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**

```
SELECT d.atm_number,
d.atm_manufacturer,
l.location,
count(atm_status) as inactive_count,
count(f.atm_id) as total_transaction_count,
(inactive_count / total_transaction_count * 100) as count_percent
FROM atm_trans_fact f
INNER JOIN atm_dim d
ON (f.atm_id = d.atm_id)
INNER JOIN location_dim l on (l.location_id = d.atm_location_id)
WHERE atm_status = "Inactive"
GROUP BY d.atm_number, d.atm_manufacturer,l.location
ORDER BY inactive_count desc limit 10;
```

atm_number	atm_manufacturer	location	inactive_count	total_transaction_count	count_percent
16	NCR	Skive	44043	44043	100
12	NCR	ÃƒËœesterÃƒÂ¥ Duus	33982	33982	100
2	NCR	Vejgaard	33725	33725	100
88	NCR	Storcenter indg. A	32183	32183	100
47	NCR	Frederiksberg	30883	30883	100
52	NCR	Intern Hjallerup	27361	27361	100
50	NCR	Aarhus	23416	23416	100
29	NCR	Skelagervej 15	20773	20773	100
81	NCR	Spar KÃƒÂ¸bmand TornhÃƒÂ¸j	20148	20148	100
102	NCR	LÃƒÂ¸gstÃƒÂ¸r	18297	18297	100

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

```
SELECT f.weather_main,
COUNT(f.trans_id) as total_transaction_count,
COUNT(
  CASE WHEN f.atm_status = "Inactive" THEN 1
  END) as inactive_count,
(Cast(((inactive_count * 100.00)/total_transaction_count) as decimal(18,2))) as
inactive_count_percent
FROM
atm_trans_fact f
GROUP BY f.weather_main
ORDER BY inactive_count desc;
```

weather_main	total_transcation_count	inactive_count	inactive_count_percent
Clouds	1181901	194027	16.41
Rain	545135	86017	15.77
Clear	543949	85531	15.72
Mist	82801	12864	15.53
Drizzle	62530	8670	13.86
Snow	23405	4813	20.56
Fog	18174	3729	20.51
Thunderstorm	2549	361	14.16
TORNADO	38	1	2.63
Haze	3	0	0.00

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

```
SELECT d.atm_number,
d.atm_manufacturer,
l.location,
COUNT(f.atm_id) AS total_transcation_count
FROM atm_trans_fact f
INNER JOIN atm_dim d ON (f.atm_id = d.atm_id)
INNER JOIN location_dim l ON(l.location_id = d.atm_location_id)
GROUP BY d.atm_number, d.atm_manufacturer, l.location
ORDER BY total_transaction_count DESC
LIMIT 10;
```

Search rows

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	StÃƒÂ¦vring	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(f.trans_id) AS total_transaction_count,
COUNT(CASE WHEN f.atm_status = “Inactive” THEN 1 END) AS inactive_count,
CAST(((inactive_count * 100) / total_transaction_count) as decimal(18,2))) AS
inactive_count_percent
FROM atm_trans_fact f
INNER JOIN date_dim d
ON (f.date_id = d.date_id)
GROUP BY d.month, d.year
ORDER BY month
```

year	month	total_transcation_count	inactive_count	inactive_count_percent
2017	Apr	203352	33591	16.51
2017	Aug	210830	33972	16.11
2017	Dec	204674	30505	14.90
2017	Feb	187956	29862	15.88
2017	Jan	225455	37790	16.76
2017	Jul	219626	36522	16.62
2017	Jun	218172	36535	16.74
2017	Mar	204704	31194	15.23
2017	May	206177	34644	16.80
2017	Nov	190046	30268	15.92

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

```
SELECT d.atm_number,
d.atm_manufacturer,
l.location,
SUM(transaction_amount) AS total_transaction_amount
FROM atm_trans_fact f
INNER JOIN atm_dim d ON (f.atm_id = d.atm_id)
INNER JOIN location_dim l ON (l.location_id = d.atm_location_id)
GROUP BY d.atm_number, d.atm_manufacturer, l.location
ORDER BY total_transaction_amount DESC;
```

atm_number	atm_manufacturer	location	total_transaction_a
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	StÃƒÂ¸vring	213721117
48	Diebold Nixdorf	BrÃƒÂ¸nderslev	212883099

6. **Number of failed ATM transactions across various card types**

```
SELECT d.card_type,
COUNT(f.trans_id) AS total_transaction_count,
COUNT(CASE WHEN f.atm_status = 'Inactive' THEN 1 END) AS inactive_count,
CAST(((inactive_count * 100) / total_transaction_count) as decimal(18,2))) AS
inactive_count_percent
FROM atm_trans_fact f
INNER JOIN card_type_dim d ON (f.card_type = d.card_type_id)
GROUP BY d.card_type
ORDER BY inactive_count_desc;
```

card_type	total_transcation_count	inactive_count	inactive_count_percent
Visa Dankort - on-us	748805	112972	15.08
Mastercard - on-us	458226	86000	18.76
MasterCard	400507	63482	15.85
Visa Dankort	427840	60547	14.15
VISA	170828	30713	17.97
Dankort - on-us	143813	24680	17.16
HÃƒÂ¦vekort - on-us	62487	10331	16.53
Dankort	28581	4557	15.94
CIRRUS	17362	2953	17.00
HÃƒÂ¦vekort	8459	1208	14.28

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 d.atm_number,
d.atm_manufacturer,
l.location,
CASE WHEN dd.weekday = “Sunday” OR dd.weekday = “Saturday” THEN 1 ELSE 0 END
AS weekend_flag,
COUNT(f.trans_id) AS total_transaction_count,
FROM atm_trans_fact f
INNER JOIN atm_dim d ON (f.atm_id = d,atm_id)
INNER JOIN location_dim l ON (l.location_id = d.atm_location_id)
INNER JOIN date_dim dd ON (f.date_id = dd.date_id)
GROUP BY d.atm_number, d.atm_manufacturer, l.location, weekend_flag
ORDER BY d.atm_number, d.atm_manufacturer, l.location, weekend_flag,
total_transaction_count DESC LIMIT 10;
```

atm_number	atm_manufacturer	location	weekend_flag	total_transaction_count
1	NCR	StÃƒÂ¸vring	0	31268
1	NCR	StÃƒÂ¸vring	1	11519
10	NCR	NÃƒÂ¦rresundby	0	38899
10	NCR	NÃƒÂ¦rresundby	1	14895
100	NCR	Intern Skive	0	16635
100	NCR	Intern Skive	1	4957
101	NCR	Bryggen Vejle	0	10930
101	NCR	Bryggen Vejle	1	4010
102	NCR	LÃƒÂ¥gstÃƒÂ¥r	0	13212
102	NCR	LÃƒÂ¥gstÃƒÂ¥r	1	5085

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

```
SELECT a.atm_number, a.atm_manufacturer, l.location, dd.weekday, COUNT(f.trans_id) AS
transaction_count
FROM atm_trans_fact f
INNER JOIN atm_dim a ON (f.atm_id = a.atm_id)
INNER JOIN location_dim l ON (l.location_id = f.weather_loc_id)
INNER JOIN date_dim dd ON (dd.date_id = f.date_id)
```

WHERE l.location = ‘Vejgaard’
GROUP BY a.atm_id, a.atm_manufacturer, l.location, dd.weekday
ORDER BY transaction_count DESC, dd.weekday;

atm_number	atm_manufacturer	location	weekday	transaction_count
2	NCR	Vejgaard	Friday	5369
2	NCR	Vejgaard	Saturday	4969
2	NCR	Vejgaard	Wednesday	4963
2	NCR	Vejgaard	Monday	4793
2	NCR	Vejgaard	Thursday	4759
2	NCR	Vejgaard	Tuesday	4643
2	NCR	Vejgaard	Sunday	4229
103	Diebold Nixdorf	Vejgaard	Tuesday	3288
103	Diebold Nixdorf	Vejgaard	Friday	3256
103	Diebold Nixdorf	Vejgaard	Monday	3192