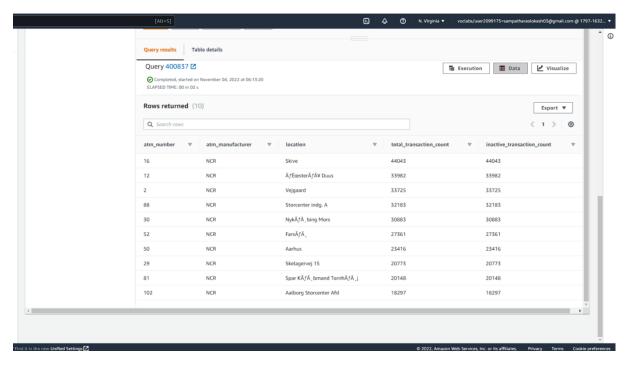
Solving analytical queries on Redshift Cluster:

Here, I 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 a.atm_number, a.atm_manufacturer, l.location,
count(trans_id) as total_transaction_count,
sum(case when atm_status = 'lnactive' then 1 else 0 end) as
inactive_transaction_count
from etl_bankatm_data.fact_atm_trans f, etl_bankatm_data.dim_atm a,
etl_bankatm_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;

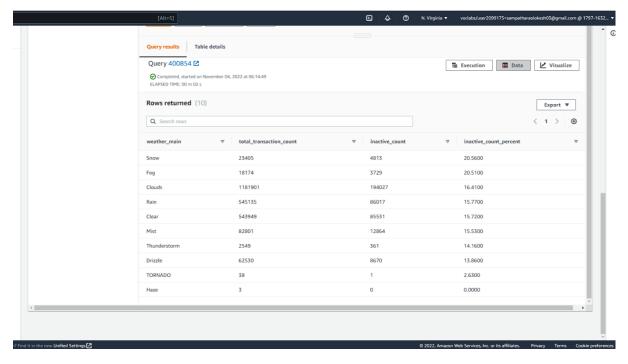
<a href="mailto:screenshot">Screenshot</a> of the resultant table>
```



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

```
<Query>
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_bankatm_data.fact_atm_trans f
where f.weather_main != "
group by f.weather_main
order by inactive_count_percent desc
limit 10;
```

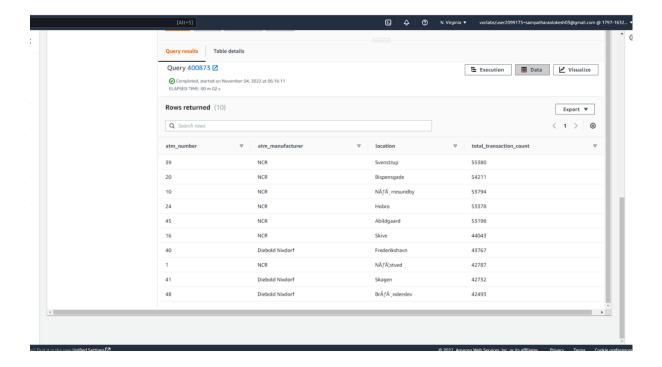
<Screenshot of the resultant table>



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

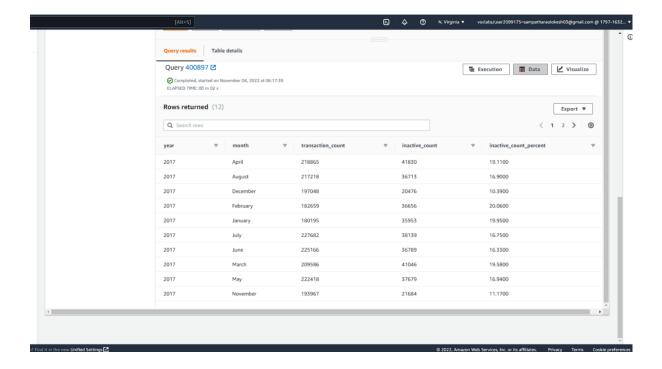
```
<Query>
select a.atm_number, a.atm_manufacturer, l.location,
count(trans_id) as total_transaction_count
from etl_bankatm_data.fact_atm_trans f, etl_bankatm_data.dim_atm a,
etl_bankatm_data.dim_location l
where f.atm_id = a.atm_id and a.atm_location_id = l.location_id
group by a.atm_number, a.atm_manufacturer, l.location
order by total_transaction_count desc
limit 10;
```

<Screenshot of the resultant table>



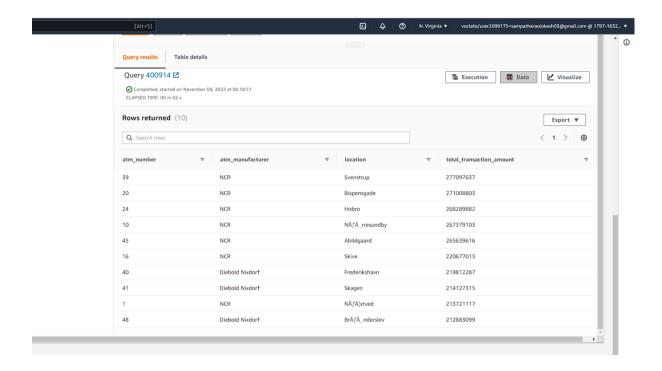
4. Number of overall ATM transactions going inactive per month for each month

```
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_bankatm_data.dim_date
a,etl_bankatm_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_bankatm_data.dim_date
a,etl_bankatm_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;
```



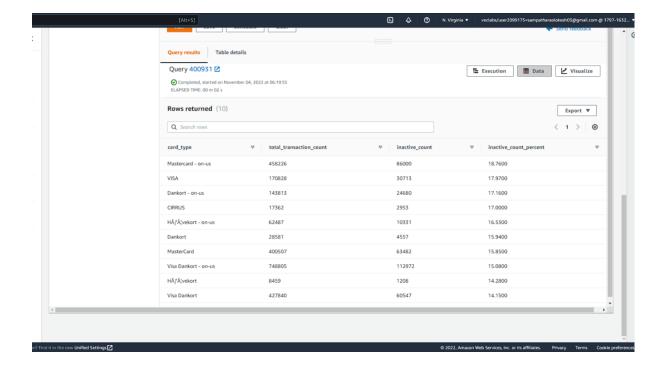
5. Top 10 ATMs with the highest total withdrawn amount throughout the year

select a.atm_number, a.atm_manufacturer, l.location, sum(transaction_amount) as total_transaction_amount from etl_bankatm_data.fact_atm_trans f, etl_bankatm_data.dim_atm a, etl_bankatm_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;



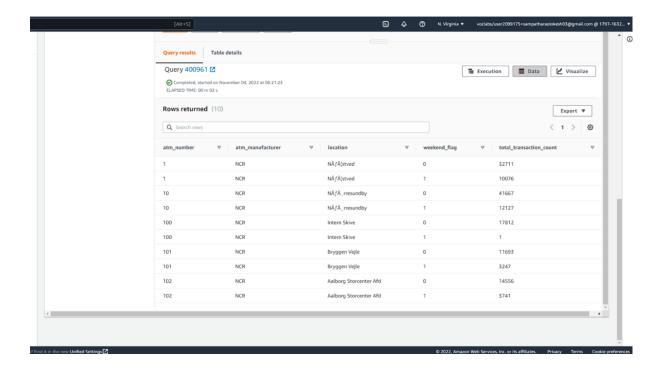
6. Number of failed ATM transactions across various card types

```
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_bankatm_data.fact_atm_trans f, etl_bankatm_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;
```



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_bankatm_data.fact_atm_trans f, etl_bankatm_data.dim_atm a,
etl_bankatm_data.dim_location l,
etl_bankatm_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;
```



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 bankatm data.fact atm trans f inner join etl bankatm data.dim atm a on
f.atm id =
a.atm id
inner join et | bankatm data.dim | location | on a.atm | location | id = | l.location | id
inner join et | bankatm data.dim date d on f.date id = d.date id
where I.location = 'Vejgaard' and d.weekday in
select d.weekday
from etl_bankatm_data.fact_atm_trans f inner join etl_bankatm_data.dim_date d
on f.date id = d.date id
inner join etl_bankatm_data.dim_location I on f.weather_loc_id = I.location_id
where I.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;
```

