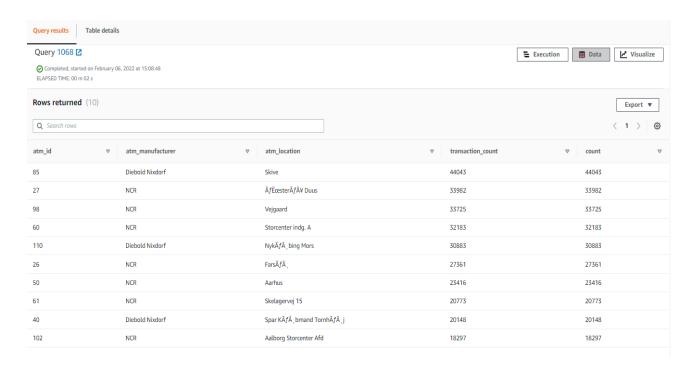




Solving analytical queries on Redshift Cluster

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

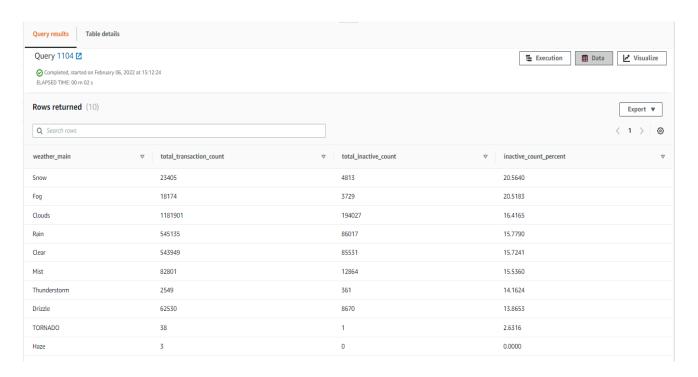






2. Number of ATM failures corresponding to the different weather conditions recorded at the time of the transactions

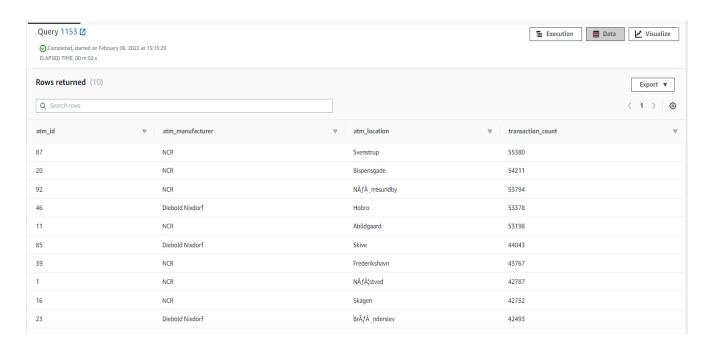
```
SELECT c.weather main
        ,c.total_transaction_count
        ,NVL(d.inactive_count::INT, 0) AS total_inactive_count
        ,round(100.0000 * total_inactive_count / c.total_transaction_count, 4) AS inactive_count_percent
FROM (
        SELECT a.weather_main
                ,count(a.trans_id) AS total_transaction_count
        FROM etl.FACT ATM TRANS a
        WHERE a.weather main != ' '
        GROUP BY a.weather_main
       ) c
LEFT OUTER JOIN (
        SELECT b.weather_main
                ,count(b.atm_status) AS inactive_count
        FROM etl.FACT ATM TRANS b
        WHERE b.atm_status = 'Inactive'
                AND b.weather_main != ' '
        GROUP BY b.weather main
       ) d ON c.weather_main = d.weather_main
GROUP BY c.weather_main
        ,c.total_transaction_count
        ,total_inactive_count
ORDER BY inactive_count_percent DESC;
```







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

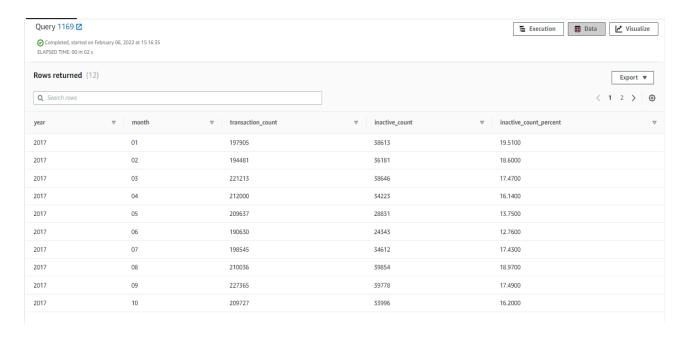






4. Number of overall ATM transactions going inactive per month for each month

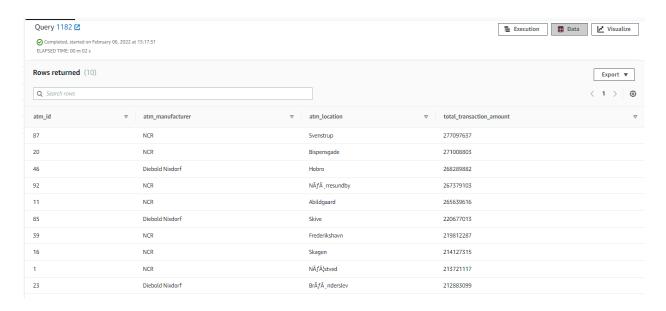
```
SELECT c.year
         ,c.month
         ,c.transaction_count
         ,d.inactive count
         ,CAST(trunc(100.0 * d.inactive_count / c.transaction_count, 2) AS NUMERIC(10, 4)) AS inactive_count_percent
FROM (
         SELECT a.year
                  ,a.month
                  ,count(b.trans_id) AS transaction_count
         FROM etl.DATE a
                  ,etl.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.DATE a
                  ,etl.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

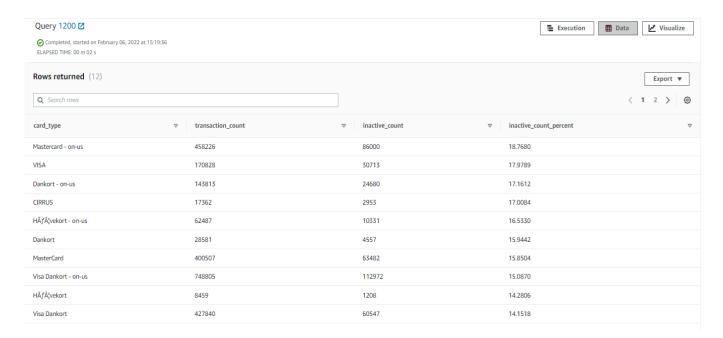






6. Number of failed ATM transactions across various card types

```
SELECT a.card_type
        ,a.transaction_count
        ,b.inactive_count
        ,round(100.0000 * b.inactive_count / a.transaction_count, 4) AS inactive_count_percent
FROM (
        SELECT c.card_type
                 ,count(d.trans_id) AS transaction_count
        FROM etl.card c
                ,etl.FACT ATM TRANS d
        WHERE c.card_type_id = d.card_type_id
        GROUP BY c.card_type
        ) a
LEFT JOIN (
        SELECT c.card_type
                ,count(d.atm_status) AS inactive_count
        FROM etl.card c
                ,etl.FACT_ATM_TRANS d
        WHERE c.card_type_id = d.card_type_id
                AND d.atm status = 'Inactive'
        GROUP BY c.card_type
        ) b ON a.card_type = b.card_type
ORDER BY inactive_count_percent DESC;
```

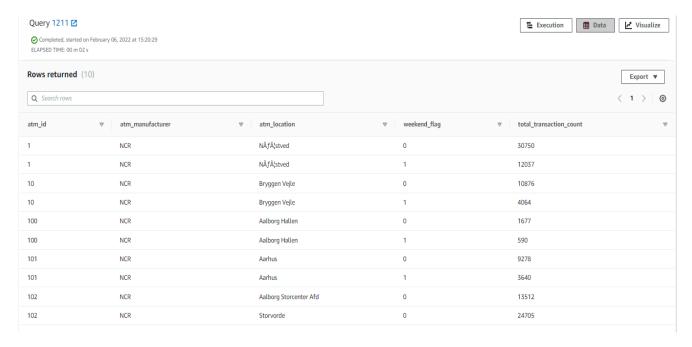






7. Number of transactions happening on an ATM on weekdays and on weekends throughout the year. Order this by the ATM_number, ATM_manufacturer, location, weekend_flag and then total_transaction_count

SELECT a.atm_id, a.atm_manufacturer, b.atm_location, CASE c.weekday WHEN 'Monday' THEN '0' WHEN 'Tuesday' THEN '0' WHEN 'Wednesday' THEN '0' WHEN 'Thursday' THEN '0' WHEN 'Friday' THEN '0' **ELSE '1'** END AS weekend_flag, count(d.trans id) as total transaction count from etl.atm a, etl.loc b, etl.date c, etl.FACT_ATM_TRANS d where d.atm_prim_id = a.atm_prim_id and b.location_id = d.location_id and c.date id = d.date id group by a.atm id, a.atm manufacturer, b.atm location, weekend flag order by a.atm_id asc, weekend_flag asc limit 10







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

```
SELECT atm_id,
atm_manufacturer,
atm_location,
weekday,
total_transaction_count
FROM (
select atm id,
atm_manufacturer,
atm_location,
weekday,
total transaction count,
max(total_transaction_count) over (partition by atm_id) as max_version
from (SELECT a.atm_id, a.atm_manufacturer, b.atm_location, c.weekday,
count(d.trans_id) as total_transaction_count
from etl.atm a, etl.loc b, etl.date c, etl.FACT_ATM_TRANS d
where d.atm_prim_id = a.atm_prim_id
and b.location_id = d.location_id
and b.atm location = 'Vejgaard'
and c.date_id = d.date_id
group by a.atm_id, a.atm_manufacturer, b.atm_location, c.weekday) c
) t
where total_transaction_count = max_version;
```

