Solving analytical queries on Redshift Cluster

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

<Query>

```
select atm_number,
        atm_manufacturer,
        location,
        count(atm_number) as total_transaction_count,
        count(atm_status) as inactive_count
from etl_project.fact_atm_trans left join etl_project.dim_atm
on etl_project.fact_atm_trans.atm_id = etl_project.dim_atm.atm_id
left join etl_project.dim_location
on atm_location_id = location_id
group by atm_number, atm_status, atm_manufacturer, location
having atm_status = 'Inactive'
order by total_transaction_count desc
limit 10;
```

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

<Query>

```
select 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 inactive_count_percent
from etl_project.fact_atm_trans
where weather_main !=''
group by weather_main
order by inactive_count_percent desc;
```

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

<Query>

```
select atm_number,
        atm_manufacturer,
        location,
        count(trans_id) as total_transaction_count
from etl_project.fact_atm_trans left join etl_project.dim_atm
on etl_project.fact_atm_trans.atm_id = etl_project.dim_atm.atm_id
left join etl_project.dim_location
on atm_location_id = location_id
group by atm_number, atm_manufacturer, location
order by total_transaction_count desc
limit 10;
```

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

<Query>

```
select year,
        month,
        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 inactive_count_percent
from etl_project.fact_atm_trans, etl_project.dim_date
where etl_project.fact_atm_trans.date_id = etl_project.dim_date.date_id
group by year, month
order by month;
```

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 **Top 10 ATMs with the highest total withdrawn amount throughout the year**

<Query>

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

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

<Query>

```
select 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 inactive_count_percent
from etl_project.fact_atm_trans, etl_project.dim_card_type
where etl_project.fact_atm_trans.card_type_id = etl_project.dim_card_type.card_type_id
group by card_type
order by inactive_count_percent desc;
```

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 atm_number,
        atm_manufacturer,
        location,
        case when weekday = 'Monday' then 0
                when weekday = 'Tuesday' then 0
                when weekday = 'Wednesday' then 0
                when weekday = 'Thursday' then 0
                when weekday = 'Friday' then 0
                when weekday = 'Saturday' then 1
                else 1
       end as weekend_flag,
       count(trans_id) as total_transaction_count
from etl_project.fact_atm_trans left join etl_project.dim_atm
on etl_project.fact_atm_trans.atm_id = etl_project.dim_atm.atm_id
left join etl_project.dim_location
on atm_location_id = location_id
left join etl_project.dim_date
on etl_project.fact_atm_trans.date_id = etl_project.dim_date.date_id
group by atm_number, atm_manufacturer, location, weekend_flag
order by atm_number, atm_manufacturer, location, weekend_flag, total_transaction_count;
```

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

<Query>

```
select atm_number,
        atm_manufacturer,
        location,
        weekday,
        count(trans_id) as total_transaction_count
from etl_project.fact_atm_trans left join etl_project.dim_atm
on etl_project.fact_atm_trans.atm_id = etl_project.dim_atm.atm_id
left join etl_project.dim_location
on atm_location_id = location_id
left join etl_project.dim_date
on etl_project.fact_atm_trans.date_id = etl_project.dim_date.date_id
where location = 'Vejgaard' and
weekday in (select weekday
                from etl_project.fact_atm_trans left join etl_project.dim_atm
                on etl_project.fact_atm_trans.atm_id = etl_project.dim_atm.atm_id
                left join etl_project.dim_location
                on atm_location_id = location_id
                left join etl_project.dim_date
                on etl_project.fact_atm_trans.date_id = etl_project.dim_date.date_id
                where location = 'Vejgaard'
                group by weekday
                order by count(trans_id) desc
                limit 1)
group by atm_number, atm_manufacturer,location, weekday
order by total_transaction_count;
```