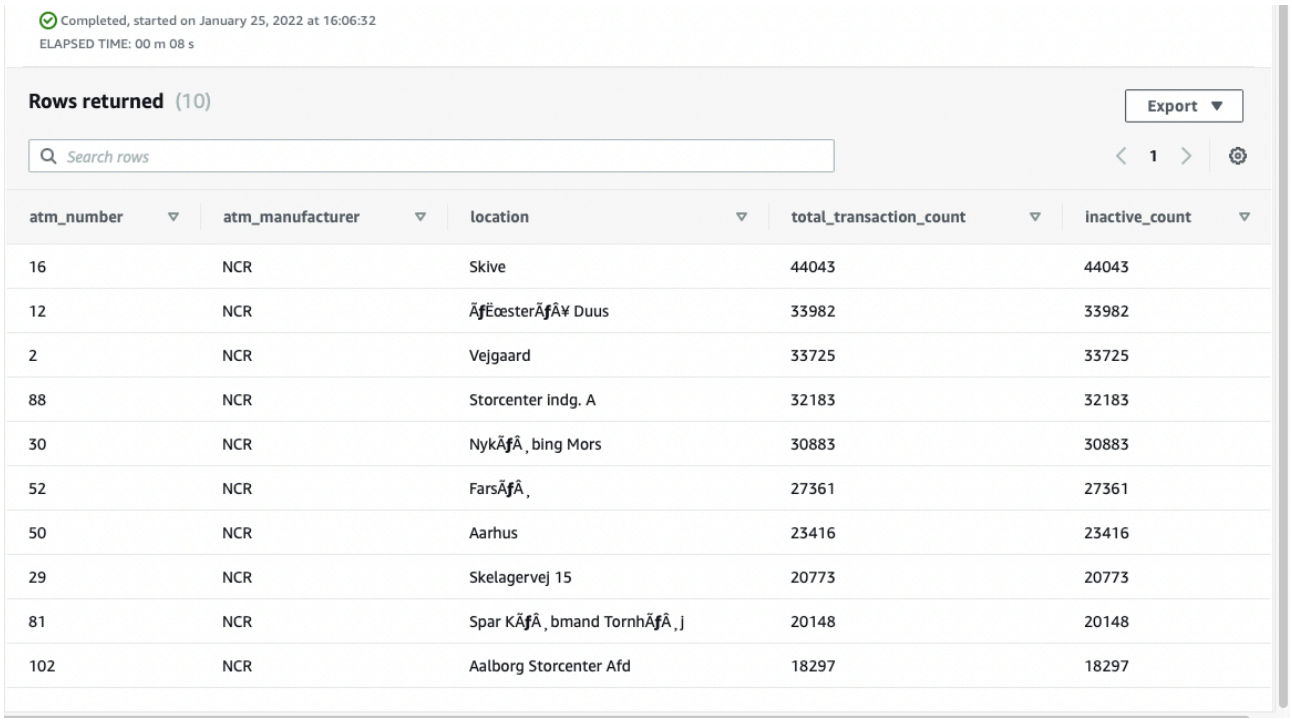
Solving analytical queries on Redshift Cluster

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, l.location, count(t.atm_id) as total_transaction_count, count(t.atm_id) as inactive_count from etl_schema.txn_fact t join etl_schema.atm_dimen a on t.atm_id=a.atm_id join etl_schema.loc_dimen l on a.atm_location_id=l.location_id where t.atm_status='Inactive'

group by a.atm_number , a.atm_manufacturer , l.location order by total_transaction_count desc limit 10;


Completed, started on January 25, 2022 at 16:06:32
ELAPSED TIME: 00 m 08 s

Rows returned (10)

Export

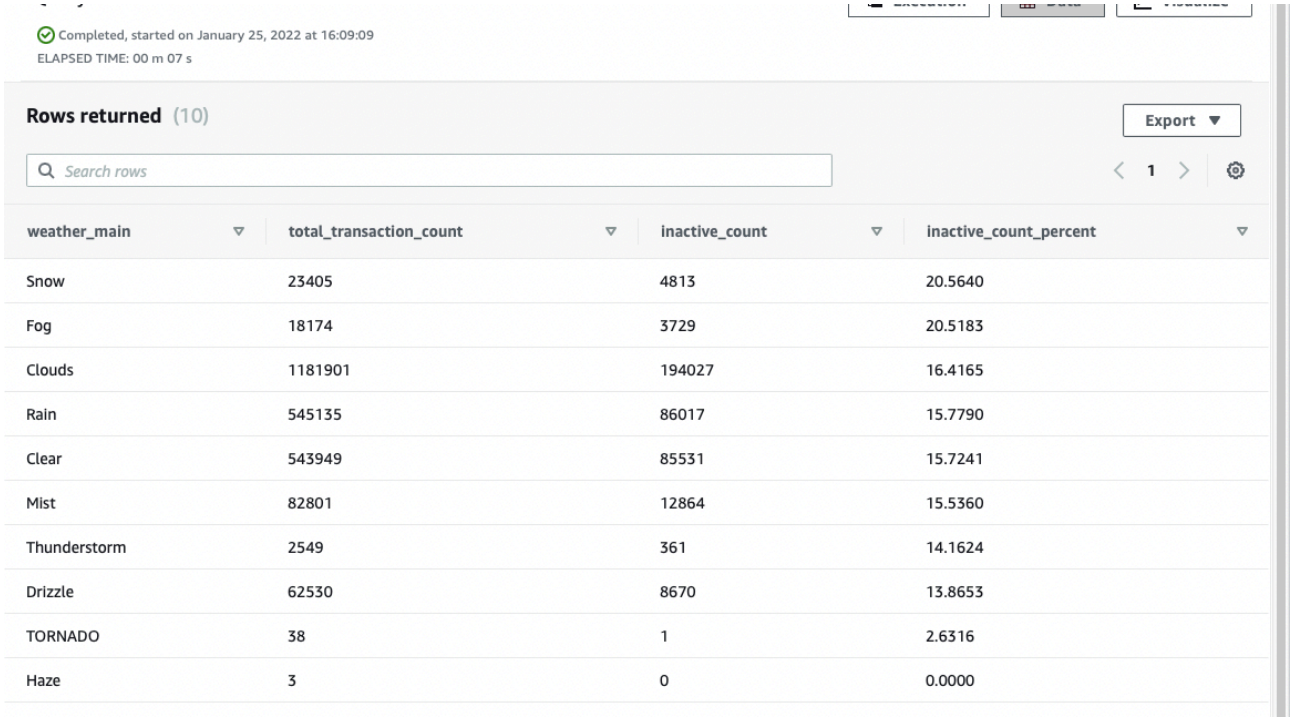
Search rows

1

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

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

select weather_main, count(atm_id) as total_transaction_count, SUM(CASE WHEN atm_status='Inactive' THEN 1 ELSE 0 END) as inactive_count, round(inactive_count*100.00/ total_transaction_count,4) as inactive_count_percent from etl_schema.txn_fact where len(weather_main)>0 group by weather_main order by inactive_count_percent desc limit 10;



Completed, started on January 25, 2022 at 16:09:09
ELAPSED TIME: 00 m 07 s

Rows returned (10)

Export

Search rows

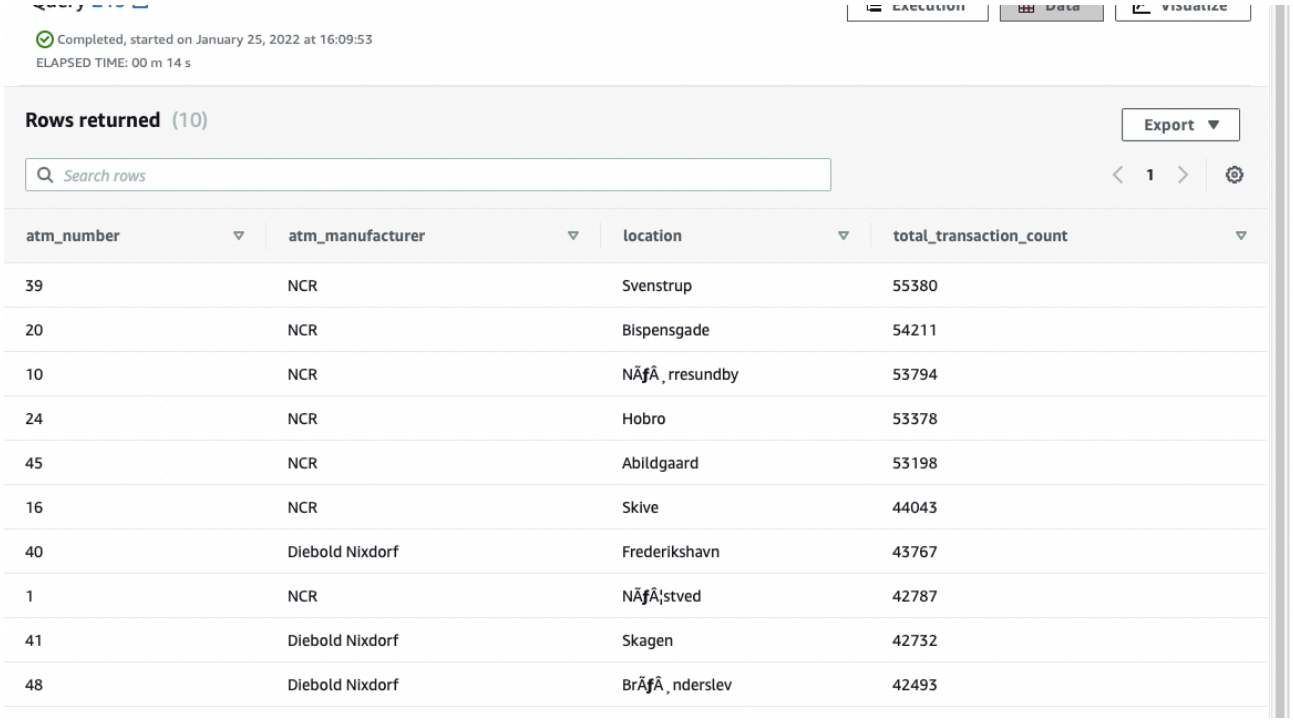
1

weather_main	total_transaction_count	inactive_count	inactive_count_percent
Snow	23405	4813	20.5640
Fog	18174	3729	20.5183
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
Haze	3	0	0.0000

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

select a.atm_number, a.atm_manufacturer, l.location, count(t.atm_id) as total_transaction_count
from etl_schema.txn_fact t join etl_schema.atm_dimen a on t.atm_id=a.atm_id join etl_schema.loc_dimen l on 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;



Completed, started on January 25, 2022 at 16:09:53
ELAPSED TIME: 00 m 14 s

Rows returned (10)

Export

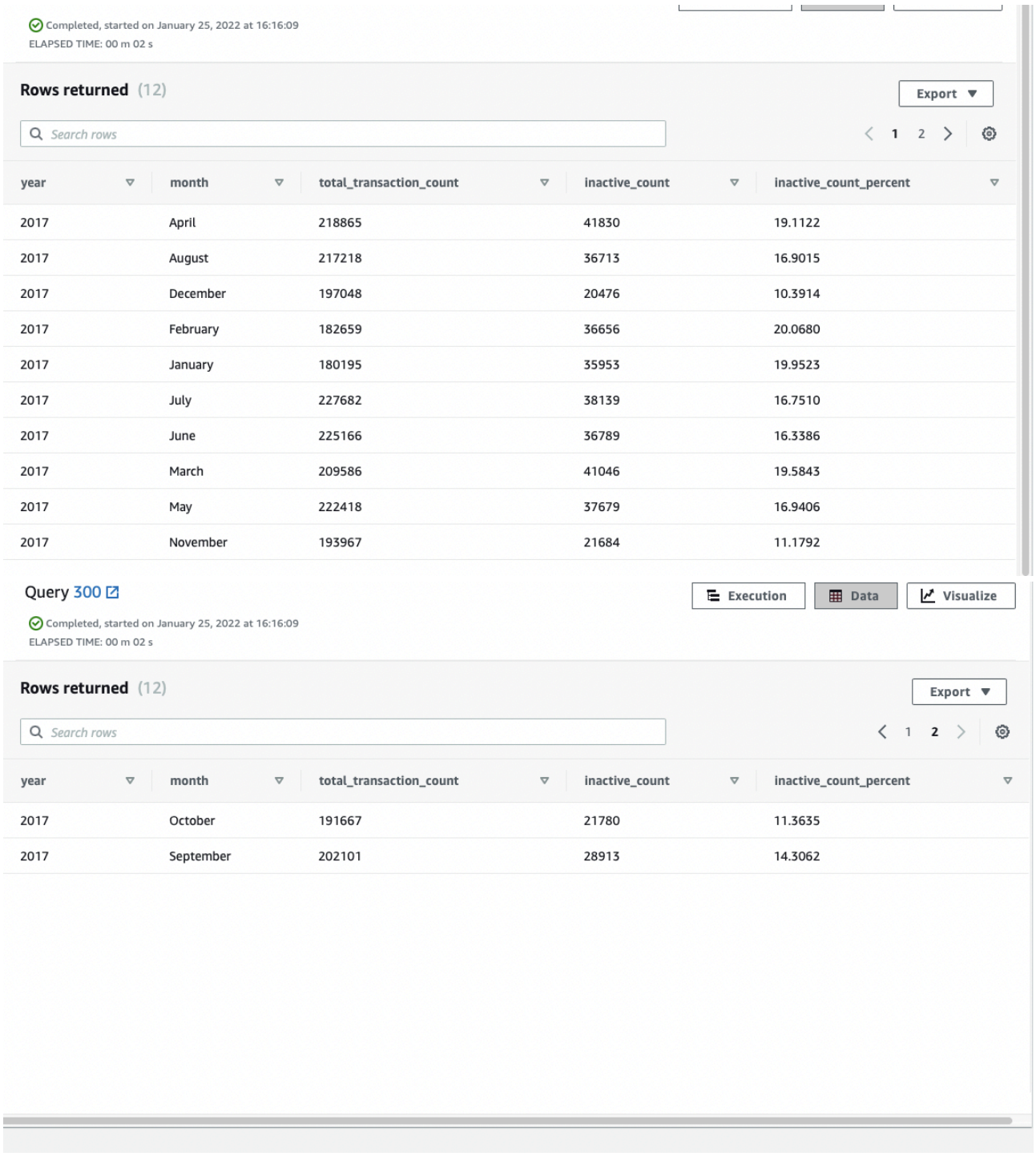
Search rows

1

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

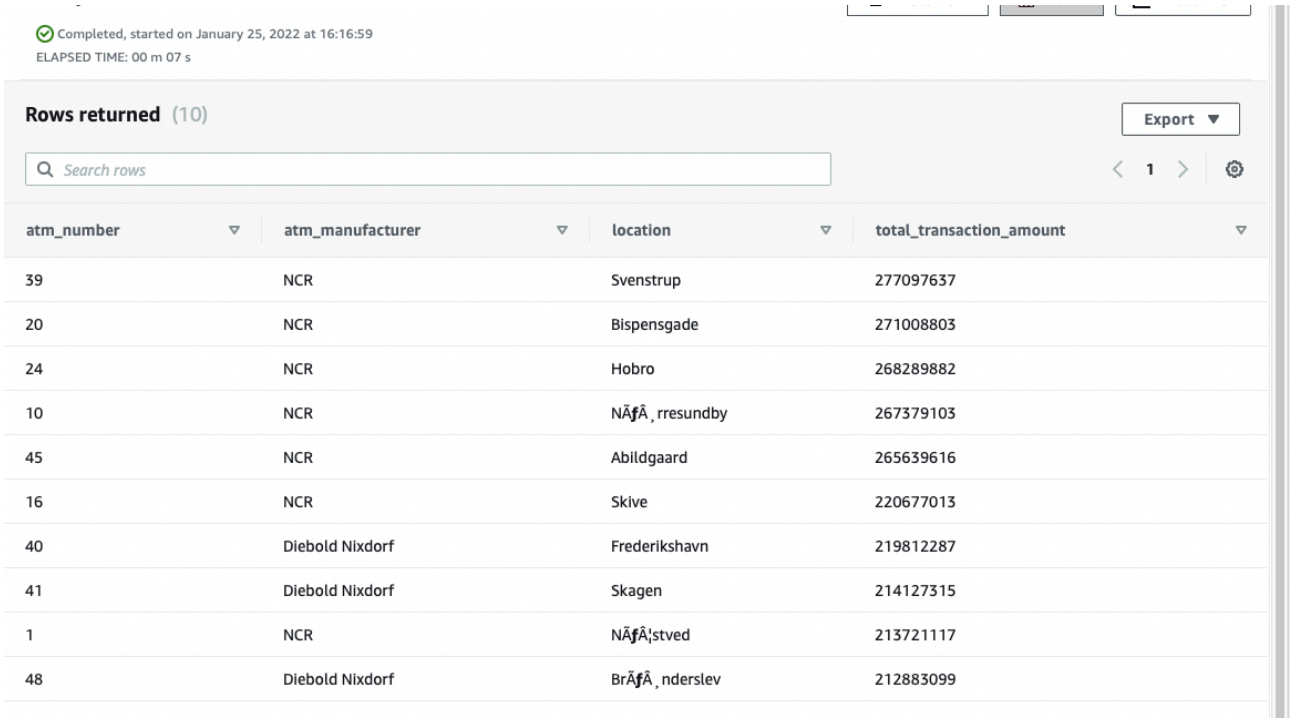
select d.year, d.month, count(t.trans_id) as total_transaction_count, SUM(CASE WHEN atm_status='Inactive' THEN 1 ELSE 0 END) as inactive_count, round(inactive_count*100.00/ total_transaction_count,4) as inactive_count_percent from etl_schema.txn_fact t join etl_schema.date_dimen d on t.date_id=d.date_id group by d.month,d.year
order by d.month limit 12;



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

select a.atm_number, a.atm_manufacturer, l.location, sum(t.transaction_amount) as total_transaction_amount from etl_schema.txn_fact t join etl_schema.atm_dimen a on t.atm_id=a.atm_id join etl_schema.loc_dimen l on a.atm_location_id=l.location_id

where t.currency is not null group by a.atm_number , a.atm_manufacturer , l.location order by total_transaction_amount desc limit 10;



Completed, started on January 25, 2022 at 16:16:59
ELAPSED TIME: 00 m 07 s

Rows returned (10)

Export

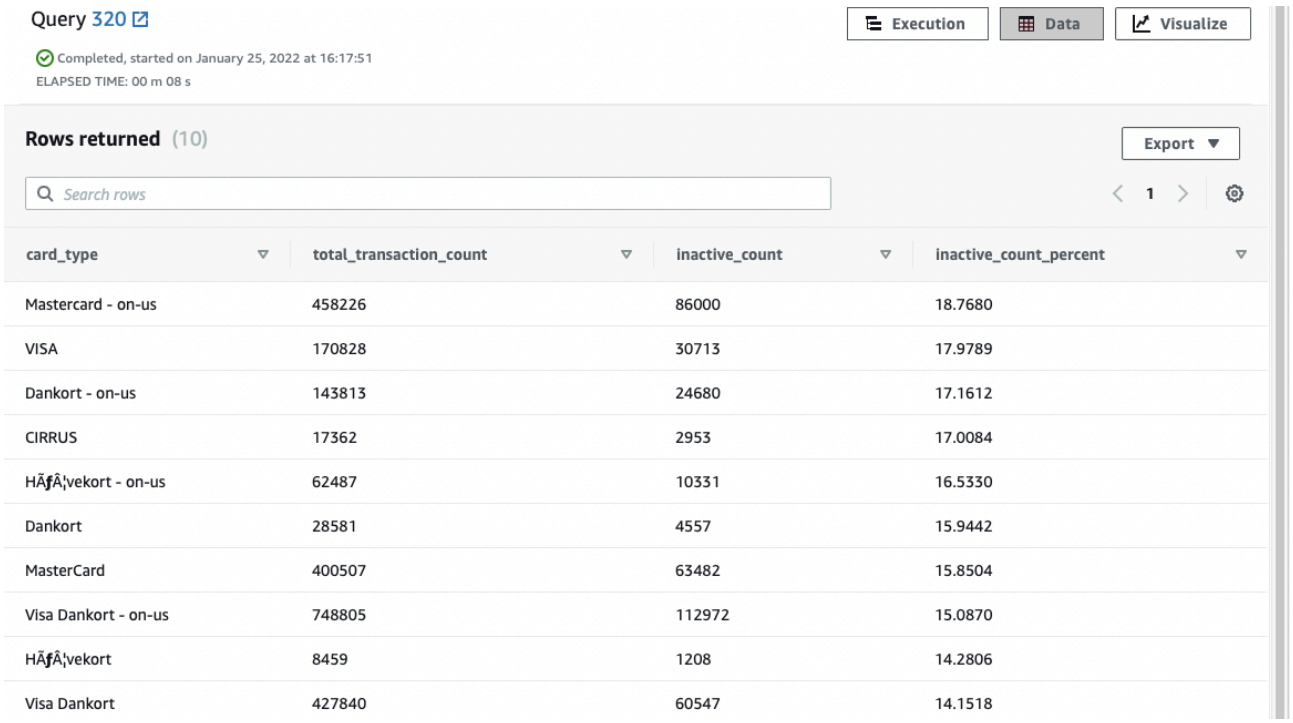
Search rows

1

atm_number	atm_manufacturer	location	total_transaction_amount
39	NCR	Svenstrup	277097637
20	NCR	Bispensgade	271008803
24	NCR	Hobro	268289882
10	NCR	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 c.card_type, count(t.trans_id) as total_transaction_count, SUM(CASE WHEN atm_status='Inactive' THEN 1 ELSE 0 END) as inactive_count, round(inactive_count*100.00/ total_transaction_count,4) as inactive_count_percent from etl_schema.txn_fact t join etl_schema.card_type_dimen c on t.card_type_id=c.card_type_id
group by c.card_type
order by inactive_count_percent desc limit 10;



Query 320

Completed, started on January 25, 2022 at 16:17:51
ELAPSED TIME: 00 m 08 s

Execution | Data | Visualize

Rows returned (10)

Export

Search rows

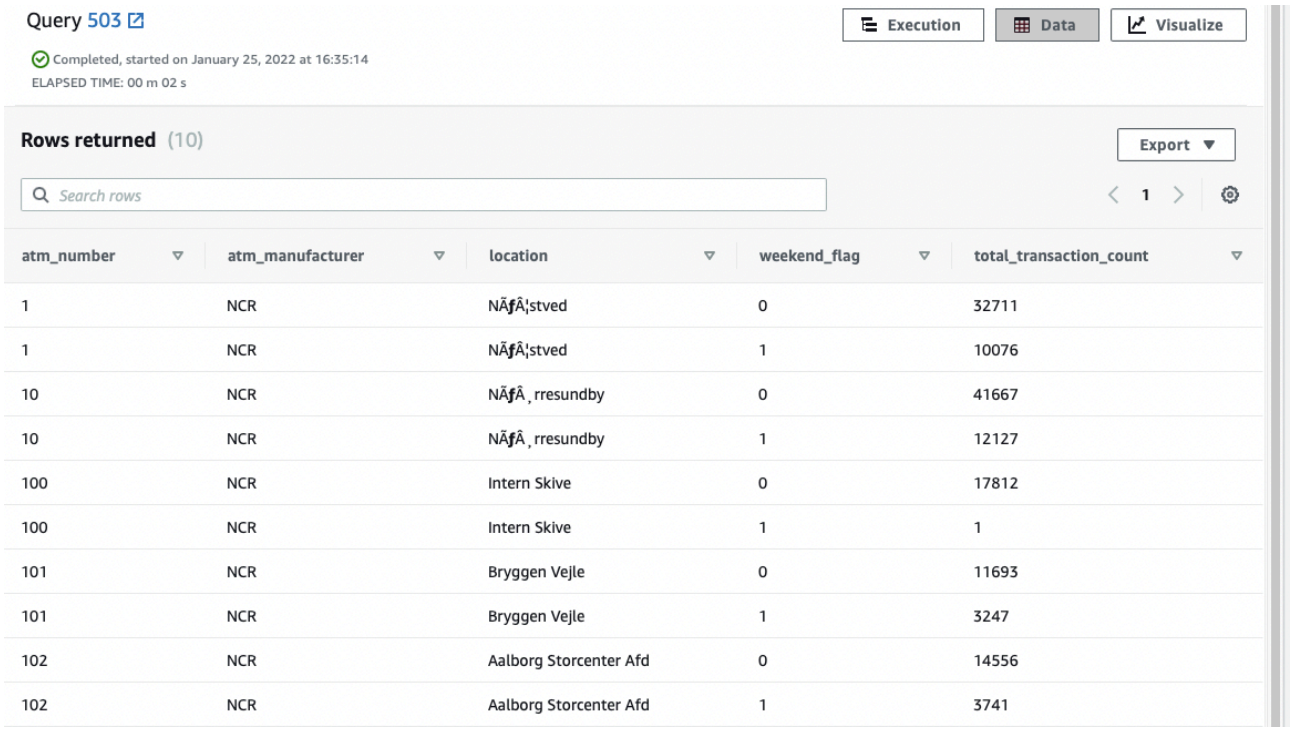
1

card_type	total_transaction_count	inactive_count	inactive_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

```
select a.atm_number, a.atm_manufacturer, l.location, (CASE WHEN
d.weekday in ('Saturday','Sunday') THEN 1 ELSE 0 END) as
weekend_flag,
count(t.atm_id) as total_transaction_count from
etl_schema.txn_fact t join etl_schema.atm_dimen a on
t.atm_id=a.atm_id join

etl_schema.loc_dimen l on a.atm_location_id=l.location_id join
etl_schema.date_dimen d on d.date_id=t.date_id
group by a.atm_number , a.atm_manufacturer , l.location,
weekend_flag
order by a.atm_number, a.atm_manufacturer, l.location,
weekend_flag, total_transaction_count asc limit 10;
```



Query 503

Execution | Data | Visualize

Completed, started on January 25, 2022 at 16:35:14
ELAPSED TIME: 00 m 02 s

Rows returned (10)

Export

Search rows

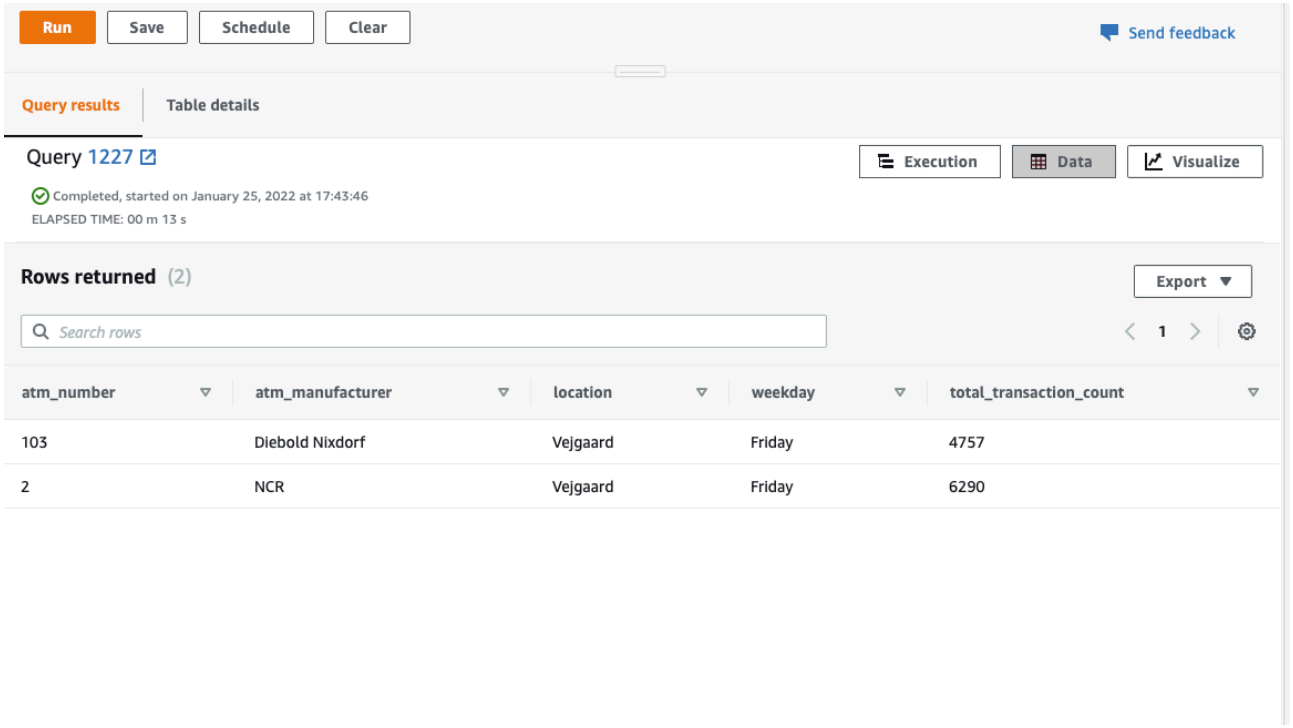
1

atm_number	atm_manufacturer	location	weekend_flag	total_transaction_count
1	NCR	NÃƒÂ¦stved	0	32711
1	NCR	NÃƒÂ¦stved	1	10076
10	NCR	NÃƒÂ¸rresundby	0	41667
10	NCR	NÃƒÂ¸rresundby	1	12127
100	NCR	Intern Skive	0	17812
100	NCR	Intern Skive	1	1
101	NCR	Bryggen Vejle	0	11693
101	NCR	Bryggen Vejle	1	3247
102	NCR	Aalborg Storcenter Afd	0	14556
102	NCR	Aalborg Storcenter Afd	1	3741

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

```
select a.atm_number, a.atm_manufacturer, l.location, d.weekday,
count(t.atm_id) as total_transaction_count
from etl_schema.txn_fact t join etl_schema.atm_dimen a on
t.atm_id=a.atm_id join etl_schema.loc_dimen l on
a.atm_location_id=l.location_id join etl_schema.date_dimen d on
d.date_id=t.date_id

where l.location='Vejgaard' and d.weekday = (select d.weekday
from etl_schema.txn_fact t join etl_schema.date_dimen d on
d.date_id=t.date_id join etl_schema.loc_dimen l on
t.weather_loc_id=l.location_id where l.location='Vejgaard' group by
d.weekday order by count(t.atm_id) desc limit 1) group by
a.atm_number, a.atm_manufacturer , l.location, d.weekday
order by a.atm_number, a.atm_manufacturer, l.location, d.weekday,
total_transaction_count desc limit 2;
```


Run
Save
Schedule
Clear
Send feedback
Query results
Table details
Query 1227
Execution
Data
Visualize
Completed, started on January 25, 2022 at 17:43:46
ELAPSED TIME: 00 m 13 s
Rows returned (2)
Export
Search rows
1
atm_number
atm_manufacturer
location
weekday
total_transaction_count
103
Diebold Nixdorf
Vejgaard
Friday
4757
2
NCR
Vejgaard
Friday
6290