

ETL Project : ATM Transactions

Redshift Analytical Queries

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

Query :

```
SELECT atm_number, atm_manufacturer, location, COUNT(trans_id) AS
transactions
FROM atm_trans.fact_atm_trans T
INNER JOIN atm_trans.dim_atm A ON T.atm_id = A.atm_id
INNER JOIN atm_trans.dim_location L ON A.atm_location_id = L.location_id
WHERE atm_status = 'Inactive'
GROUP BY atm_number, atm_manufacturer, location
ORDER BY transactions DESC
LIMIT 10;
```

Result :

Rows returned (10)				Export ▼
<input type="text" value="Search rows"/>				< 1 > ⚙
atm_number ▼	atm_manufacturer ▼	location ▼	transactions ▼	
16	NCR	Skive	44043	
12	NCR	Århus	33982	
2	NCR	Vejgaard	33725	
88	NCR	Aalborg Storcenter Afd	32183	
30	NCR	Nykøbing Mors	30883	
52	NCR	Farsø	27361	
50	NCR	Aarhus	23416	
29	NCR	Skelagervej 15	20773	
81	NCR	Spar København Tørhøj	20148	
102	NCR	Aalborg Storcenter indg. D	18297	

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

Query :

```
WITH weather_atm_failure AS
(
  SELECT
    weather_main,
    COUNT(trans_id) AS transactions,
    SUM(CASE WHEN atm_status='Inactive' THEN 1 ELSE 0 END) AS
inactive_count
  FROM
    atm_trans.fact_atm_trans
  WHERE
    weather_main != "
  GROUP BY
    weather_main
)
SELECT
  *,
  ROUND(CAST(inactive_count AS numeric(10,2)) / transactions * 100, 2) AS
inactive_count_percent
FROM
  weather_atm_failure
ORDER BY
  inactive_count_percent DESC;
```

Result:

Rows returned (10)				Export ▼
<input type="text" value="Search rows"/>				< 1 > ⚙
weather_main ▼	transactions ▼	inactive_count ▼	inactive_count_percent ▼	
Snow	23405	4813	20.56	
Fog	18174	3729	20.52	
Clouds	1181901	194027	16.42	
Rain	545135	86017	15.78	
Clear	543949	85531	15.72	
Mist	82801	12864	15.54	
Thunderstorm	2549	361	14.16	
Drizzle	62530	8670	13.87	
TORNADO	38	1	2.63	
Haze	3	0	0.00	

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

Query :

```
SELECT
    atm_number,
    atm_manufacturer,
    location,
    COUNT(trans_id) AS transactions
FROM
    atm_trans.fact_atm_trans T
INNER JOIN
    atm_trans.dim_atm A ON T.atm_id = A.atm_id
INNER JOIN
    atm_trans.dim_location L ON A.atm_location_id = L.location_id
GROUP BY
    atm_number,
    atm_manufacturer,
    location
ORDER BY
    transactions DESC
LIMIT 10;
```

Result:

Rows returned (10)				Export ▼
<input type="text" value="Search rows"/>				< 1 > ⚙
atm_number ▼	atm_manufacturer ▼	location ▼	transactions ▼	
39	NCR	Svenstrup	55380	
20	NCR	Bispensgade	54211	
10	NCR	NÅfÅ, rresundby	53794	
24	NCR	Hobro	53378	
45	NCR	Abildgaard	53198	
16	NCR	Skive	44043	
40	Diebold Nixdorf	Frederikshavn	43767	
1	NCR	NÅfÅ!stved	42787	
41	Diebold Nixdorf	Skagen	42732	
48	Diebold Nixdorf	BrÅfÅ, nderslev	42493	

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

Query :

```
WITH monthwise_atm_failure AS (  
    SELECT  
        month,  
        COUNT(F.trans_id) AS transactions,  
        SUM(CASE WHEN F.atm_status='Inactive' THEN 1 ELSE 0 END) AS  
inactive_count  
    FROM  
        atm_trans.fact_atm_trans F  
    JOIN  
        atm_trans.dim_date D ON F.date_id = D.date_id  
    GROUP BY  
        D.month  
)  
SELECT  
    *,  
    ROUND(CAST(inactive_count AS numeric(10,2)) / transactions * 100, 2) AS  
inactive_count_percent  
FROM  
    monthwise_atm_failure  
ORDER BY  
    inactive_count_percent DESC;
```

Result:

Rows returned (12)				Export ▼
<input type="text" value="Search rows"/>				< 1 2 > ⚙
month ▼	transactions ▼	inactive_count ▼	inactive_count_percent ▼	
February	182659	36656	20.07	
January	180195	35953	19.95	
March	209586	41046	19.58	
April	218865	41830	19.11	
May	222418	37679	16.94	
August	217218	36713	16.90	
July	227682	38139	16.75	
June	225166	36789	16.34	
September	202101	28913	14.31	
October	191667	21780	11.36	

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

Query :

```
SELECT
    atm_number,
    atm_manufacturer,
    location,
    SUM(transaction_amount) AS total_transaction_amount
FROM
    atm_trans.fact_atm_trans T
INNER JOIN
    atm_trans.dim_atm A ON T.atm_id = A.atm_id
INNER JOIN
    atm_trans.dim_location L ON A.atm_location_id = L.location_id
GROUPBY
    atm_number,
    atm_manufacturer,
    location
ORDER BY
    total_transaction_amount DESC
LIMIT
    10;
```

Result:

Rows returned (10)				Export ▼
<input type="text" value="Search rows"/>				< 1 > ⚙
atm_number ▼	atm_manufacturer ▼	location ▼	total_transaction_amount ▼	
39	NCR	Svenstrup	277097637	
20	NCR	Bispensgade	271008803	
24	NCR	Hobro	268289882	
10	NCR	NÃfÃ_resundby	267379103	
45	NCR	Abildgaard	265639616	
16	NCR	Skive	220677013	
40	Diebold Nixdorf	Frederikshavn	219812287	
41	Diebold Nixdorf	Skagen	214127315	
1	NCR	NÃfÃ'stved	213721117	
48	Diebold Nixdorf	BrÃfÃ_nderslev	212883099	

ETL Project : ATM Transactions - Redshift Analytical Queries

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

Query :

```
WITH card_type_failure AS (
    SELECT
        card_type,
        COUNT(trans_id) AS transactions,
        SUM(CASE WHEN atm_status = 'Inactive' THEN 1 ELSE 0 END) AS
inactive_count
    FROM
        atm_trans.fact_atm_trans F
    JOIN
        atm_trans.dim_card_type C ON F.card_type_id = C.card_type_id
    GROUP BY
        card_type
)
SELECT
    *,
    ROUND(CAST(inactive_count AS numeric(10,2)) / transactions * 100, 2) AS
inactive_count_percent
FROM
    card_type_failure
ORDER BY
    inactive_count_percent DESC;
```

Result:

Rows returned (12)				Export ▼
<input type="text" value="Search rows"/>				< 1 2 > ⚙
card_type ▼	transactions ▼	inactive_count ▼	inactive_count_percent ▼	
Mastercard - on-us	458226	86000	18.77	
VISA	170828	30713	17.98	
Dankort - on-us	143813	24680	17.16	
CIRRUS	17362	2953	17.01	
HÃfÃ\vekort - on-us	62487	10331	16.53	
Dankort	28581	4557	15.94	
MasterCard	400507	63482	15.85	
Visa Dankort - on-us	748805	112972	15.09	
HÃfÃ\vekort	8459	1208	14.28	
Visa Dankort	427840	60547	14.15	

ETL Project : ATM Transactions - Redshift Analytical Queries

7. Top 10 records with the number of transactions ordered by the ATM_number, ATM_manufacturer, location, weekend_flag and then total_transaction_count, on weekdays and on weekends throughout the year

Query :

```
SELECT
    atm_number, atm_manufacturer, location,
    CASE WHEN weekday IN ('Sunday', 'Saturday') THEN 1 ELSE 0 END AS
weekend_flag,
    COUNT(trans_id) AS transactions
FROM
    atm_trans.fact_atm_trans T
INNER JOIN
    atm_trans.dim_atm A ON T.atm_id = A.atm_id
INNER JOIN
    atm_trans.dim_location L ON A.atm_location_id = L.location_id
INNER JOIN
    atm_trans.dim_date D ON T.date_id = D.date_id
GROUPBY
    atm_number,
    atm_manufacturer,
    location,
    weekend_flag
ORDER BY
    atm_number, atm_manufacturer,
    location, weekend_flag,
    transactions
LIMIT 10;
```

Result:

Rows returned (10)					Export ▼
<input type="text" value="Search rows"/>					< 1 > ⚙️
atm_number ▼	atm_manufacturer ▼	location ▼	weekend_flag ▼	transactions ▼	
1	NCR	NÃfÃstved	0	32711	
1	NCR	NÃfÃstved	1	10076	
10	NCR	NÃfÃ, rresundby	0	41667	
10	NCR	NÃfÃ, rresundby	1	12127	
100	NCR	Skive	0	17812	
100	NCR	Skive	1	1	
101	NCR	Bryggen Vejle	0	11693	
101	NCR	Bryggen Vejle	1	3247	
102	NCR	Aalborg Storcenter indg. D	0	14556	
102	NCR	Aalborg Storcenter indg. D	1	3741	

ETL Project : ATM Transactions - Redshift Analytical Queries

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

Query :

```
WITH atm_weekday_details AS (  
    SELECT atm_number, atm_manufacturer, location, weekday, COUNT(trans_id)  
    AS transactions  
    FROM atm_trans.fact_atm_trans T  
    INNER JOIN atm_trans.dim_atm A ON T.atm_id = A.atm_id  
    INNER JOIN atm_trans.dim_location L ON A.atm_location_id = L.location_id  
    INNER JOIN atm_trans.dim_date D ON T.date_id = D.date_id  
    WHERE location = 'Vejgaard'  
    GROUP BY atm_number, atm_manufacturer, location, weekday  
)  
max_weekday AS (  
    SELECT weekday  
    FROM atm_weekday_details  
    WHERE transactions = (SELECT MAX(transactions) FROM  
atm_weekday_details)  
    LIMIT 1  
)  
SELECT *  
FROM atm_weekday_details  
WHERE weekday = (SELECT weekday FROM max_weekday)  
ORDER BY transactions;
```

Result:

Rows returned (2)					Export ▼
<input type="text" value="Search rows"/>					< 1 > ⚙
atm_number ▼	atm_manufacturer ▼	location ▼	weekday ▼	transactions ▼	
103	Diebold Nixdorf	Vejgaard	Friday	4757	
2	NCR	Vejgaard	Friday	6290	