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 atm.atm_number, atm.atm_manufacturer, loc.location,
        count(fact.transaction_amount) as count_tran_amt
from etl_schema.fact_atm_trans as fact join etl_schema.dim_atm as atm
on fact.atm_id=atm.atm_id
join etl_schema.dim_location as loc
on atm.atm_location_id=loc.location_id
where fact.atm_status='Inactive'
group by atm.atm_number, atm.atm_manufacturer, loc.location
order by count(fact.transaction_amount) desc
limit 10;
```

atm_number	atm_manufacturer	location	count_tran_amt
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**

```
select b.weather_main, b.total_trans_count, a.inactive_count
from (select weather_main, count(1) as inactive_count
      from etl_schema.fact_atm_trans
      where atm_status='Inactive'
      group by weather_main) a right join (select weather_main, count(1) as total_trans_count
                                           from etl_schema.fact_atm_trans
                                           group by weather_main) b
on a.weather_main = b.weather_main
where b.weather_main <> '';
```

weather_main	total_trans_count	inactive_count
Snow	23405	4813
Clouds	1181901	194027
Mist	82801	12864
Rain	545135	86017
Clear	543949	85531
Drizzle	62530	8670
Fog	18174	3729
Thunderstorm	2549	361
TORNADO	38	1
Haze	3	

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

```
select atm.atm_number, atm.atm_manufacturer, loc.location,
        count(fact.transaction_amount) as count_tran_amt
from etl_schema.fact_atm_trans as fact join etl_schema.dim_atm as atm
on fact.atm_id=atm.atm_id
join etl_schema.dim_location as loc
on atm.atm_location_id=loc.location_id
join etl_schema.dim_date dat
on fact.date_id=dat.date_id
group by atm.atm_number, atm.atm_manufacturer, loc.location
order by count(fact.transaction_amount) desc
limit 10;
```

atm_number	atm_manufacturer	location	count_tran_amt
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 dat.year, dat.month, count(fact.trans_id) as count_inactive
from etl_schema.fact_atm_trans fact join
etl_schema.dim_date dat
on fact.date_id=dat.date_id
where fact.atm_status = 'Inactive'
group by dat.year, dat.month
order by count(fact.trans_id) desc;
```

year	month	count_inactive
2017	April	41830
2017	March	41046
2017	July	38139
2017	May	37679
2017	June	36789
2017	August	36713
2017	February	36656
2017	January	35953
2017	September	28913
2017	October	21780
2017	November	21684
2017	December	20476

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 **Top 10 ATMs with the highest total withdrawn amount throughout the year**

```
select atm.atm_number, atm.atm_manufacturer, loc.location,
            sum(fact.transaction_amount) as count_tran_amt
from etl_schema.fact_atm_trans as fact join etl_schema.dim_atm as atm
on fact.atm_id=atm.atm_id
join etl_schema.dim_location as loc
on atm.atm_location_id=loc.location_id
join etl_schema.dim_date dat
on fact.date_id=dat.date_id
group by atm.atm_number, atm.atm_manufacturer, loc.location
order by sum(fact.transaction_amount) desc
limit 10;
```

atm_number	atm_manufacturer	location	count_tran_amt
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 a.card_type, b.count_total_trans, a.count_inactive
from (select card.card_type, count(fact.trans_id) as count_inactive
from etl_schema.fact_atm_trans fact join
etl_schema.dim_card_type card
on fact.card_type_id=card.card_type_id
where fact.atm_status='Inactive'
group by card.card_type) a join
(select card.card_type, count(fact.trans_id) as count_total_trans
from etl_schema.fact_atm_trans fact join
etl_schema.dim_card_type card
on fact.card_type_id=card.card_type_id
group by card.card_type) b
on a.card_type=b.card_type;
```

card_type	count_total_trans	count_inactive
HÃƒÂ¦vekort - on-us	62487	10331
HÃƒÂ¦vekort	8459	1208
CIRRUS	17362	2953
VisaPlus	1134	150
Dankort	28581	4557
Visa Dankort - on-us	748805	112972
VISA	170828	30713
Visa Dankort	427840	60547
Mastercard - on-us	458226	86000
MasterCard	400507	63482
Dankort - on-us	143813	24680
Maestro	530	65

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.atm_number, atm.atm_manufacturer, loc.location,
        decode(dat.weekday,'Sunday',0,'Tuesday',1,'Friday',1,'Wednesday',1,'Thursday',1,'Mond
ay',1,'Saturday',0,0) weekflag,
        count(fact.transaction_amount) as count_tran_amt
from etl_schema.fact_atm_trans as fact join etl_schema.dim_atm as atm
on fact.atm_id=atm.atm_id
join etl_schema.dim_location as loc
on atm.atm_location_id=loc.location_id
join etl_schema.dim_date dat
on dat.date_id=fact.date_id
group by atm.atm_number, atm.atm_manufacturer, loc.location,
        decode(dat.weekday,'Sunday',0,'Tuesday',1,'Friday',1,'Wednesday',1,'Thursday',1,'Mond
ay',1,'Saturday',0,0)
order by atm.atm_number, atm.atm_manufacturer, loc.location
limit 10;
```

atm_number	atm_manufacturer	location	weekflag	count_tran_amt
1	NCR	NÃƒÂ¦stved	1	32711
1	NCR	NÃƒÂ¦stved		10076
10	NCR	NÃƒÂ¸rresundby		12127
10	NCR	NÃƒÂ¸rresundby	1	41667
100	NCR	Intern Skive		1
100	NCR	Intern Skive	1	17812
101	NCR	Bryggen Vejle	1	11693
101	NCR	Bryggen Vejle		3247
102	NCR	Aalborg Storcenter Afd	1	14556
102	NCR	Aalborg Storcenter Afd		3741

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

```
select aa.atm_number, aa.atm_manufacturer, aa.location, aa.weekday, aa.count_tran_amt
from
(select atm.atm_number, atm.atm_manufacturer, loc.location, dat.weekday,
        count(fact.transaction_amount) as count_tran_amt
from etl_schema.fact_atm_trans as fact join etl_schema.dim_atm as atm
on fact.atm_id=atm.atm_id
join etl_schema.dim_location as loc
on atm.atm_location_id=loc.location_id
join etl_schema.dim_date dat
on dat.date_id=fact.date_id
where loc.location='Vejgaard'
group by atm.atm_number, atm.atm_manufacturer, loc.location, dat.weekday) aa
join
(select atm_number, max(count_tran_amt) as count_tran_amt
from (select atm.atm_number, atm.atm_manufacturer, loc.location, dat.weekday,
        count(fact.transaction_amount) as count_tran_amt
from etl_schema.fact_atm_trans as fact join etl_schema.dim_atm as atm
on fact.atm_id=atm.atm_id
join etl_schema.dim_location as loc
on atm.atm_location_id=loc.location_id
join etl_schema.dim_date dat
on dat.date_id=fact.date_id
where loc.location='Vejgaard'
group by atm.atm_number, atm.atm_manufacturer, loc.location, dat.weekday)
group by atm_number) bb
on aa.atm_number=bb.atm_number and aa.count_tran_amt=bb.count_tran_amt;
```

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