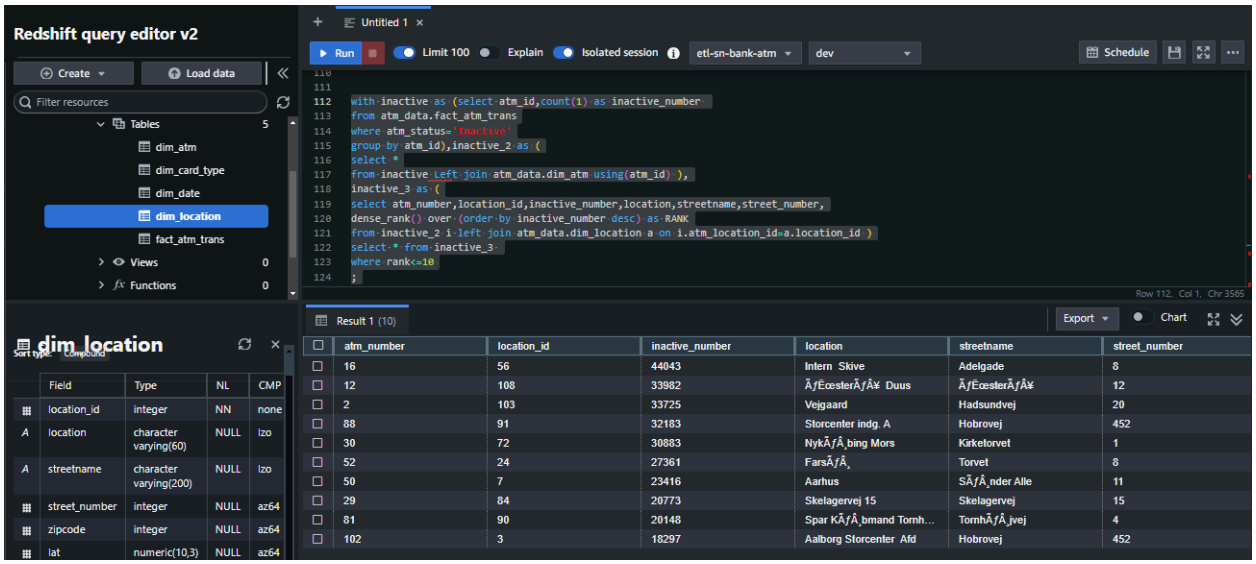
Analytical queries on Redshift Cluster

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

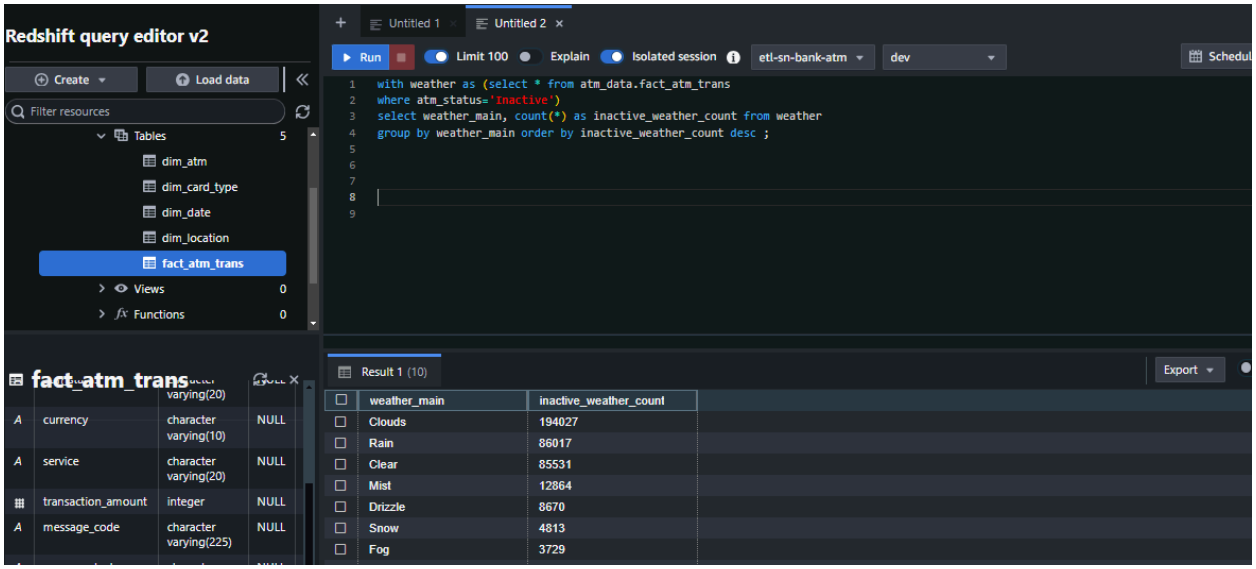
```
WITH inactive AS (
    -- Calculate the count of inactive ATMs
    SELECT
        atm_id,
        COUNT(1) AS inactive_number
    FROM
        atm_data.fact_atm_trans
    WHERE
        atm_status = 'Inactive'
    GROUP BY
        atm_id
),
inactive_2 AS (
    -- Join inactive ATMs with ATM details
    SELECT
        *
    FROM
        inactive
    LEFT JOIN
        atm_data.dim_atm USING (atm_id)
),
inactive_3 AS (
    -- Further join with location details and calculate
dense rank
    SELECT
        atm_number,
        location_id,
        inactive_number,
        location,
        streetname,
        street_number,
        DENSE_RANK() OVER (ORDER BY inactive_number DESC)
AS RANK
    FROM
        inactive_2 i
    LEFT JOIN
```

```
        atm_data.dim_location a ON i.atm_location_id =
a.location_id
)
-- Select top 10 inactive ATMs based on rank
SELECT
    *
FROM
    inactive_3
WHERE
    RANK <= 10;
```



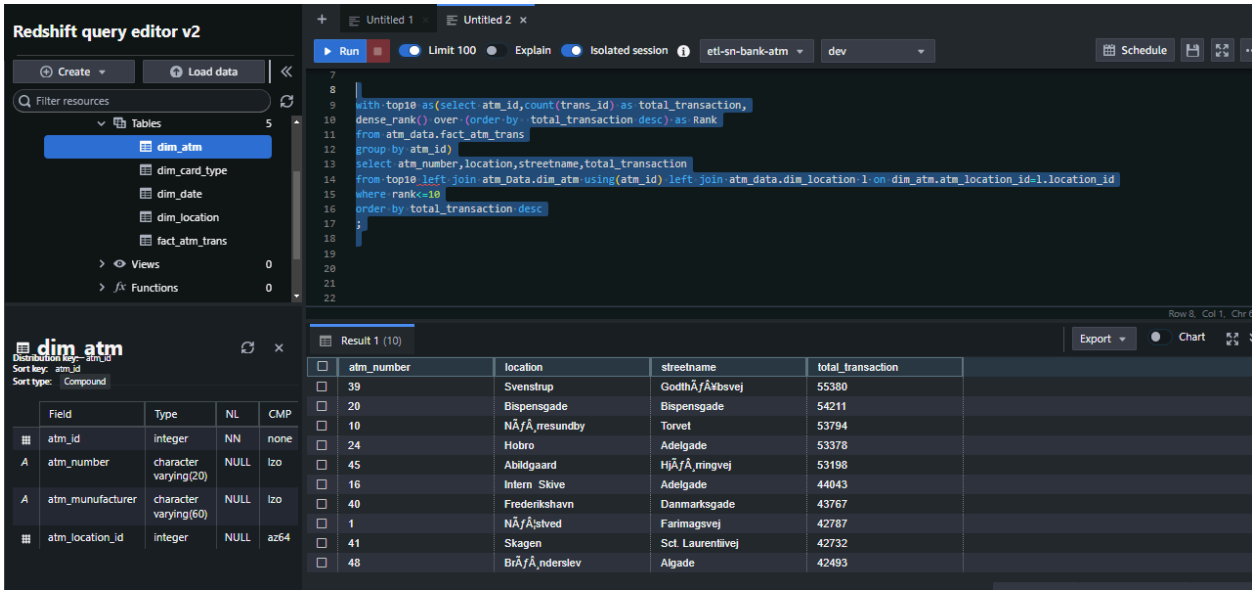
2. Number of ATM failures corresponding to the different weather conditions recorded at the time of the transactions

```
WITH weather AS (
    -- Select weather data for inactive ATMs
    SELECT *
    FROM atm_data.fact_atm_trans
    WHERE atm_status = 'Inactive'
)
-- Count occurrences of each weather condition for
inactive ATMs
SELECT
    weather_main,
    COUNT(*) AS inactive_weather_count
FROM
    weather
GROUP BY
    weather_main
ORDER BY
    inactive_weather_count DESC;
```



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

```
WITH top10 AS (
    -- Calculate total transactions and assign rank to each ATM
    SELECT
        atm_id,
        COUNT(trans_id) AS total_transaction,
        DENSE_RANK() OVER (ORDER BY total_transaction DESC) AS
Rank
    FROM
        atm_data.fact_atm_trans
    GROUP BY
        atm_id
)
-- Select top 10 ATMs and retrieve location details
SELECT
    atm_number,
    location,
    streetname,
    total_transaction
FROM
    top10
LEFT JOIN
    atm_data.dim_atm USING (atm_id)
LEFT JOIN
    atm_data.dim_location l ON dim_atm.atm_location_id =
l.location_id
WHERE
    Rank <= 10
ORDER BY
    total_transaction DESC;
```


Redshift query editor v2
Create
Load data
Filter resources
Tables 5
dim_atm
dim_card_type
dim_date
dim_location
fact_atm_trans
Views 0
Functions 0
Untitled 1
Untitled 2
Run
Limit 100
Explain
Isolated session
etl-sn-bank-atm
dev
Schedule
with top10 as(select atm_id,count(trans_id) as total_transaction,
dense_rank() over (order by total_transaction desc)as Rank
from atm_data.fact_atm_trans
group by atm_id)
select atm_number,location,streetname,total_transaction
from top10 left join atm_Data.dim_atm using(atm_id) left join atm_data.dim_location l on dim_atm.atm_location_id=l.location_id
where rank<=10
order by total_transaction desc
;
Result 1 (10)
Export
Chart
atm_number
location
streetname
total_transaction
39 Svenstrup GodthÃƒÂ¥bsvej 55380
20 Bispensgade Bispensgade 54211
10 NÃƒÂ¸rresundby Torvet 53794
24 Hobro Adelgade 53378
45 Abildgaard HjÃƒÂ¸rringvej 53198
16 Intern Skive Adelgade 44043
40 Frederikshavn Danmarksgade 43767
1 NÃƒÂ¦stved Farimagsvej 42787
41 Skagen Sct. Laurentiivej 42732
48 BrÃƒÂ¸nderslev Algade 42493
dim_atm
Distribution key: atm_id
Sort key: atm_id
Sort type: Compound
Field
Type
NL
CMP
atm_id integer NN none
atm_number character varying(20) NULL lzo
atm_munufacturer character varying(60) NULL lzo
atm_location_id integer NULL az64

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

```
SELECT
    year,
    month,
    COUNT(trans_id) AS
total_inactive_transaction_count
FROM
    atm_data.fact_atm_trans
LEFT JOIN
    atm_data.dim_date USING (date_id)
WHERE
    atm_status = 'Inactive'
GROUP BY
    year,
    Month;
```

```sql
select year,month,count(trans_id) as total_inactive__transaction_count
from atm_data.fact_atm_trans left join atm_data.dim_date using(date_id)
where atm_status='Inactive'
group by year,month
;
```

Result 1 (12)

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

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

```
WITH top10 AS (
    -- Calculate total sum of transaction amounts and
assign rank to each ATM
    SELECT
        atm_id,
        SUM(transaction_amount) AS total_sum_transaction,
        DENSE_RANK() OVER (ORDER BY total_sum_transaction
DESC) AS Rank
    FROM
        atm_data.fact_atm_trans
    GROUP BY
        atm_id
)
-- Select top 10 ATMs and retrieve location details
SELECT
    atm_number,
    location,
    streetname,
    total_sum_transaction
FROM
    top10
LEFT JOIN
    atm_data.dim_atm USING (atm_id)
LEFT JOIN
    atm_data.dim_location l ON dim_atm.atm_location_id =
l.location_id
WHERE
    Rank <= 10
ORDER BY
    total_sum_transaction DESC;
```

```sql
with top10 as(select atm_id,sum(transaction_amount) as total_sum_transaction,
dense_rank() over (order by  total_sum_transaction desc) as Rank
from atm_data.fact_atm_trans
group by atm_id)
select atm_number,location,streetname,total_sum_transaction
from top10 left join atm_Data.dim_atm using(atm_id) left join atm_data.dim_location l on dim_atm.atm_location_id=l.location_id
where rank<=10
order by total_sum_transaction desc
;
```

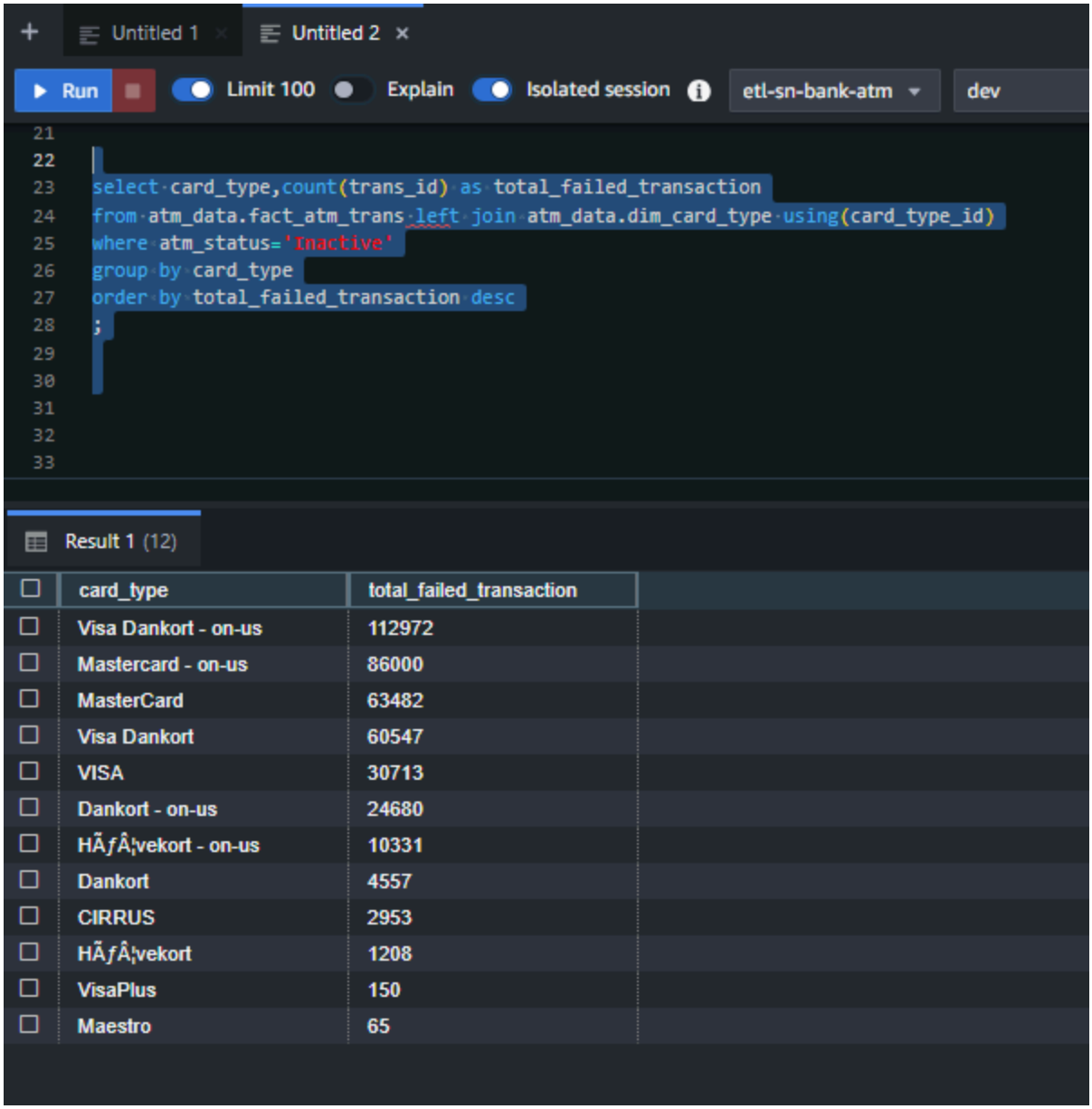
Result 1 (10)

Export

atm_number	location	streetname	total_sum_transaction
39	Svenstrup	GodthÃƒÂ¥bsvej	277097637
20	Bispensgade	Bispensgade	271008803
24	Hobro	Adelgade	268289882
10	NÃƒÂ¸rresundby	Torvet	267379103
45	Abildgaard	HjÃƒÂ¸rringvej	265639616
16	Intern Skive	Adelgade	220677013
40	Frederikshavn	Danmarksgade	219812287
41	Skagen	Sct. Laurentiivej	214127315
1	NÃƒÂ¦stved	Farimagsvej	213721117
48	BrÃƒÂ¸nderslev	Algade	212883099

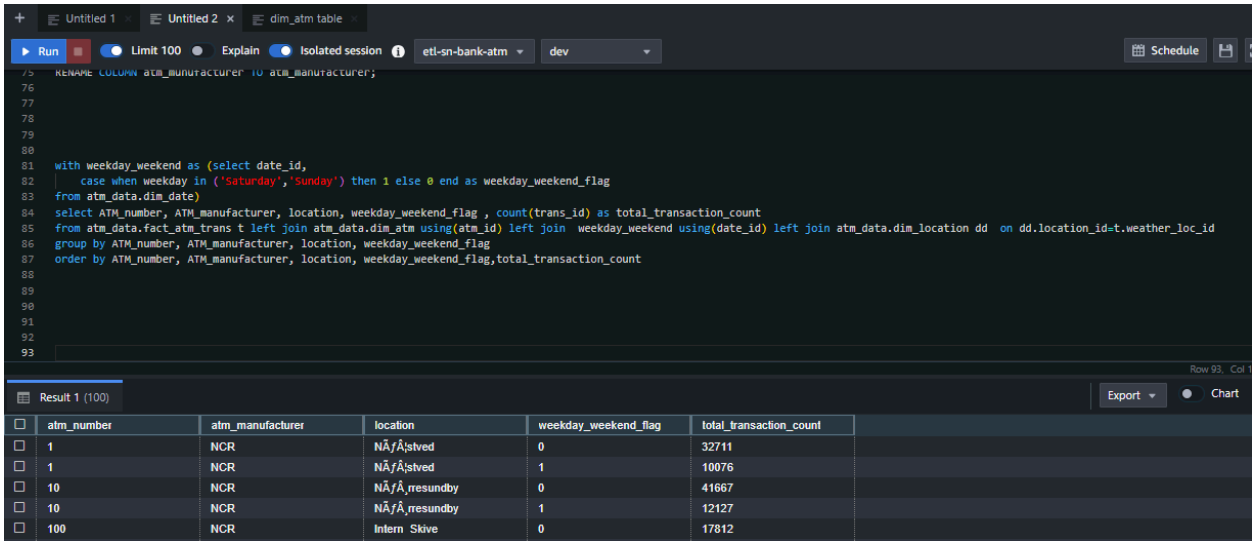
6. Number of failed ATM transactions across various card types

```
SELECT
    card_type,
    COUNT(trans_id) AS total_failed_transaction
FROM
    atm_data.fact_atm_trans
LEFT JOIN
    atm_data.dim_card_type USING (card_type_id)
WHERE
    atm_status = 'Inactive'
GROUP BY
    card_type
ORDER BY
    total_failed_transaction DESC;
```


Untitled 1
Untitled 2
Run
Limit 100
Explain
Isolated session
etl-sn-bank-atm
dev
21
22
23 select card_type,count(trans_id) as total_failed_transaction
24 from atm_data.fact_atm_trans left join atm_data.dim_card_type using(card_type_id)
25 where atm_status='Inactive'
26 group by card_type
27 order by total_failed_transaction desc
28 ;
29
30
31
32
33
Result 1 (12)
card_type
total_failed_transaction
Visa Dankort - on-us 112972
Mastercard - on-us 86000
MasterCard 63482
Visa Dankort 60547
VISA 30713
Dankort - on-us 24680
HÃƒÂ¦vekort - on-us 10331
Dankort 4557
CIRRUS 2953
HÃƒÂ¦vekort 1208
VisaPlus 150
Maestro 65

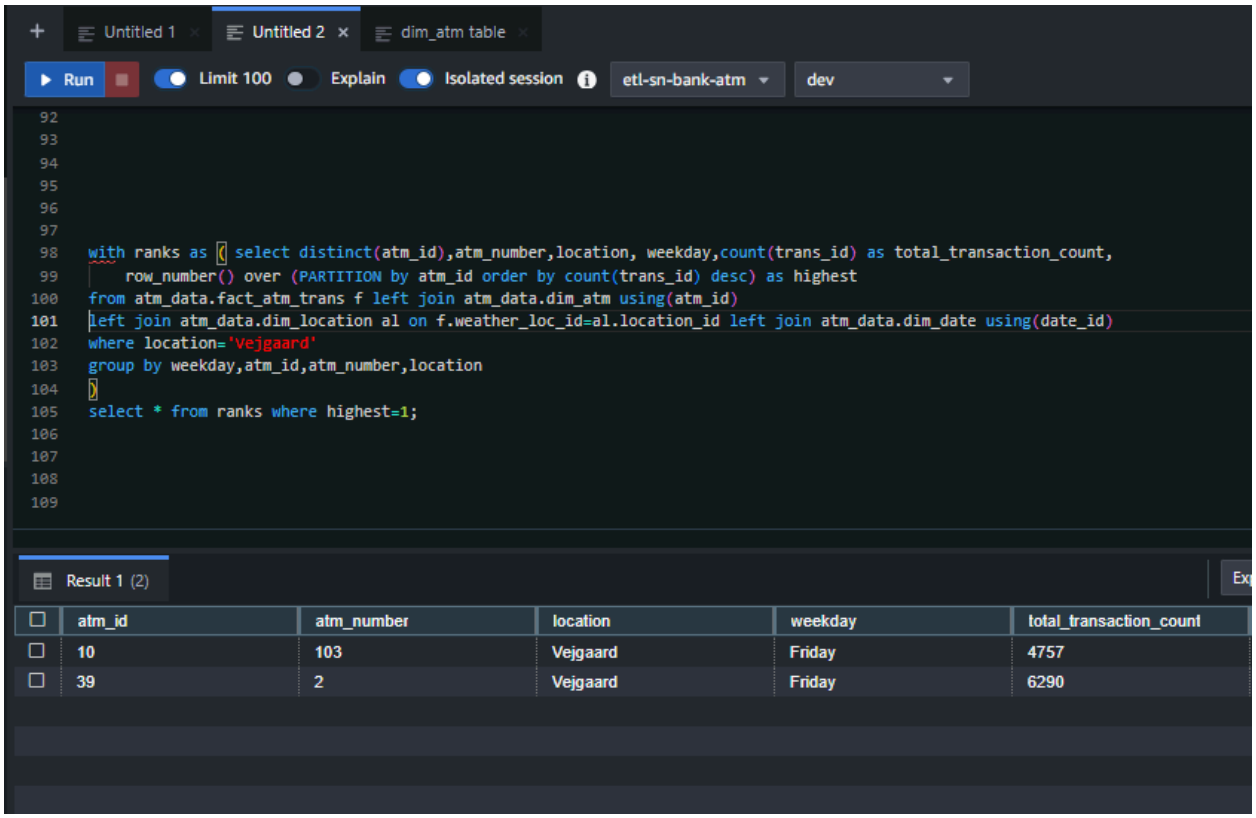
7. **Number of transactions happening on an ATM on weekdays and on weekends throughout the year. Order this by the ATM_number, ATM_manufacturer, location, weekend_flag and then total_transaction_count**

```
WITH weekday_weekend AS (
    -- Assign a flag for weekday and weekend
    SELECT
        date_id,
        CASE WHEN weekday IN ('Saturday', 'Sunday') THEN 1 ELSE
0 END AS weekday_weekend_flag
    FROM
        atm_data.dim_date
)
-- Select transaction details and count total transactions for
each ATM
SELECT
    atm.ATM_number,
    atm.ATM_manufacturer,
    loc.location,
    wd.weekday_weekend_flag,
    COUNT(t.trans_id) AS total_transaction_count
FROM
    atm_data.fact_atm_trans t
LEFT JOIN
    atm_data.dim_atm atm USING (atm_id)
LEFT JOIN
    weekday_weekend wd USING (date_id)
LEFT JOIN
    atm_data.dim_location loc ON loc.location_id =
t.weather_loc_id
GROUP BY
    atm.ATM_number,
    atm.ATM_manufacturer,
    loc.location,
    wd.weekday_weekend_flag
ORDER BY
    atm.ATM_number,
    atm.ATM_manufacturer,
    loc.location,
    wd.weekday_weekend_flag,
    total_transaction_count;
```


Untitled 1
Untitled 2
dim_atm table
Run
Limit 100
Explain
Isolated session
etl-sn-bank-atm
dev
Schedule
RENAME COLUMN atm_manufacturer TO atm_manufacturer;
with weekday_weekend as (select date_id,
case when weekday in ('Saturday','Sunday') then 1 else 0 end as weekday_weekend_flag
from atm_data.dim_date)
select ATM_number, ATM_manufacturer, location, weekday_weekend_flag , count(trans_id) as total_transaction_count
from atm_data.fact_atm_trans t left join atm_data.dim_atm using(atm_id) left join weekday_weekend using(date_id) left join atm_data.dim_location dd on dd.location_id=t.weather_loc_id
group by ATM_number, ATM_manufacturer, location, weekday_weekend_flag
order by ATM_number, ATM_manufacturer, location, weekday_weekend_flag,total_transaction_count
Result 1 (100)
Export
Chart
atm_number
atm_manufacturer
location
weekday_weekend_flag
total_transaction_count
1 NCR NÃƒÂ¦stved 0 32711
1 NCR NÃƒÂ¦stved 1 10076
10 NCR NÃƒÂ¸resundby 0 41667
10 NCR NÃƒÂ¸resundby 1 12127
100 NCR Intern Skive 0 17812

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

```
WITH ranks AS (
    SELECT DISTINCT
        atm_id,
        atm_number,
        location,
        weekday,
        COUNT(trans_id) AS total_transaction_count,
        ROW_NUMBER() OVER (PARTITION BY atm_id
ORDER BY COUNT(trans_id) DESC) AS highest
    FROM
        atm_data.fact_atm_trans f
    LEFT JOIN
        atm_data.dim_atm USING (atm_id)
    LEFT JOIN
        atm_data.dim_location al ON
f.weather_loc_id = al.location_id
    LEFT JOIN
        atm_data.dim_date USING (date_id)
    WHERE
        location = 'Vejgaard'
    GROUP BY
        weekday,
        atm_id,
        atm_number,
        location
)
SELECT * FROM ranks WHERE highest = 1;
```


Untitled 1
Untitled 2
dim_atm table
Run
Limit 100
Explain
Isolated session
etl-sn-bank-atm
dev
92
93
94
95
96
97
98 with ranks as (select distinct(atm_id),atm_number,location, weekday,count(trans_id) as total_transaction_count,
99 row_number() over (PARTITION by atm_id order by count(trans_id) desc) as highest
100 from atm_data.fact_atm_trans f left join atm_data.dim_atm using(atm_id)
101 left join atm_data.dim_location al on f.weather_loc_id=al.location_id left join atm_data.dim_date using(date_id)
102 where location='Vejgaard'
103 group by weekday,atm_id,atm_number,location
104)
105 select * from ranks where highest=1;
106
107
108
109
Result 1 (2)
atm_id
atm_number
location
weekday
total_transaction_count
10
103
Vejgaard
Friday
4757
39
2
Vejgaard
Friday
6290