Solving analytical queries on RedShift Cluster

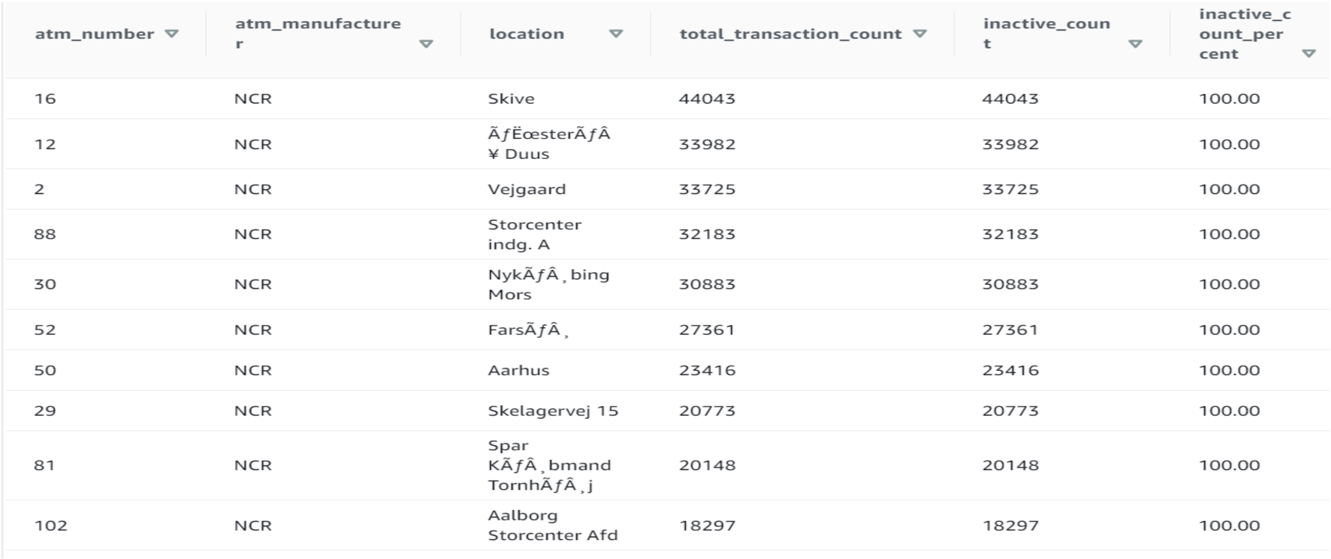
Here, you have to write the query used for solving the question and the screenshots of the table which is outputted after the query is run on the AWS RedShift Query editor UI.

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

<Query>

```
SELECT top 10 atm_number, atm_manufacturer, location, count(*) as total_transaction_count,
sum(CASE WHEN atm_status = 'Inactive' THEN 1 ELSE 0 END) as inactive_count,
ROUND(((inactive_count*100.0)/total_transaction_count),2) as inactive_count_percent
FROM transactions.fact_atm_trans as ft
JOIN transactions.dim_atm as dt ON ft.atm_id = dt.atm_id
JOIN transactions.dim_location as dl ON ft.weather_loc_id = dl.location_id
GROUP BY atm_number, atm_manufacturer, location
ORDER BY inactive_count DESC;
```

<Screenshot of the resultant table>



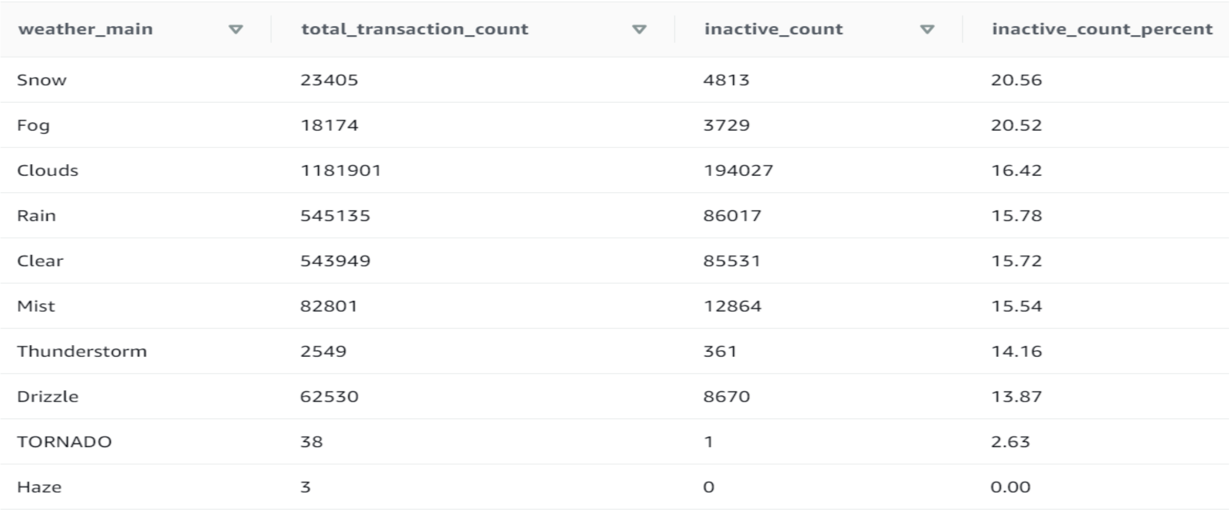
atm_number	atm_manufacturer	location	total_transaction_count	inactive_count	inactive_count_percent
16	NCR	Skive	44043	44043	100.00
12	NCR	ÃƒËœsterÃƒÂ¥ Duus	33982	33982	100.00
2	NCR	Vejgaard	33725	33725	100.00
88	NCR	Storcenter indg. A	32183	32183	100.00
30	NCR	NykÃƒÂ¸bing Mors	30883	30883	100.00
52	NCR	FarsÃƒÂ¸	27361	27361	100.00
50	NCR	Aarhus	23416	23416	100.00
29	NCR	Skelagervej 15	20773	20773	100.00
81	NCR	Spar KÃƒÂ¸bmand TornhÃƒÂ¸j	20148	20148	100.00
102	NCR	Aalborg Storcenter Afd	18297	18297	100.00

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

<Query>

```
SELECT weather_main, count(*) as total_transaction_count,  sum(CASE WHEN atm_status =
'Inactive' THEN 1 ELSE 0 END) as inactive_count,
ROUND(((inactive_count*100.0)/total_transaction_count),2) as inactive_count_percent
FROM transactions.fact_atm_trans
where weather_main is not null
GROUP BY weather_main
ORDER BY inactive_count_percent DESC;
```

<Screenshot of the resultant table>



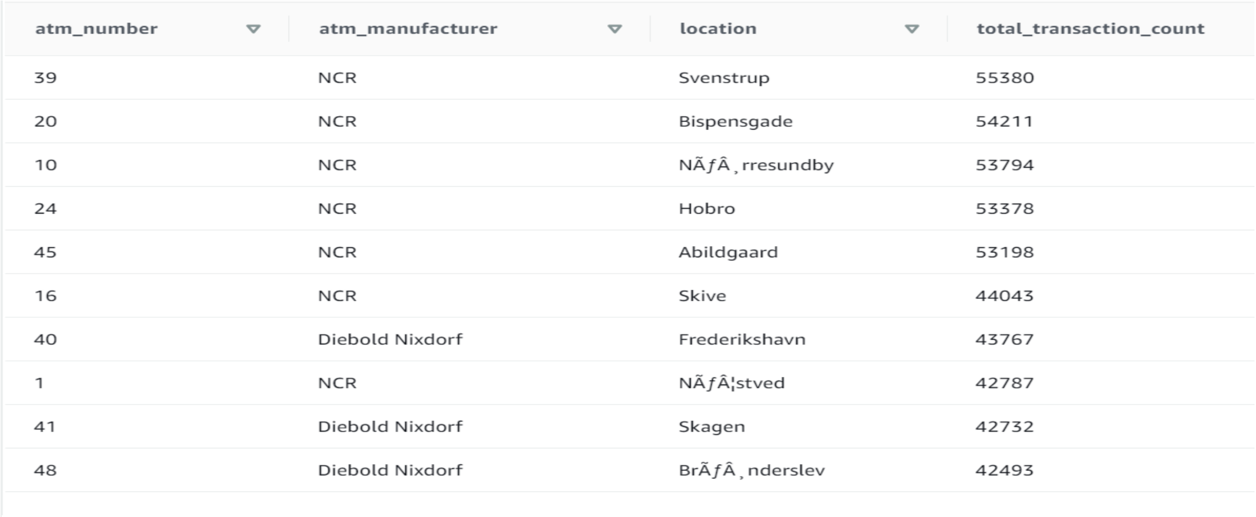
weather_main	total_transaction_count	inactive_count	inactive_count_percent
Snow	23405	4813	20.56
Fog	18174	3729	20.52
Clouds	1181901	194027	16.42
Rain	545135	86017	15.78
Clear	543949	85531	15.72
Mist	82801	12864	15.54
Thunderstorm	2549	361	14.16
Drizzle	62530	8670	13.87
TORNADO	38	1	2.63
Haze	3	0	0.00

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

<Query>

```
SELECT top 10 atm_number, atm_manufacturer, location, count(*) as total_transaction_count
FROM transactions.fact_atm_trans as ft
JOIN transactions.dim_atm as dt ON ft.atm_id = dt.atm_id
JOIN transactions.dim_location as dl ON ft.weather_loc_id = dl.location_id
GROUP BY atm_number, atm_manufacturer, location
ORDER BY total_transaction_count DESC;
```

<Screenshot of the resultant table>



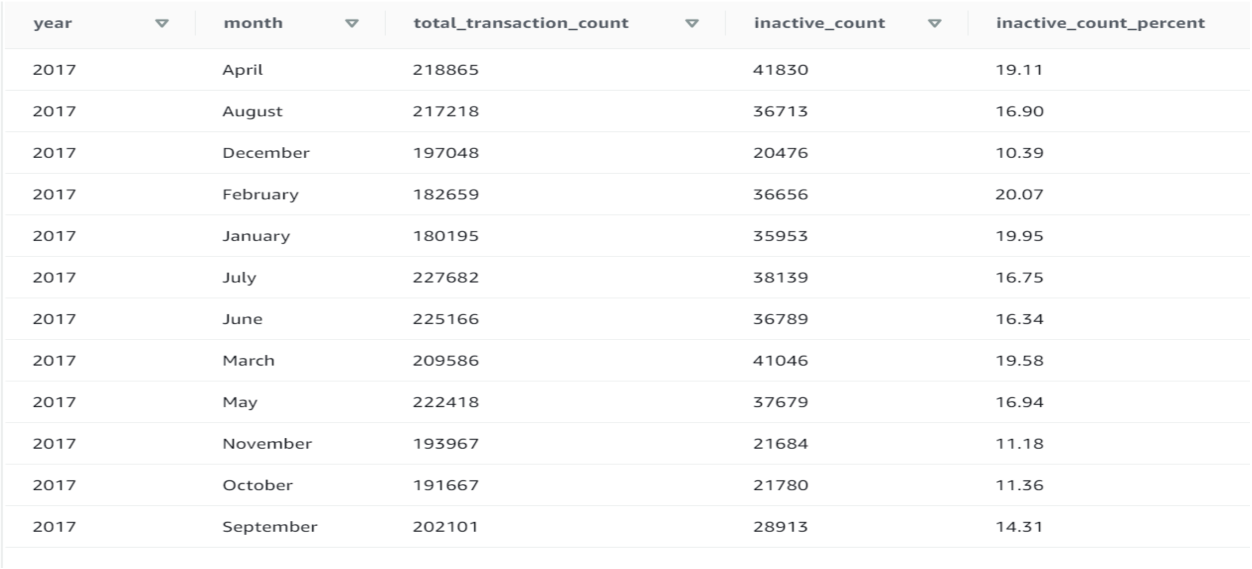
atm_number	atm_manufacturer	location	total_transaction_count
39	NCR	Svenstrup	55380
20	NCR	Bispensgade	54211
10	NCR	NÃƒÂ¸rresundby	53794
24	NCR	Hobro	53378
45	NCR	Abildgaard	53198
16	NCR	Skive	44043
40	Diebold Nixdorf	Frederikshavn	43767
1	NCR	NÃƒÂ¦stved	42787
41	Diebold Nixdorf	Skagen	42732
48	Diebold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

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

<Query>

```
SELECT year, month, count(*) as total_transaction_count, sum(CASE WHEN atm_status =
'Inactive' THEN 1 ELSE 0 END) as inactive_count,
ROUND(((inactive_count*100.0)/total_transaction_count),2) as inactive_count_percent
FROM transactions.fact_atm_trans as ft
JOIN transactions.dim_date as dat ON ft.date_id = dat.date_id
GROUP BY year, month
ORDER BY month;
```

<Screenshot of the resultant table>



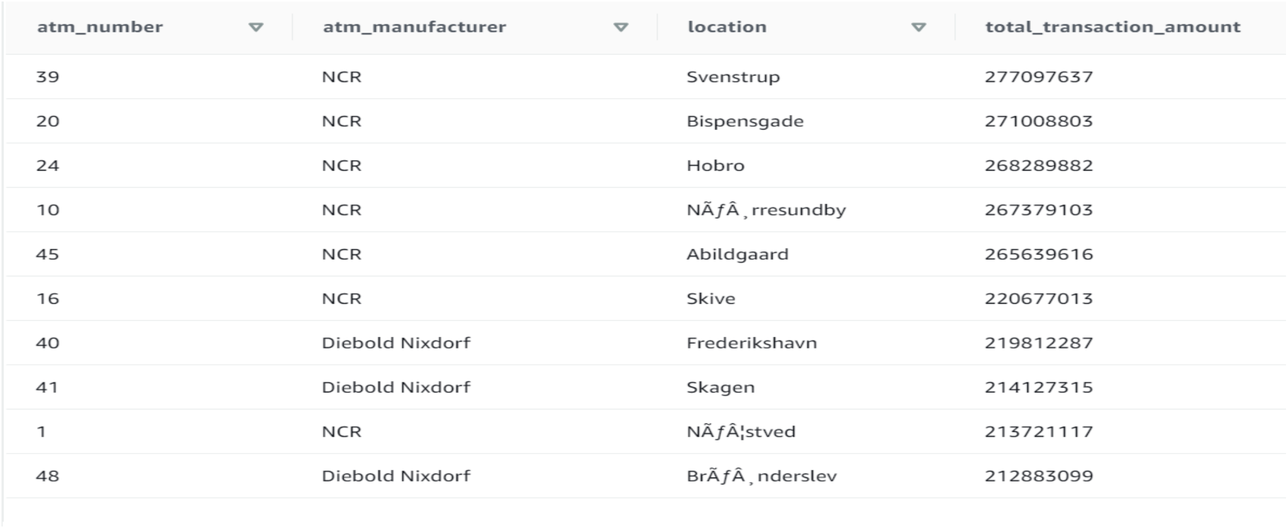
year	month	total_transaction_count	inactive_count	inactive_count_percent
2017	April	218865	41830	19.11
2017	August	217218	36713	16.90
2017	December	197048	20476	10.39
2017	February	182659	36656	20.07
2017	January	180195	35953	19.95
2017	July	227682	38139	16.75
2017	June	225166	36789	16.34
2017	March	209586	41046	19.58
2017	May	222418	37679	16.94
2017	November	193967	21684	11.18
2017	October	191667	21780	11.36
2017	September	202101	28913	14.31

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

<Query>

SELECT top 10 atm_number, atm_manufacturer, location, sum(transaction_amount) as total_transaction_amount
FROM transactions.fact_atm_trans as ft
JOIN transactions.dim_atm as dt ON ft.atm_id = dt.atm_id
JOIN transactions.dim_location as dl ON ft.weather_loc_id = dl.location_id
GROUP BY atm_number, atm_manufacturer, location
ORDER BY total_transaction_amount DESC;

<Screenshot of the resultant table>



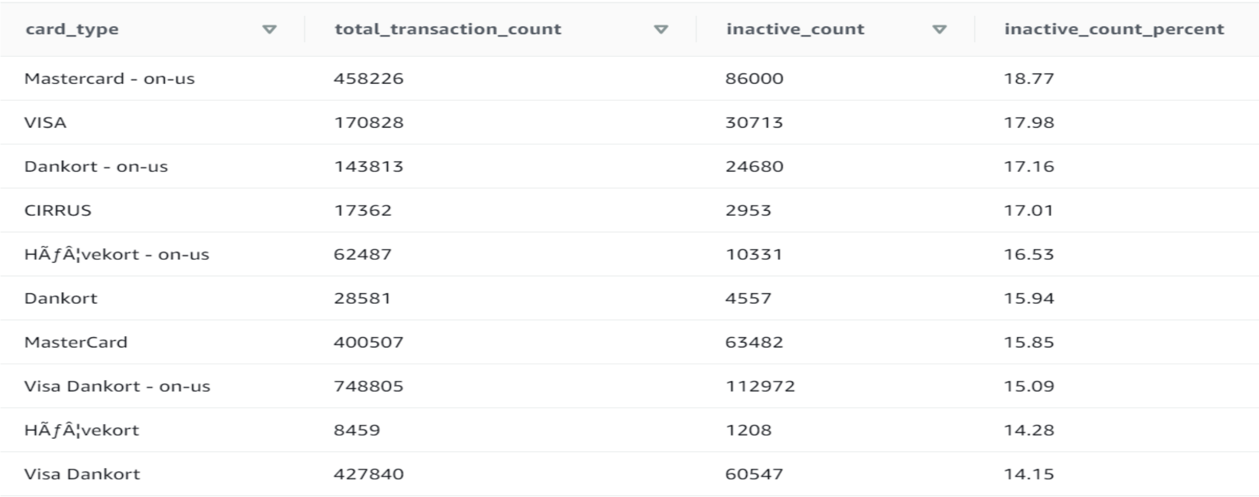
atm_number	atm_manufacturer	location	total_transaction_amount
39	NCR	Svenstrup	277097637
20	NCR	Bispensgade	271008803
24	NCR	Hobro	268289882
10	NCR	NÃƒÂ¸rresundby	267379103
45	NCR	Abildgaard	265639616
16	NCR	Skive	220677013
40	Diebold Nixdorf	Frederikshavn	219812287
41	Diebold Nixdorf	Skagen	214127315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

6. Number of failed ATM transactions across various card types

<Query>

```
SELECT card_type, count(*) as total_transaction_count,  sum(CASE WHEN atm_status =
'Inactive' THEN 1 ELSE 0 END) as inactive_count,
ROUND(((inactive_count*100.0)/total_transaction_count),2) as inactive_count_percent
FROM transactions.fact_atm_trans as ft
JOIN transactions.dim_card_type as ct ON ft.card_type_id = ct.card_type_id
GROUP BY card_type
ORDER BY inactive_count_percent DESC;
```

<Screenshot of the resultant table>



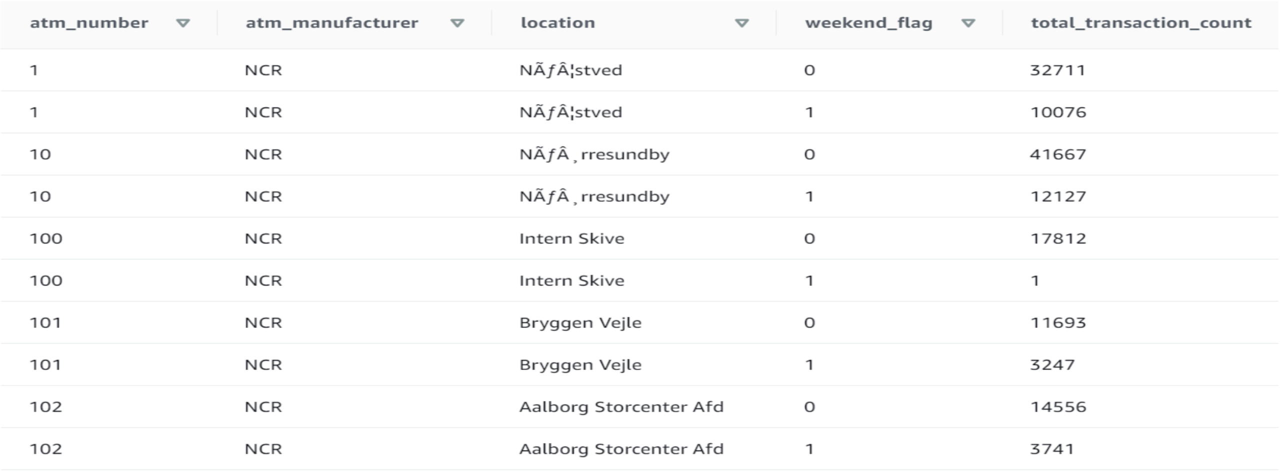
card_type	total_transaction_count	inactive_count	inactive_count_percent
Mastercard - on-us	458226	86000	18.77
VISA	170828	30713	17.98
Dankort - on-us	143813	24680	17.16
CIRRUS	17362	2953	17.01
HÃƒÂ¦vekort - on-us	62487	10331	16.53
Dankort	28581	4557	15.94
MasterCard	400507	63482	15.85
Visa Dankort - 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**

<Query>

```
SELECT top 10 atm_number, atm_manufacturer, location, (CASE WHEN (weekday = 'Saturday'
OR weekday = 'Sunday') THEN 1 ELSE 0 END) as weekend_flag,  count(*) as
total_transaction_count
FROM transactions.fact_atm_trans as ft
JOIN transactions.dim_atm as at ON ft.atm_id = at.atm_id
JOIN transactions.dim_location as dl ON ft.weather_loc_id = dl.location_id
JOIN transactions.dim_date as dt ON ft.date_id = dt.date_id
GROUP BY atm_number, atm_manufacturer, location, weekend_flag
ORDER BY atm_number, atm_manufacturer, location, weekend_flag, total_transaction_count
```

<Screenshot of the resultant table>



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

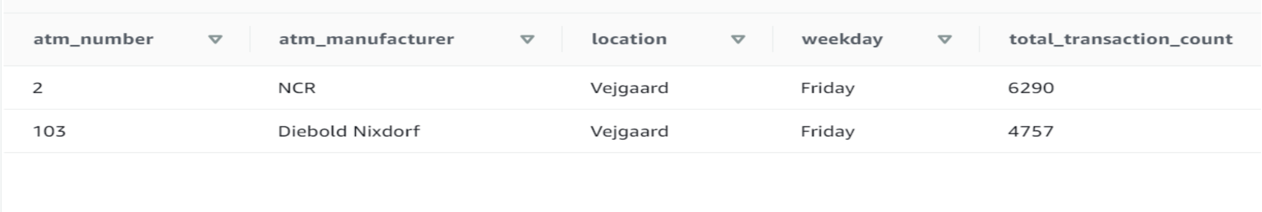
8. Most active day in each ATMs from location "Vejgaard"

<Query>

```
create view temp_trans as (SELECT atm_number, atm_manufacturer, location, weekday,
count(*) as total_transaction_count
FROM transactions.fact_atm_trans as ft
JOIN transactions.dim_atm as at ON ft.atm_id = at.atm_id
JOIN transactions.dim_location as dl ON ft.weather_loc_id = dl.location_id
JOIN transactions.dim_date as dt ON ft.date_id = dt.date_id
WHERE location = 'Vejgaard'
GROUP BY atm_number, atm_manufacturer, location, weekday
ORDER BY atm_number, atm_manufacturer, location, weekday, total_transaction_count)


SELECT atm_number, atm_manufacturer, location, weekday, total_transaction_count
FROM temp_trans
WHERE total_transaction_count in (SELECT max(total_transaction_count)
                              FROM temp_trans
                              GROUP BY atm_number, atm_manufacturer, location)
```

<Screenshot of the resultant table>



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