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 t1.atm_number,t1.atm_manufacturer,t3.location,count(t2.trans_id) as
total_transaction_count, count(t2.atm_status) as inactive_count,
round((1.0*(100 *  inactive_count)/total_transaction_count),2) as inactive_count_percent
from redshift_etl_project.fact_atm_trans t2
join redshift_etl_project.dim_atm t1
on t2.atm_id = t1.atm_id
join redshift_etl_project.dim_location t3
on t2.location_id = t3.location_id
where t2.atm_status = 'Inactive'
group by atm_number,t1.atm_manufacturer,t3.location
order by total_transaction_count desc;
```

<Screenshot of the resultant table>

atm_number	atm_manufacturer	location	total_transaction_count	inative_count	ount_per cent
16	NCR	Skive	44043	44043	100.0
12	NCR	Ã¯sterÃ¥ Duus	33982	33982	100.0
2	NCR	Vejgaard	33725	33725	100.0
88	NCR	Storcenter indg. A	32183	32183	100.0
30	NCR	NykÃ¸bing Mors	30883	30883	100.0
52	NCR	FarsÃ¸	27361	27361	100.0
50	NCR	Aarhus	23416	23416	100.0
29	NCR	Skelagervej 15	20773	20773	100.0
81	NCR	Spar KÃ¸bmand TornhÃ¸j	20148	20148	100.0
102	NCR	Aalborg Storcenter Afd	18297	18297	100.0

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

<Query>

```
select T11.weather_main as weather_main ,T11.total_transaction_count as
total_transaction_count,T12.inactive_count as inactive_count ,
        round(((cast(inactive_count as float)/cast( total_transaction_count  as float))*100),2) as
inactive_count_percent
        from
   ((select T2.weather_main as weather_main ,count(T2.transaction_amount) as
total_transaction_count
   from redshift_etl_project.fact_atm_trans as T2
   group by T2.weather_main
   ) T11
    inner join
   (select T1.weather_main,count(T1.transaction_amount )as inactive_count
   from redshift_etl_project.fact_atm_trans as T1
   where
   T1.atm_status = 'Inactive'
   group by T1.weather_main
   ) T12
on T11.weather_main=T12.weather_main)
order by inactive_count_percent desc
limit 10;
```

<Screenshot of the resultant table>

weather_main	total_transaction_count	inative_count	inative_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

<Query>

```
select t1.atm_number,t1.atm_manufacturer,t3.location,count(t2.transaction_amount) as
total_transaction_count
from redshift_etl_project.fact_atm_trans t2
join redshift_etl_project.dim_atm t1
on t2.atm_id = t1.atm_id
join redshift_etl_project.dim_location t3
on t2.location_id = t3.location_id
group by atm_number,t1.atm_manufacturer,t3.location
order by total_transaction_count desc
limit 10;
```

<Screenshot of the resultant table>

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

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

<Query>

```
Select t4.year, t4.month,
count(t2.atm_status) as total_trascation_count,
sum(case when t2.atm_status = 'Inactive' then 1 else 0 end)  as inactive_count,
round((1.0*(100 * sum(case when t2.atm_status = 'Inactive' then 1 else 0
end))/count(t2.atm_status)),4) as inactive_count_percent
from redshift_etl_project.fact_atm_trans t2, redshift_etl_project.dim_date t4
where t2.date_id = t4.date_id
group by t4.month , t4.year
order by inactive_count_percent desc;
```

<Screenshot of the resultant table>

year	month	total_transaction_count	inative_count	inative_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
2017	October	191667	21780	11.3600
2017	September	202101	28913	14.3000

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

<Query>

```
select t1.atm_number ,t1.atm_manufacturer,t3.location,sum(t2.transaction_amount) as
total_transaction_amount
from redshift_etl_project.fact_atm_trans t2
join redshift_etl_project.dim_atm t1
on t2.atm_id = t1.atm_id
join redshift_etl_project.dim_location t3
on t2.location_id = t3.location_id
group by atm_number,t1.atm_manufacturer,t3.location
order by total_transaction_amount desc
limit 10;
```

<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	NÃ¸rresundby	267379103
45	NCR	Abildgaard	265639616
16	NCR	Skive	220677013
40	Diebold Nixdorf	Frederikshavn	219812287
41	Diebold Nixdorf	Skagen	214127315
1	NCR	NÃ¦stved	213721117
48	Diebold Nixdorf	BrÃ¸nderslev	212883099

6. Number of failed ATM transactions across various card types

<Query>

```
select d.card_type,count(*) as total_transaction_count,count( decode (atm_status,'Inactive',1))
 as Inactive_count ,round(((cast(inactive_count as float)/cast( total_transaction_count  as
float))*100),2) as inactive_count_percent
 from redshift_etl_project.fact_atm_trans e join redshift_etl_project.dim_card_type d on
d.card_type_id=e.card_type_id
 group by d.card_type
 order by total_transaction_count desc;
```

<Screenshot of the resultant table>

card_type	total_transaction_count	inative_count	inative_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¦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¦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**

<Query>

```
select t1.atm_number,t1.atm_manufacturer,t2.location,
(case when t4.weekday in('Sunday','Saturday') then 1 else 0 end) as weekend_flag,
count(t3.trans_id) as total_transaction_count
from redshift_etl_project.fact_atm_trans t3
join redshift_etl_project.dim_atm t1
on t1.atm_id=t3.atm_id
join redshift_etl_project.dim_location t2
on t3.location_id=t2.location_id
join redshift_etl_project.dim_date t4
on t3.date_id=t4.date_id
group by t1.atm_number,t1.atm_manufacturer,t2.location,weekend_flag
order by t1.atm_number,t1.atm_manufacturer,t2.location,weekend_flag, total_transaction_count
limit 10;
```

<Screenshot of the resultant table>

atm_number	atm_manufacturer	location	weekend_flag	total_transaction_count
1	NCR	NÃ¦stved	0	32711
1	NCR	NÃ¦stved	1	10076
10	NCR	NÃ¸rresundby	0	41667
10	NCR	NÃ¸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>

```
select t1.atm_number as
atm_number,t1.atm_manufacturer,t3.location,t4.weekday,count(t2.transaction_amount) as
total_transaction_count
from redshift_etl_project.fact_atm_trans t2
join redshift_etl_project.dim_atm t1
on t2.atm_id = t1.atm_id
join redshift_etl_project.dim_location t3
on t2.location_id = t3.location_id
join redshift_etl_project.dim_date t4
on t2.date_id = t4.date_id
where t3.location = 'Vejgaard' and  t4.weekday = (select weekday from
redshift_etl_project.dim_date group by weekday order by count(weekday) desc limit 1)
group by atm_number,t1.atm_manufacturer,t3.location,t4.weekday
order by total_transaction_count asc
limit 2;
```

<Screenshot of the resultant table>

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