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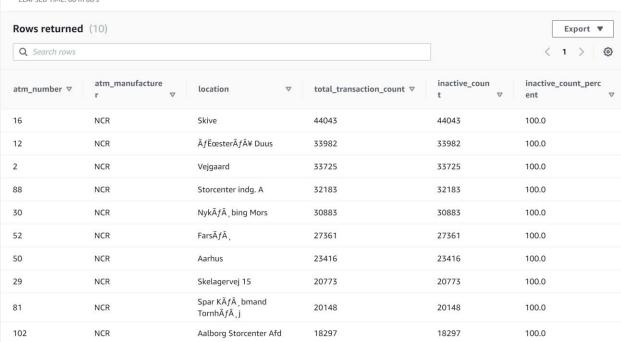
1. **Top 10 ATMs where most transactions are in the 'inactive' state**

select
atm_number,
atm_manufacturer,
location,
total_transaction_count,
inactive_count,
inactive_count_percent
from (
select a.*, b.* from
etl_proj_atm.dim_atm as a
join etl_proj_atm.dim_location as b
on (a.atm_location_id=b.location_id)
) as a
join
(select count(*) as total_transaction_count, count(*) as inactive_count,atm_id,
(inactive_count/total_transaction_count)*100.0 as inactive_count_percent
from etl_proj_atm.fact_atm_trans
group by atm_id , atm_status
having atm_status='Inactive'
) as c
on a.atm_id=c.atm_id
order by c.inactive_count desc
limit 10

Rows returned (10)

Export ▼

🔍 Search rows

‹ 1 › ⚙

atm_number ▽	atm_manufacturer ▽	location ▽	total_transaction_count ▽	inactive_count ▽	inactive_count_percent ▽
16	NCR	Skive	44043	44043	100.0
12	NCR	ÃƒËœsterÃƒÂ¥ Duus	33982	33982	100.0
2	NCR	Vejgaard	33725	33725	100.0
88	NCR	Storcenter indg. A	32183	32183	100.0
30	NCR	NykÃƒÂ¸bing Mors	30883	30883	100.0
52	NCR	FarsÃƒÂ¸	27361	27361	100.0
50	NCR	Aarhus	23416	23416	100.0
29	NCR	Skelagervej 15	20773	20773	100.0
81	NCR	Spar KÃƒÂ¸bmand TornhÃƒÂ¸j	20148	20148	100.0
102	NCR	Aalborg Storcenter Afd	18297	18297	100.0

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

```
select
a.weather_main,total_transaction_count,inactive_count,round(((inactive_count*100.0)/(total_tra
nsaction_count*100.0)*100),4) as inactive_count_percent
from
(
(select weather_main, count(*) as total_transaction_count
from etl_proj_atm.fact_atm_trans
group by weather_main
order by weather_main desc
) as a
join
(select weather_main, count(*) as inactive_count
from etl_proj_atm.fact_atm_trans
where atm_status='Inactive'
group by weather_main
order by weather_main desc
)
as b
on (a.weather_main=b.weather_main)
)
order by inactive_count_percent desc
```

weather_main	total_transaction_count	inactive_count	inactive_count_percent
Snow	23405	4813	20.5640
Fog	18174	3729	20.5183
	8087	1645	20.3413
Clouds	1181901	194027	16.4165
Rain	545135	86017	15.7790
Clear	543949	85531	15.7241
Mist	82801	12864	15.5360
Thunderstorm	2549	361	14.1624
Drizzle	62530	8670	13.8653
TORNADO	38	1	2.6316

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

```
select
atm_number,
atm_manufacturer,
location,
total_transaction_count
from
(
(select a.*, b.* from
etl_proj_atm.dim_atm as a
join etl_proj_atm.dim_location as b
on (a.atm_location_id=b.location_id)
) as a
join
(select count(*) as total_transaction_count, atm_id
from etl_proj_atm.fact_atm_trans
group by atm_id
)
as c
on a.atm_id=c.atm_id
)
order by c.total_transaction_count desc
limit 10
```

Completed, started on January 17, 2021 at 21:52:00
ELAPSED TIME: 00 m 14 s

Rows returned (10)

Search rows

atm_number	atm_manufacturer	location	total_transaction_count
39	NCR	Svenstrup	55380
20	NCR	Bispensgade	54211
10	NCR	NÃƒÂ¸rresundby	53794
24	NCR	Hobro	53378
45	NCR	Abildgaard	53198
16	NCR	Skive	44043
40	Diebold Nixdorf	Frederikshavn	43767
1	NCR	NÃƒÂ¦stved	42787
41	Diebold Nixdorf	Skagen	42732
48	Diebold Nixdorf	BrÃƒÂ¸nderslev	42493

4. Number of overall ATM transactions going inactive per month for each month
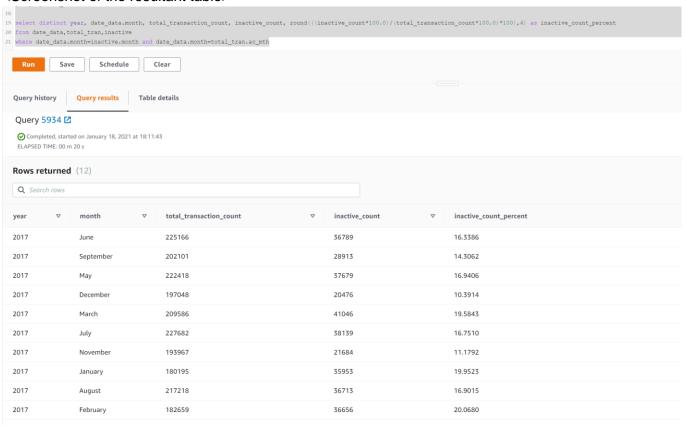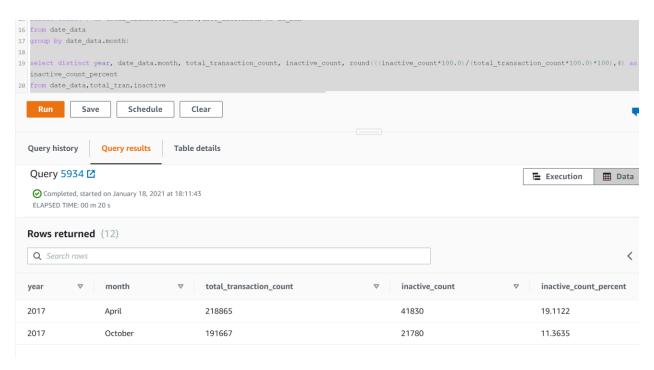
\<Query\>

```
with date_data as (select year, a.month ,b.*
from etl_proj_atm .dim_date a ,etl_proj_atm.fact_atm_trans b
where a.date_id=b.date_id),

inactive as (
select date_data.month, count(*) as inactive_count,date_data.month as in_mth
from date_data --etl_proj_atm.fact_atm_trans
where atm_status='Inactive'
group by date_data.month),

total_tran as (
select count(*) as total_transaction_count,date_data.month as ac_mth
from date_data
group by date_data.month)

select distinct year, date_data.month, total_transaction_count, inactive_count,
round(((inactive_count*100.0)/(total_transaction_count*100.0)*100),4) as
inactive_count_percent
from date_data,total_tran,inactive
where date_data.month=inactive.month and date_data.month=total_tran.ac_mth
```

<Screenshot of the resultant table>

```
18
19 select distinct year, date_data.month, total_transaction_count, inactive_count, round(((inactive_count*100.0)/(total_transaction_count*100.0)*100),4) as inactive_count_percent
20 from date_data,total_tran,inactive
21 where date_data.month=inactive.month and date_data.month=total_tran.ac_mth
```

Run   Save   Schedule   Clear

Query history   **Query results**   Table details

Query 5934 ↗

⊘ Completed, started on January 18, 2021 at 18:11:43
ELAPSED TIME: 00 m 20 s

Rows returned (12)

🔍 Search rows

year	month	total_transaction_count	inactive_count	inactive_count_percent
2017	June	225166	36789	16.3386
2017	September	202101	28913	14.3062
2017	May	222418	37679	16.9406
2017	December	197048	20476	10.3914
2017	March	209586	41046	19.5843
2017	July	227682	38139	16.7510
2017	November	193967	21684	11.1792
2017	January	180195	35953	19.9523
2017	August	217218	36713	16.9015
2017	February	182659	36656	20.0680

```
16 from date_data
17 group by date_data.month)
18
19 select distinct year, date_data.month, total_transaction_count, inactive_count, round(((inactive_count*100.0)/(total_transaction_count*100.0)*100),4) as inactive_count_percent
20 from date_data,total_tran,inactive
```

Run   Save   Schedule   Clear

Query history   **Query results**   Table details

Query 5934 ↗   ≣ Execution   ⊞ Data

⊘ Completed, started on January 18, 2021 at 18:11:43
ELAPSED TIME: 00 m 20 s

Rows returned (12)

🔍 Search rows   ‹

year	month	total_transaction_count	inactive_count	inactive_count_percent
2017	April	218865	41830	19.1122
2017	October	191667	21780	11.3635

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

```
select
atm_number,
atm_manufacturer,
location,
total_transaction_amount
from
(
(select a.*, b.* from
etl_proj_atm.dim_atm as a
join etl_proj_atm.dim_location as b
on (a.atm_location_id=b.location_id)
) as a
join
(select sum(transaction_amount) as total_transaction_amount ,atm_id
from etl_proj_atm.fact_atm_trans
group by atm_id
)
as c
on a.atm_id=c.atm_id
)
order by c.total_transaction_amount desc
limit 10
```

Rows returned (10)

🔍 Search rows

atm_number ▽	atm_manufacturer ▽	location ▽	total_transaction_amount
39	NCR	Svenstrup	277097637
20	NCR	Bispensgade	271008803
24	NCR	Hobro	268289882
10	NCR	NÃƒÂ¸rresundby	267379103
45	NCR	Abildgaard	265639616
16	NCR	Skive	220677013
40	Diebold Nixdorf	Frederikshavn	219812287
41	Diebold Nixdorf	Skagen	214127315
1	NCR	NÃƒÂ¦stved	213721117
48	Diebold Nixdorf	BrÃƒÂ¸nderslev	212883099

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

```
select
card_type,total_transaction_count,inactive_count,round(((inactive_count*100.0)/(total_transactio
n_count*100.0)*100),4) as inactive_count_percentinactive_count_percent
from
(
(select count(*) as total_transaction_count,a.card_type_id,card_type
from etl_proj_atm.fact_atm_trans as a , etl_proj_atm.dim_card_type as b
where a.card_type_id=b.card_type_id
group by a.card_type_id,b.card_type
) as c
join
(select count(*) as inactive_count,card_type_id,atm_status
from etl_proj_atm.fact_atm_trans
where atm_status='Inactive'
group by card_type_id,atm_status)
 as d
on c.card_type_id=d.card_type_id
)
order by inactive_count_percentinactive_count_percent desc
```

Rows returned (12)

🔍 Search rows

card_type ▽	total_transaction_count ▽	inactive_count ▽	inactive_count_percentinactive_count_percent
Mastercard - on-us	458226	86000	18.7680
VISA	170828	30713	17.9789
Dankort - on-us	143813	24680	17.1612
CIRRUS	17362	2953	17.0084
HÃƒÂ¦vekort - on-us	62487	10331	16.5330
Dankort	28581	4557	15.9442
MasterCard	400507	63482	15.8504
Visa Dankort - on-us	748805	112972	15.0870
HÃƒÂ¦vekort	8459	1208	14.2806
Visa Dankort	427840	60547	14.1518

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>
with atm_loc as
(select atm_id, atm_number,atm_manufacturer,location,location_id from
etl_proj_atm.dim_atm as a , etl_proj_atm.dim_location as b
where (a.atm_location_id=b.location_id)
),
dt_fact as
(select weekday,case when  weekday in (
'Tuesday',
'Monday',
'Thursday',
'Friday',
'Wednesday') then '0'
 else '1'
 end as weekend_flag ,
 b.*
from etl_proj_atm.fact_atm_trans as b,etl_proj_atm.dim_date as a
where a.date_id=b.date_id
),
count_df as
(select count(*) as total_transaction_count,dt_fact.atm_id,weekday
from dt_fact,atm_loc
where atm_loc.atm_id=dt_fact.atm_id
group by dt_fact.atm_id,weekday)

select
atm_loc.atm_number,atm_loc.atm_manufacturer,location,weekend_flag,total_transaction_count
from atm_loc,dt_fact, count_df
where dt_fact.atm_id=count_df.atm_id --atm_loc.location_id=dt_fact.weather_loc_id --
atm_loc.atm_id=dt_fact.atm_id
and dt_fact.atm_id=count_df.atm_id
order by atm_number,atm_manufacturer,location,weekend_flag,total_transaction_count
```

<Screenshot of the resultant table>

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

<Query>

```
with atm_loc as
(select atm_id, atm_number,atm_manufacturer,location,location_id from
etl_proj_atm.dim_atm as a , etl_proj_atm.dim_location as b
where (a.atm_location_id=b.location_id)
and b.location='Vejgaard'),

dt_fact as
(select weekday,b.*
from etl_proj_atm.dim_date as a,etl_proj_atm.fact_atm_trans as b
where a.date_id=b.date_id),

count_df as
(select count(*) as total_transaction_count,atm_id,weekday
from dt_fact
where weather_loc_id in (select location_id from atm_loc)
group by atm_id,weekday
order by total_transaction_count desc),

final as (select atm_number,atm_manufacturer,location,weekday,total_transaction_count,
row_number()OVER (PARTITION BY atm_number ORDER BY total_transaction_count desc)
AS row_number
from atm_loc,count_df
where atm_loc.atm_id=count_df.atm_id)
select  atm_number,atm_manufacturer,location,weekday,total_transaction_count from final
where row_number=1
```

<Screenshot of the resultant table>

```
487  count_df as
488  (select count(*) as total_transaction_count,atm_id,weekday
489  from dt_fact
```

Run Save Schedule Clear   💬 Sen

Query history   **Query results**   Table details

Query 4844 ↗   ▤ Execution ▦ Data ◹

✓ Completed, started on January 18, 2021 at 17:00:41
ELAPSED TIME: 00 m 08 s

Rows returned (2)

🔍 Search rows   ‹

atm_number ▽	atm_manufacturer ▽	location ▽	weekday ▽	total_transaction_count
2	NCR	Vejgaard	Friday	6290
103	Diebold Nixdorf	Vejgaard	Friday	4757