



## 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

```
select
 a.atm number,
 a.atm manufacturer,
 1.location, count(trans id) as total transaction count,
 sum(case when atm status = 'Inactive' then 1 else 0 end ) as
  inactive transaction count, (inactive transaction count /
 total transaction count )* 100 as count percent
   atm data.fact atm trans f, atm data.dim atm a,
   atm data.dim location 1
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
having count percent > 50
order by inactive transaction count desc
limit 10;
```

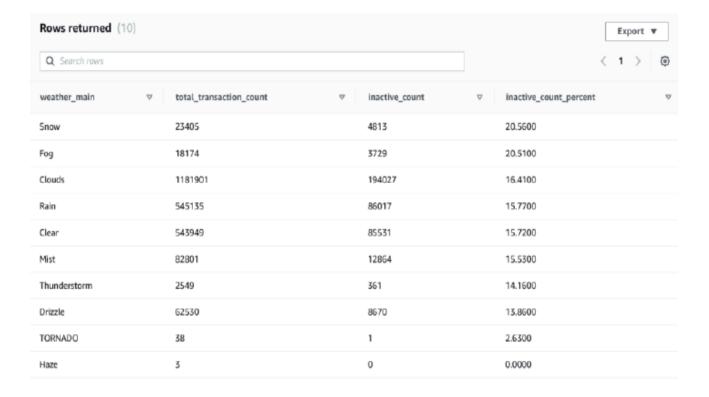
Rows returned	(10)		Export ▼		
Q Search rows			< 1 > @		
atm_numbe r ▽	atm_manufacturer ▽	location $\nabla$	total_transaction_coun  t	$inactive\_transaction\_count \   \triangledown$	count_percent ▽
16	NCR	Skive	44043	44043	100
12	NCR	$\tilde{\mathbb{A}}f$ Ëæster $\tilde{\mathbb{A}}f$ Â¥ Duus	33982	33982	100
2	NCR	Vejgaard	33725	33725	100
88	NCR	Storcenter indg. A	32183	32183	100
30	NCR	Nyk $ ilde{A} f \hat{A}$ , bing Mors	30883	30883	100
52	NCR	Fars $ ilde{A} f \hat{A}$ ,	27361	27361	100
50	NCR	Aarhus	23416	23416	100
29	NCR	Skelagervej 15	20773	20773	100
81	NCR	Spar K $ ilde{A}_f \hat{A}_j$ bmand Tornh $ ilde{A}_f \hat{A}_j$ j	20148	20148	100
102	NCR	Aalborg Storcenter Afd	18297	18297	100





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

```
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
    atm_data.fact_atm_trans f
where
    f.weather_main != ''
group by f.weather_main
order by inactive_count_percent desc
limit 10;
```







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

```
select
   a.atm_number,
   a.atm_manufacturer,
   l.location,
   count(trans_id) as total_transaction_count
from
   atm_data.fact_atm_trans f,
   atm_data.dim_atm a,
   atm_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;
```

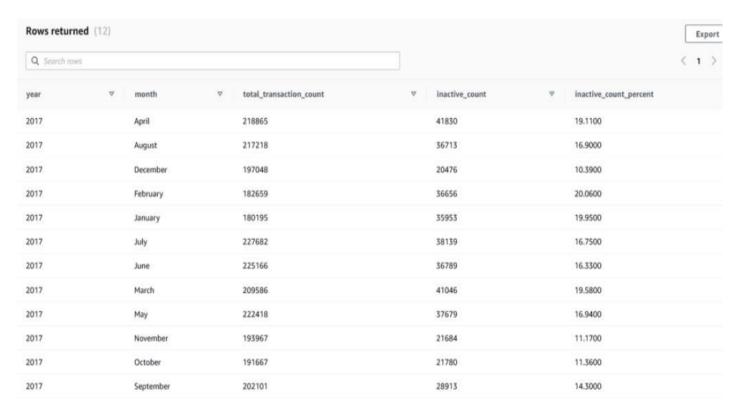
Rows returned (10)  Q Search rows					
atm_number	▽ atm_manufacturer	▽ location	▽	total_transaction_count	▼
39	NCR	Svenstrup		55380	
20	NCR	Bispensgade		54211	
10	NCR	$N ilde{A}f ilde{A}$ , rresundby		53794	
24	NCR	Hobro		53378	
45	NCR	Abildgaard		53198	
16	NCR	Skive		44043	
40	Diebold Nixdorf	Frederikshavn		43767	
1	NCR	N $ar{A}far{A}$ ¦stved		42787	
41	Diebold Nixdorf	Skagen		42732	
48	Diebold Nixdorf	BrÃfÂ, nderslev		42493	





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

```
select
  d.year,
  d.month,
  count(trans_id) as total_transaction_count,
  sum( case when atm_status = 'Inactive' then 1 else 0 end )
ainactive_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
  atm_data.fact_atm_trans f inner join atm_data.dim_date d
  on f.date_id = d.date_id
  group by d.year, d.month
  order by d.year, d.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
   atm_data.fact_atm_trans f,
   atm_data.dim_atm a,
   atm_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;
```

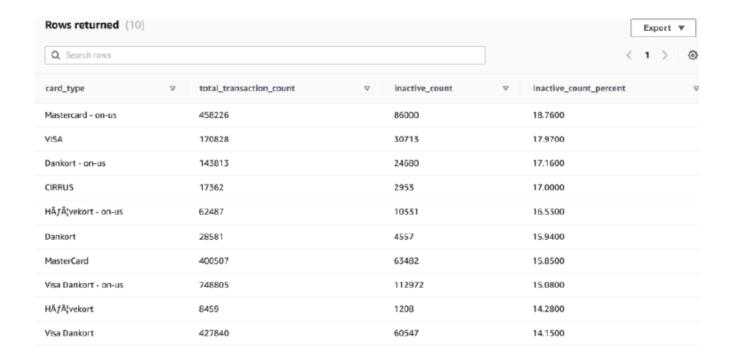
Rows returned (10)	Exp		
Q Search rows	< 1		
atm_number	▽ atm_manufacturer	▽ location	♥ total_transaction_amount
39	NCR	Svenstrup	277097637
20	NCR	Bispensgade	271008803
24	NCR	Hobro	268289882
10	NCR	$N\tilde{A}f\hat{A}$ , rresundby	267379103
45	NCR	Abildgaard	265639616
16	NCR	Skive	220677013
40	Diebold Nixdorf	Frederikshavn	219812287
41	Diebold Nixdorf	Skagen	214127315
1	NCR	NÃ $f$ Â $\sharp$ stved	213721117
48	Diebold Nixdorf	$Br\tilde{A}f\tilde{A}_{_{\parallel}}nderslev$	212883099





#### 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 )
asinactive_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
    atm_data.fact_atm_trans f, atm_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;
```

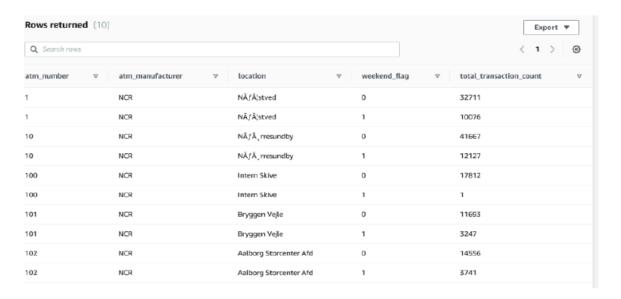






7. Number of transactions happening on an ATM on weekdays and on weekends throughout the year. Order this by the ATM\_number, ATM\_manufacturer, location, weekend\_flag and then total\_transaction\_count

```
select
a.atm number,
a.atm manufacturer,
1.location,
case when d.weekday in ('Saturday', 'Sunday') then 1 else 0 end as
weekend flag,
count(trans id) as total transaction count
atm data.fact atm trans f,
atm data.dim atm a,
atm data.dim location 1,
atm data.dim date d
where
f.atm id = a.atm id
and a.atm location id = 1.location id
and f.date id = d.date id
group by
a.atm number,
a.atm manufacturer,
1.location,
weekend flag
order by
a.atm number,
a.atm manufacturer,
1.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
  atm data.fact atm trans f
    INNER JOIN atm data.dim atm a
      ON f.atm id = a.atm id
    INNER JOIN atm data.dim location 1
      ON a.atm location id = 1.location id
    INNER JOIN atm data.dim date d
      ON f.date id = d.date id
WHERE 1.location = 'Vejgaard' AND d.weekday
    IN (SELECT d.weekday
    FROM atm data.fact atm trans f
    INNER JOIN atm data.dim date d
      ON f.date id = d.date id
    INNER JOIN atm data.dim location 1
      ON f.weather_loc_id = l.location_id
WHERE l.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;
```

