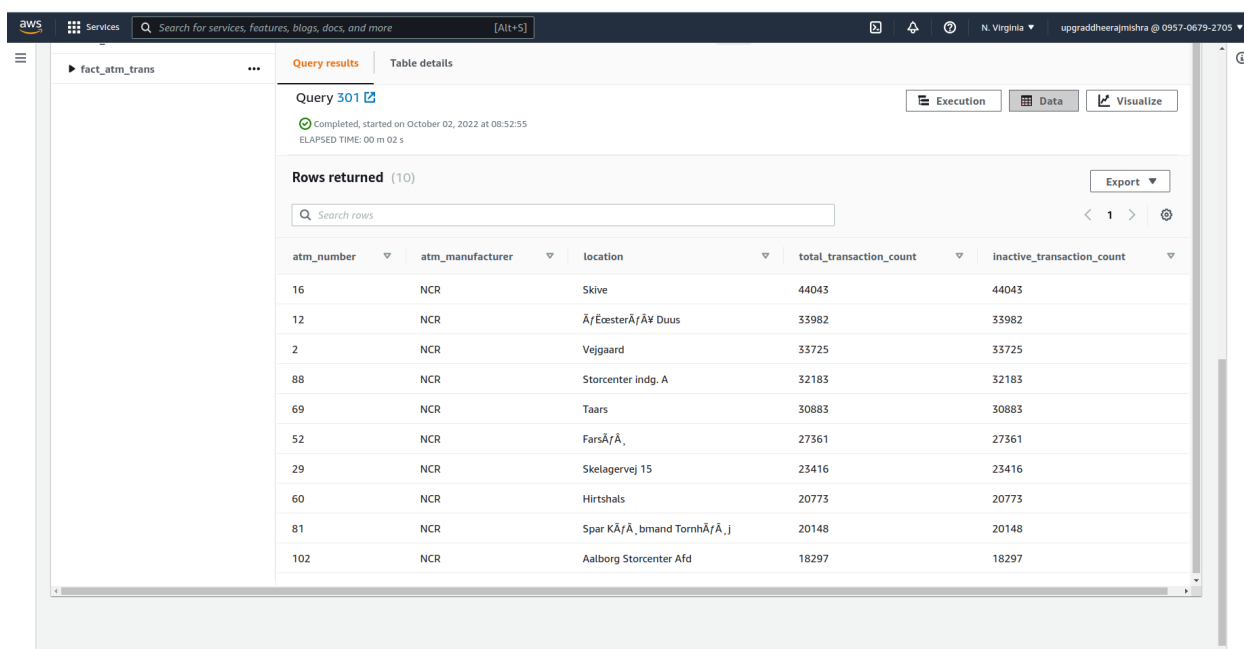


Solving analytical queries on Redshift Cluster

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
from etl_atm_data.fact_atm_trans f, etl_atm_data.dim_atm a, etl_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 inactive_transaction_count desc
limit 10;
```

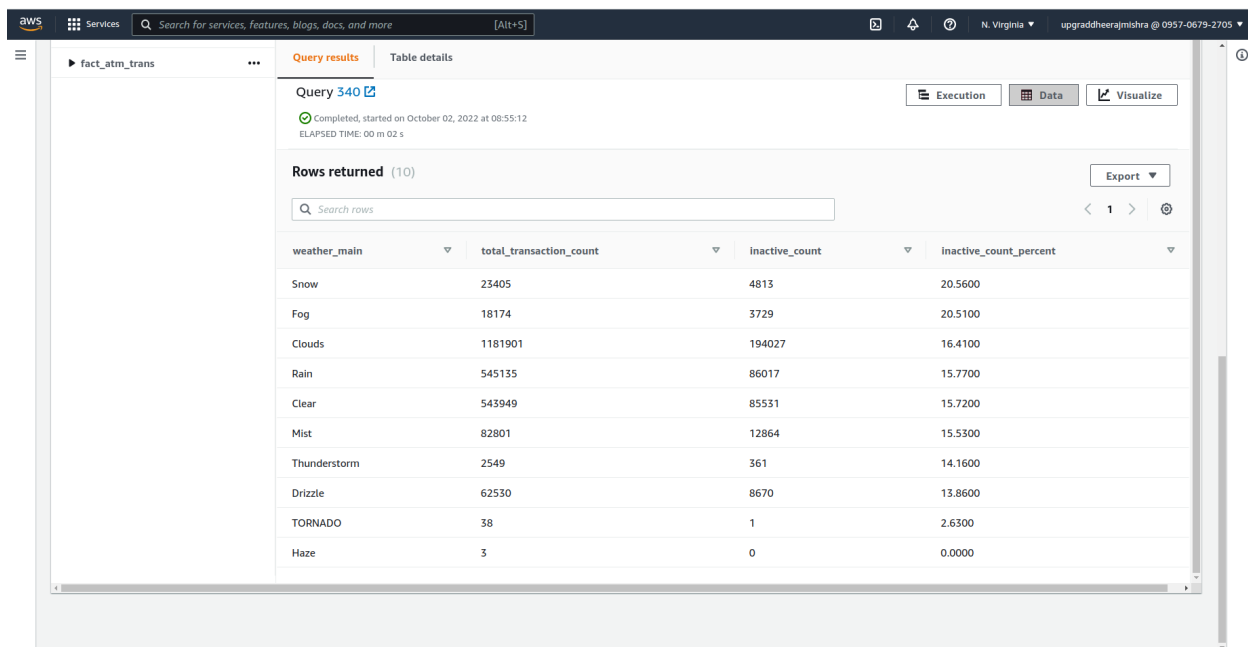


The screenshot shows the AWS Redshift Query Results interface. The query is labeled 'Query 301' and is completed. The results table shows the top 10 ATMs based on the number of inactive transactions. The columns are: atm_number, atm_manufacturer, location, total_transaction_count, and inactive_transaction_count.

atm_number	atm_manufacturer	location	total_transaction_count	inactive_transaction_count
16	NCR	Skive	44043	44043
12	NCR	Århus	33982	33982
2	NCR	Vejgaard	33725	33725
88	NCR	Storcenter indg. A	32183	32183
69	NCR	Taars	30883	30883
52	NCR	Farsø	27361	27361
29	NCR	Skelagervej 15	23416	23416
60	NCR	Hirtshals	20773	20773
81	NCR	Spar K&A, bmand Tornh&A, j	20148	20148
102	NCR	Aalborg Storcenter Afd	18297	18297

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

```
select f.weather_main,
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 etl_atm_data.fact_atm_trans f
where f.weather_main != ''
group by f.weather_main
order by inactive_count_percent desc
limit 10;
```



Query 340

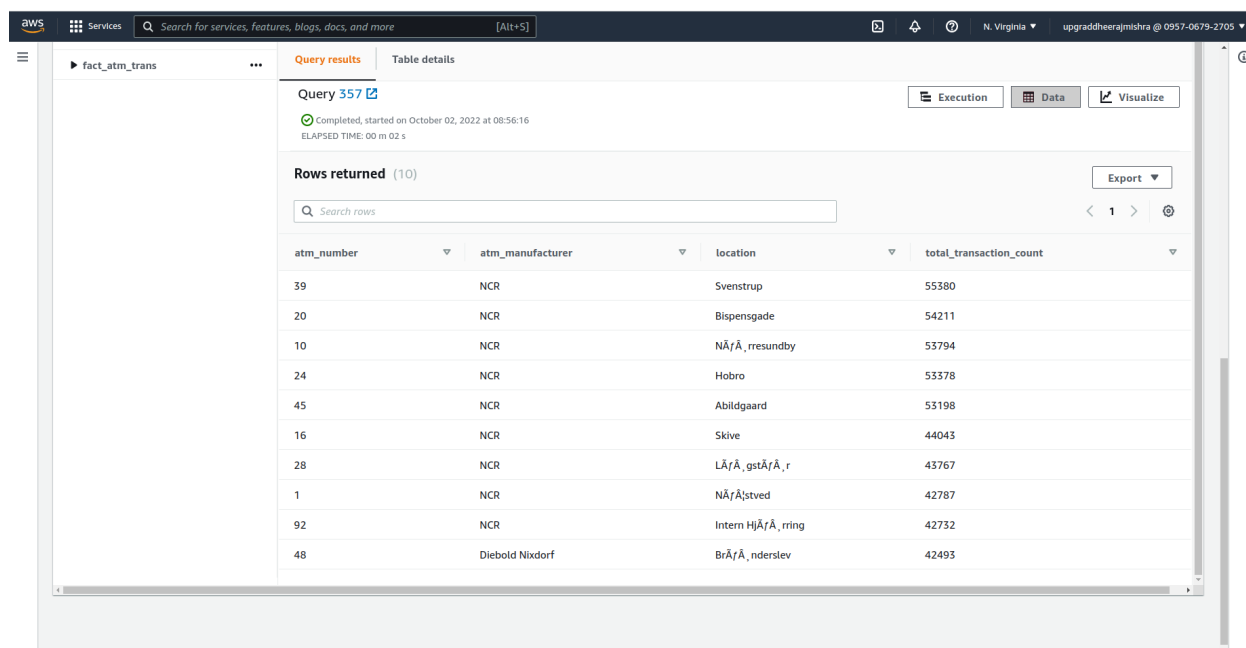
Completed, started on October 02, 2022 at 08:55:12
ELAPSED TIME: 00 m 02 s

Rows returned (10)

weather_main	total_transaction_count	inactive_count	inactive_count_percent
Snow	23405	4813	20.5600
Fog	18174	3729	20.5100
Clouds	1181901	194027	16.4100
Rain	545135	86017	15.7700
Clear	543949	85531	15.7200
Mist	82801	12864	15.5300
Thunderstorm	2549	361	14.1600
Drizzle	62530	8670	13.8600
TORNADO	38	1	2.6300
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 etl_atm_data.fact_atm_trans f, etl_atm_data.dim_atm a, etl_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;
```

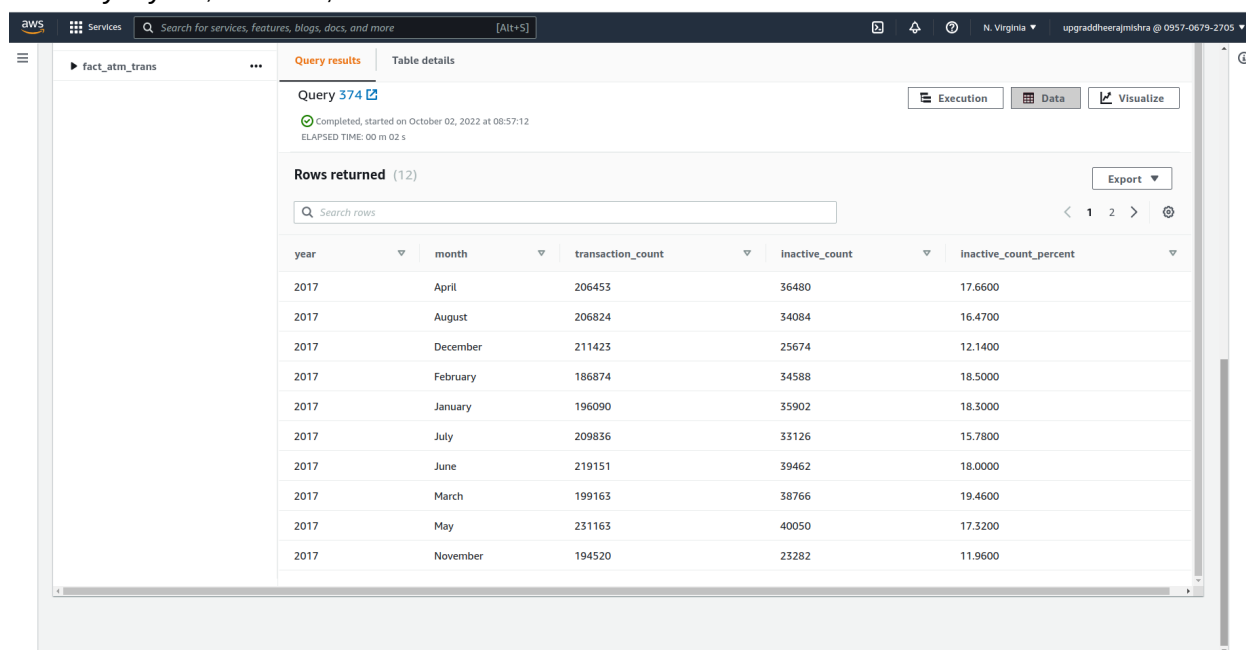


The screenshot shows the AWS Redshift Query Results interface. The query is labeled 'Query 357' and is completed. The results are displayed in a table with 4 columns: atm_number, atm_manufacturer, location, and total_transaction_count. The table shows the top 10 ATMs with the highest transaction counts.

atm_number	atm_manufacturer	location	total_transaction_count
39	NCR	Svenstrup	55380
20	NCR	Bispensgade	54211
10	NCR	NÅfÅ, resundby	53794
24	NCR	Hobro	53378
45	NCR	Abildgaard	53198
16	NCR	Skive	44043
28	NCR	LÅfÅ, gstÅfÅ, r	43767
1	NCR	NÅfÅ)stved	42787
92	NCR	Intern HjÅfÅ, ring	42732
48	Diebold Nixdorf	BrÅfÅ, nderslev	42493

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 etl_atm_data.dim_date
a,etl_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 etl_atm_data.dim_date
a,etl_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;
```



Query 374

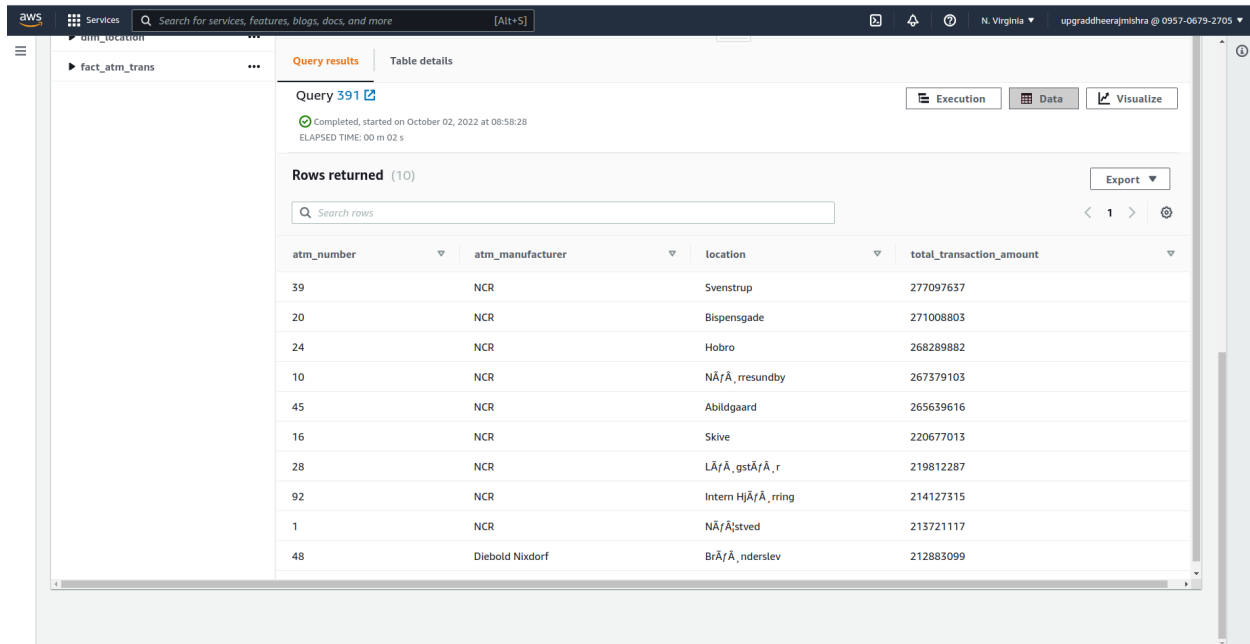
Completed, started on October 02, 2022 at 08:57:12
ELAPSED TIME: 00 m 02 s

Rows returned (12)

year	month	transaction_count	inactive_count	inactive_count_percent
2017	April	206453	36480	17.6600
2017	August	206824	34084	16.4700
2017	December	211423	25674	12.1400
2017	February	186874	34588	18.5000
2017	January	196090	35902	18.3000
2017	July	209836	33126	15.7800
2017	June	219151	39462	18.0000
2017	March	199163	38766	19.4600
2017	May	231163	40050	17.3200
2017	November	194520	23282	11.9600

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

```
select a.atm_number, a.atm_manufacturer, l.location,
sum(transaction_amount) as total_transaction_amount
from etl_atm_data.fact_atm_trans f, etl_atm_data.dim_atm a, etl_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_amount desc
limit 10;
```



Query 391

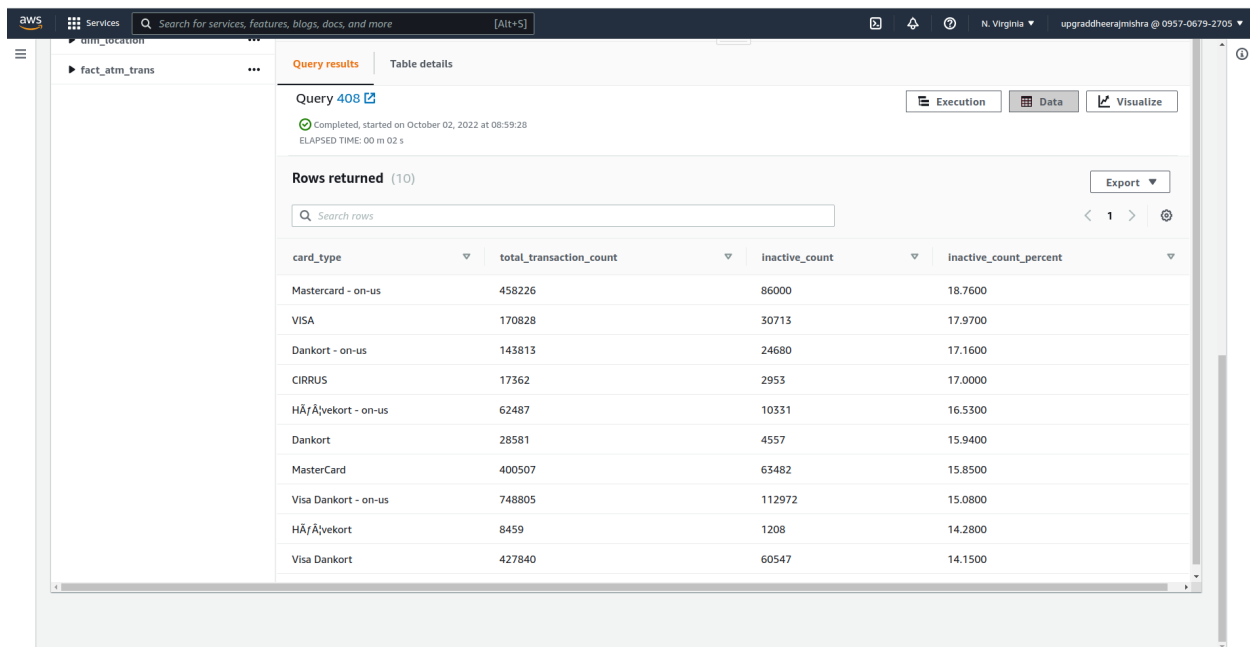
Completed, started on October 02, 2022 at 08:58:28
ELAPSED TIME: 00 m 02 s

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	NÅfÅ_rresundby	267379103
45	NCR	Abildgaard	265639616
16	NCR	Skive	220677013
28	NCR	LÅfÅ_gstÅfÅ_r	219812287
92	NCR	Intern HjÅfÅ_ring	214127315
1	NCR	NÅfÅ'stved	213721117
48	Diebold Nixdorf	BrÅfÅ_nderslev	212883099

6. Number of failed ATM transactions across various card types

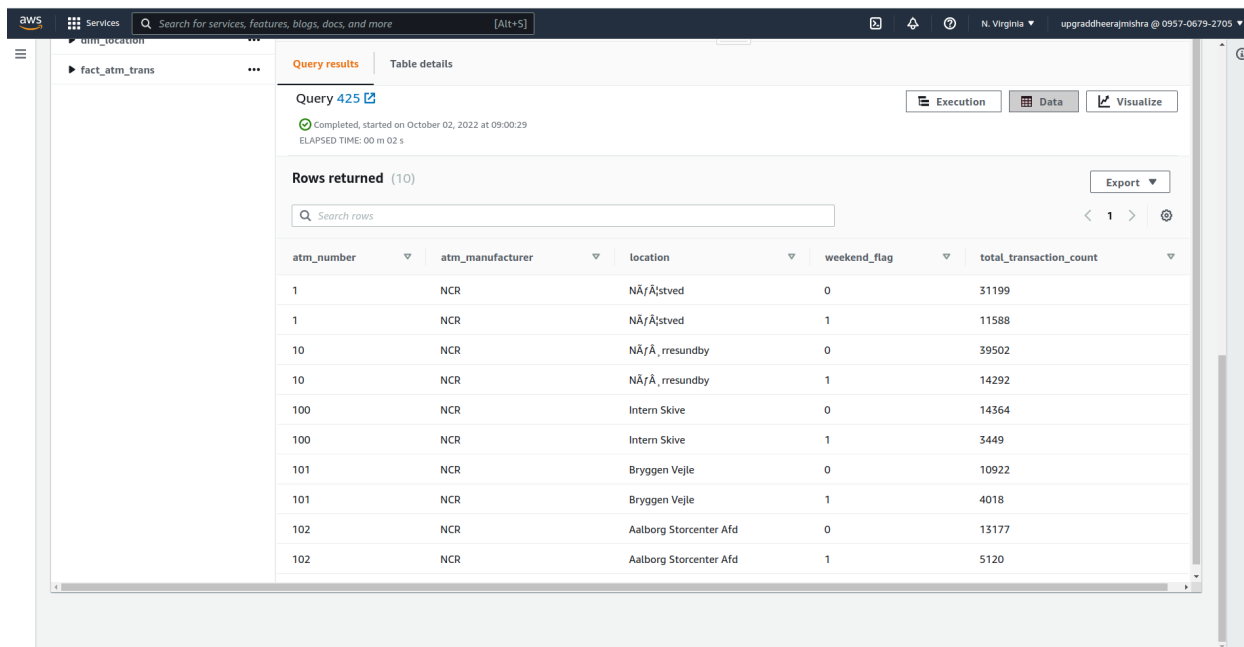
```
select ct.card_type,
count(trans_id) as total_transaction_count,
sum(case when atm_status = 'Inactive' then 1 else 0 end) as inactive_count,
case when coalesce(inactive_count, 0) = 0 then 0.0000
else trunc((cast(inactive_count as
numeric(10,4))/total_transaction_count)*100, 2)
end as inactive_count_percent
from etl_atm_data.fact_atm_trans f, etl_atm_data.dim_card_type ct
where f.card_type_id = ct.card_type_id
group by ct.card_type
order by inactive_count_percent desc
limit 10;
```



card_type	total_transaction_count	inactive_count	inactive_count_percent
Mastercard - on-us	458226	86000	18.7600
VISA	170828	30713	17.9700
Dankort - on-us	143813	24680	17.1600
CIRRUS	17362	2953	17.0000
HÃ¶fÃ¶vekort - on-us	62487	10331	16.5300
Dankort	28581	4557	15.9400
MasterCard	400507	63482	15.8500
Visa Dankort - on-us	748805	112972	15.0800
HÃ¶fÃ¶vekort	8459	1208	14.2800
Visa Dankort	427840	60547	14.1500

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

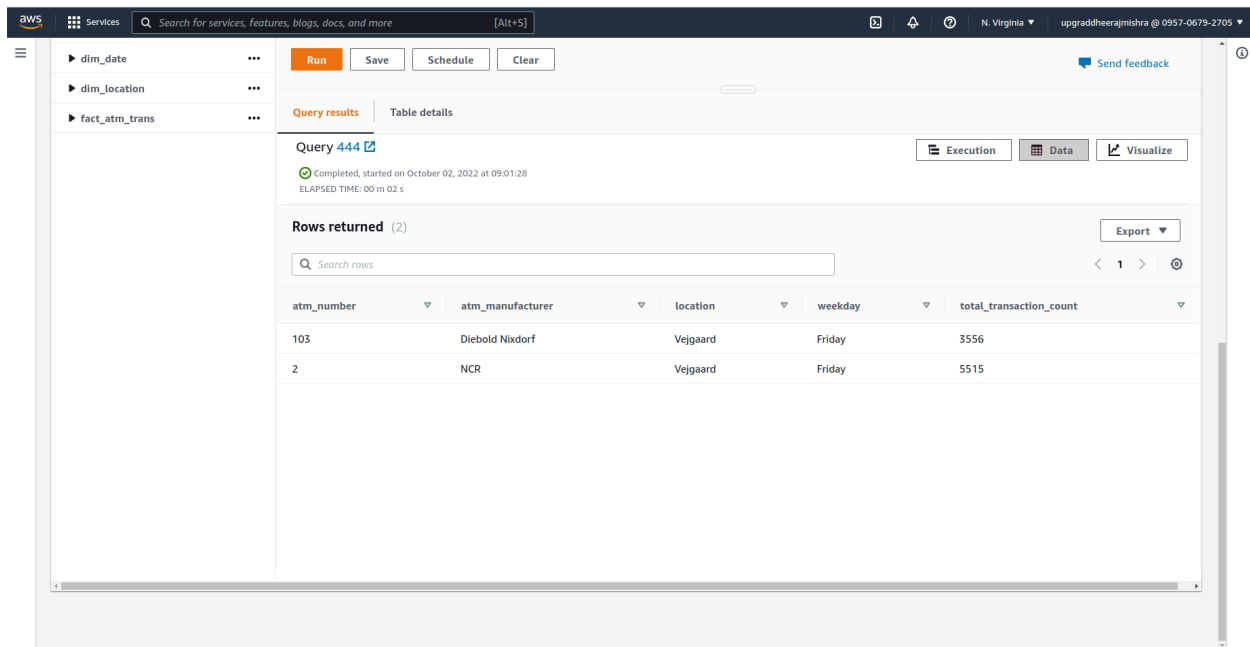


The screenshot shows the AWS Glue console interface. The 'Query results' tab is active, displaying the results of 'Query 425'. The query is completed, started on October 02, 2022 at 09:00:29, and took 00 m 02 s to execute. The results table has 10 rows and 5 columns: atm_number, atm_manufacturer, location, weekend_flag, and total_transaction_count. The data is sorted by atm_number, then atm_manufacturer, then location, then weekend_flag, and finally by total_transaction_count in descending order.

atm_number	atm_manufacturer	location	weekend_flag	total_transaction_count
1	NCR	NÅ/Åstved	0	31199
1	NCR	NÅ/Åstved	1	11588
10	NCR	NÅ/Å, rresundby	0	39502
10	NCR	NÅ/Å, rresundby	1	14292
100	NCR	Intern Skive	0	14364
100	NCR	Intern Skive	1	3449
101	NCR	Bryggen Vejle	0	10922
101	NCR	Bryggen Vejle	1	4018
102	NCR	Aalborg Storcenter Afd	0	13177
102	NCR	Aalborg Storcenter Afd	1	5120

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

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



The screenshot shows the AWS Glue console interface. On the left, a sidebar lists the data catalog tables: dim_date, dim_location, and fact_atm_trans. The main panel displays the 'Query results' for 'Query 444'. The query status is 'Completed, started on October 02, 2022 at 09:01:28' with an 'ELAPSED TIME: 00 m 02 s'. Below the status, it shows 'Rows returned (2)'. A search bar is present above a table with 5 columns: atm_number, atm_manufacturer, location, weekday, and total_transaction_count. The table contains two rows of data.

atm_number	atm_manufacturer	location	weekday	total_transaction_count
103	Diebold Nixdorf	Vejgaard	Friday	3556
2	NCR	Vejgaard	Friday	5515