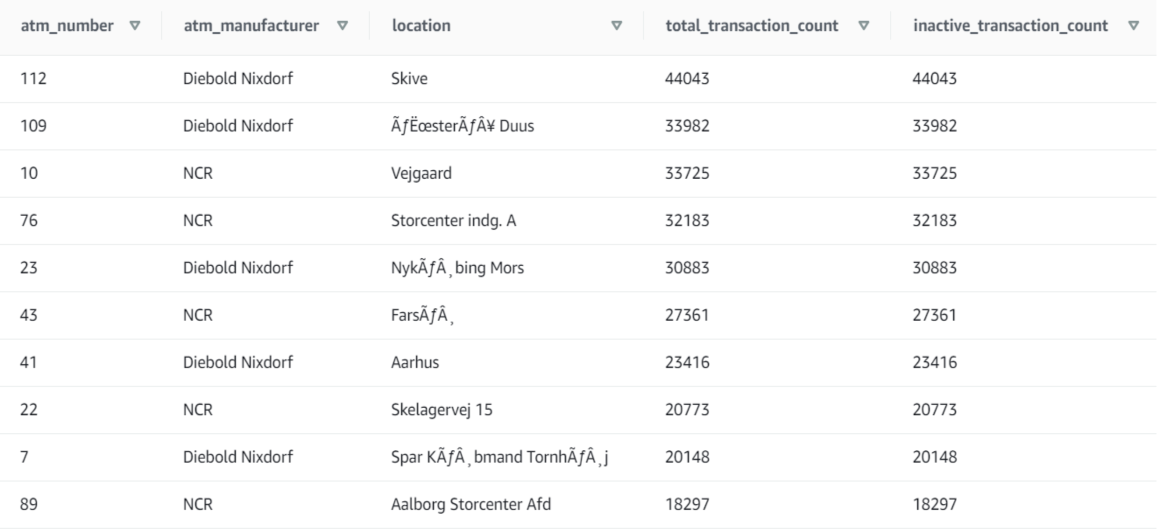
Solving analytical queries on Redshift Cluster

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   b.atm_number, b.atm_manufacturer, c.location , count(1) as Total_transaction_count,
count(1) as inactive_transaction_count
 From DE_ASSIGN.FACT_ATM_TRANS a left join DE_ASSIGN.DIM_ATM b
 on  a.ATM_ID = b.ATM_ID
 left join DE_ASSIGN.DIM_LOCATION c
 on  a.weather_loc_id = c.location_id
 where a.atm_status ='Inactive'
 group by  b.atm_number, b.atm_manufacturer,c.location
 order by 4 desc limit 10;
```

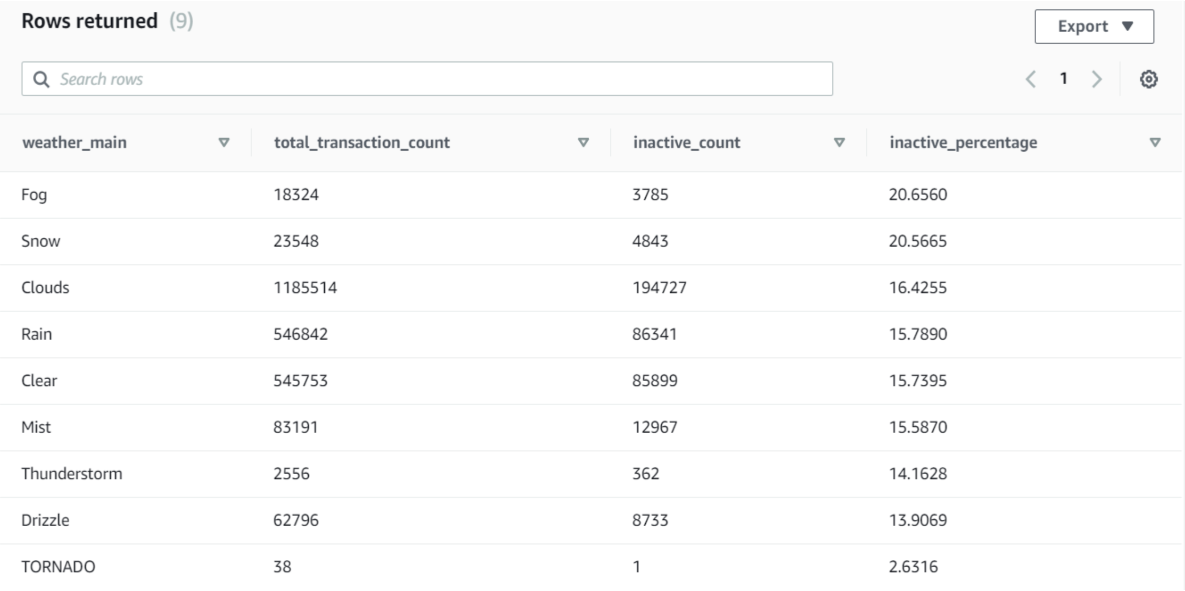
Query Result:



atm_number ▽	atm_manufacturer ▽	location ▽	total_transaction_count ▽	inactive_transaction_count ▽
112	Diebold Nixdorf	Skive	44043	44043
109	Diebold Nixdorf	ÃƒËœsterÃƒÂ¥ Duus	33982	33982
10	NCR	Vejgaard	33725	33725
76	NCR	Storcenter indg. A	32183	32183
23	Diebold Nixdorf	NykÃƒÂ¸bing Mors	30883	30883
43	NCR	FarsÃƒÂ¸	27361	27361
41	Diebold Nixdorf	Aarhus	23416	23416
22	NCR	Skelagervej 15	20773	20773
7	Diebold Nixdorf	Spar KÃƒÂ¸bmand TornhÃƒÂ¸j	20148	20148
89	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 a. weather_main, b.total_count as total_transaction_count, a.inactive_count,
ROUND(a.inactive_count * 100.00 / b.total_count, 4) as inactive_percentage from
  (select count(1) as inactive_count,weather_main  from DE_ASSIGN.FACT_ATM_TRANS
where atm_status ='Inactive'
  group by weather_main) a inner join
 (select count(1) as total_count,weather_main  from DE_ASSIGN.FACT_ATM_TRANS
  group by weather_main)  b
  on  a.weather_main  = b. weather_main
order by inactive_percentage desc ;
```

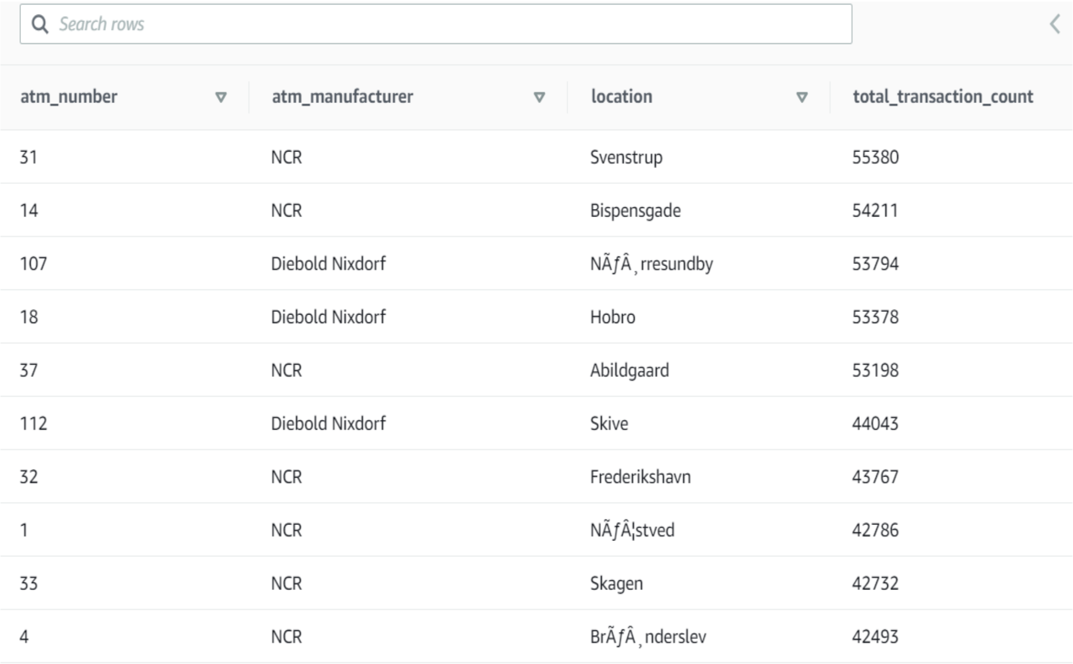


Rows returned (9)

weather_main	total_transaction_count	inactive_count	inactive_percentage
Fog	18324	3785	20.6560
Snow	23548	4843	20.5665
Clouds	1185514	194727	16.4255
Rain	546842	86341	15.7890
Clear	545753	85899	15.7395
Mist	83191	12967	15.5870
Thunderstorm	2556	362	14.1628
Drizzle	62796	8733	13.9069
TORNADO	38	1	2.6316

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

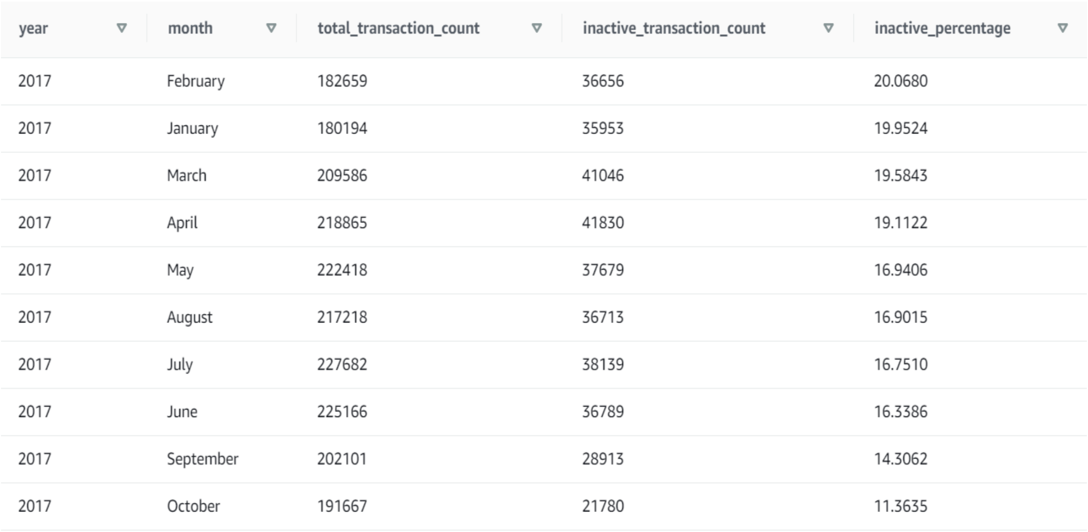
```
 select   b.atm_number, b.atm_manufacturer, c.location , count(1) as Total_transaction_count
From DE_ASSIGN.FACT_ATM_TRANS a left join DE_ASSIGN.DIM_ATM b
on  a.ATM_ID = b.ATM_ID
left join DE_ASSIGN.DIM_LOCATION c
on  a.weather_loc_id = c.location_id
group by  b.atm_number, b.atm_manufacturer,c.location
order by 4 desc limit 10;
```



atm_number	atm_manufacturer	location	total_transaction_count
31	NCR	Svenstrup	55380
14	NCR	Bispensgade	54211
107	Diebold Nixdorf	NÃƒÂ¸rresundby	53794
18	Diebold Nixdorf	Hobro	53378
37	NCR	Abildgaard	53198
112	Diebold Nixdorf	Skive	44043
32	NCR	Frederikshavn	43767
1	NCR	NÃƒÂ¦stved	42786
33	NCR	Skagen	42732
4	NCR	BrÃƒÂ¸nderslev	42493

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

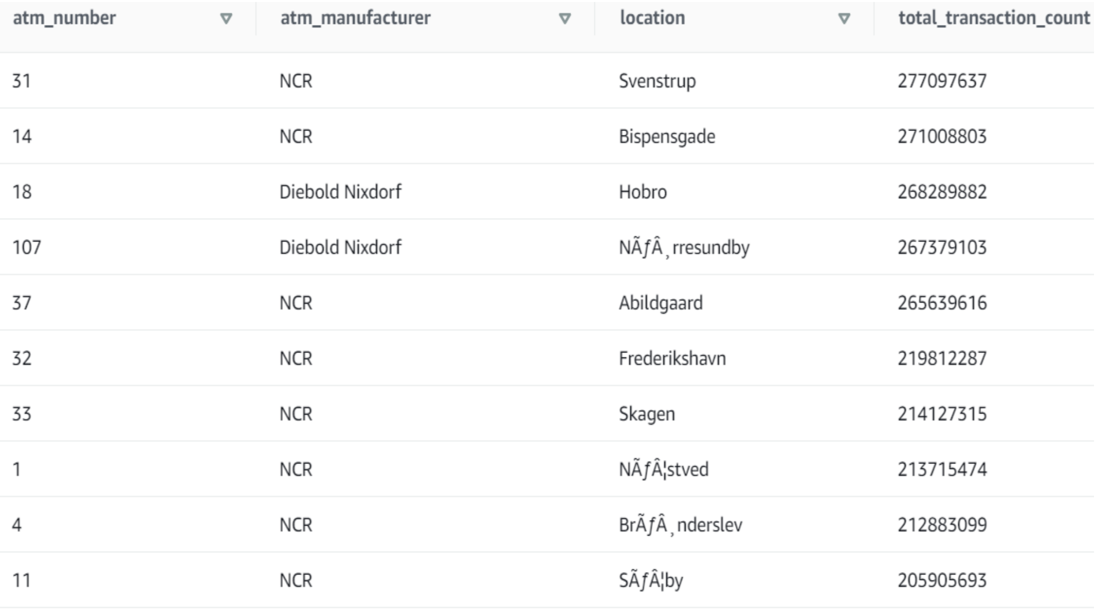
```
select
   a.year, b.month , a. Total_transaction_count , b. Inactive_transaction_count, ROUND(b.Inactive_transaction_count *
100.00 / a.Total_transaction_count, 4
   ) as Inactive_Percentage
from
   (select
      b.year,b.month ,count(1
      ) as Total_transaction_count
   From
      DE_ASSIGN.FACT_ATM_TRANS a left
   join
      DE_ASSIGN.DIM_DATE B on a.DATE_ID = b.DATE_ID
   group
      by b.year,b.month
   ) a inner
join
   ( select
      b.year,b.month, count(1
      ) as Inactive_transaction_count
   From
      DE_ASSIGN.FACT_ATM_TRANS a left
   join
      DE_ASSIGN.DIM_DATE b on a.DATE_ID = b.DATE_ID
   where
      a.atm_status='Inactive'
   group
      by b.year,b.month
   ) bon ( a. Year = b.year
      and a.month = b.month
   ) order by Inactive_Percentage desc ;
```



year	month	total_transaction_count	inactive_transaction_count	inactive_percentage
2017	February	182659	36656	20.0680
2017	January	180194	35953	19.9524
2017	March	209586	41046	19.5843
2017	April	218865	41830	19.1122
2017	May	222418	37679	16.9406
2017	August	217218	36713	16.9015
2017	July	227682	38139	16.7510
2017	June	225166	36789	16.3386
2017	September	202101	28913	14.3062
2017	October	191667	21780	11.3635

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

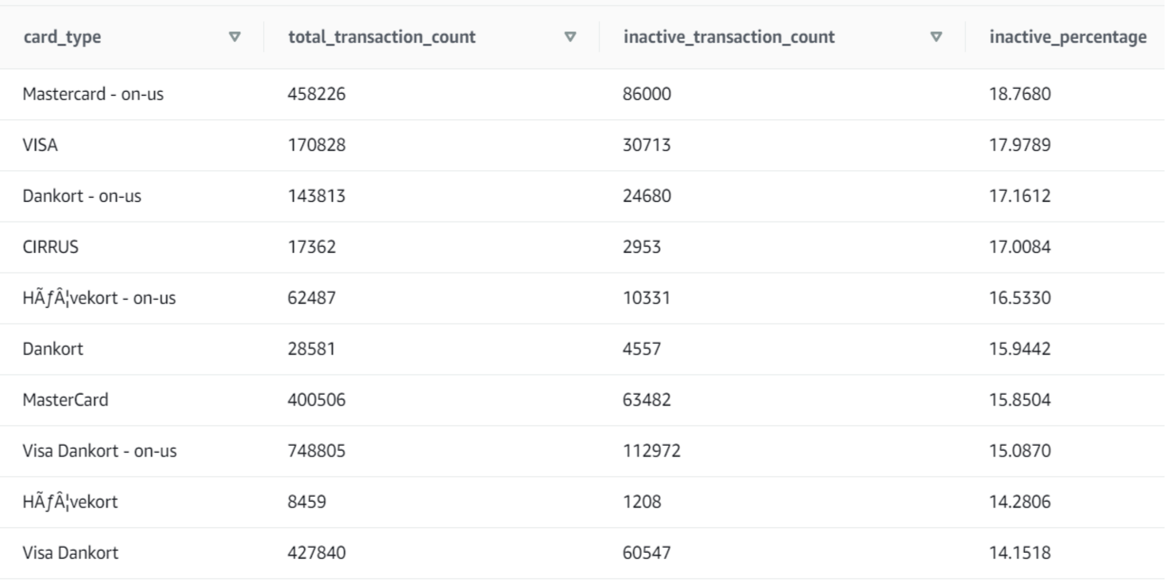
```
select   b.atm_number, b.atm_manufacturer, c.location , sum(transaction_amount) as
Total_transaction_count
 From DE_ASSIGN.FACT_ATM_TRANS a left join DE_ASSIGN.DIM_ATM b
 on  a.ATM_ID = b.ATM_ID
 left join DE_ASSIGN.DIM_LOCATION c
 on  a.weather_loc_id = c.location_id
 where service='Withdrawal' and atm_status= 'Active'
 group by  b.atm_number, b.atm_manufacturer,c.location
 order by 4 desc limit 10;
```



atm_number ▽	atm_manufacturer ▽	location ▽	total_transaction_count
31	NCR	Svenstrup	277097637
14	NCR	Bispensgade	271008803
18	Diebold Nixdorf	Hobro	268289882
107	Diebold Nixdorf	NÃƒÂ¸rresundby	267379103
37	NCR	Abildgaard	265639616
32	NCR	Frederikshavn	219812287
33	NCR	Skagen	214127315
1	NCR	NÃƒÂ¦stved	213715474
4	NCR	BrÃƒÂ¸nderslev	212883099
11	NCR	SÃƒÂ¦by	205905693

6. Number of failed ATM transactions across various card types

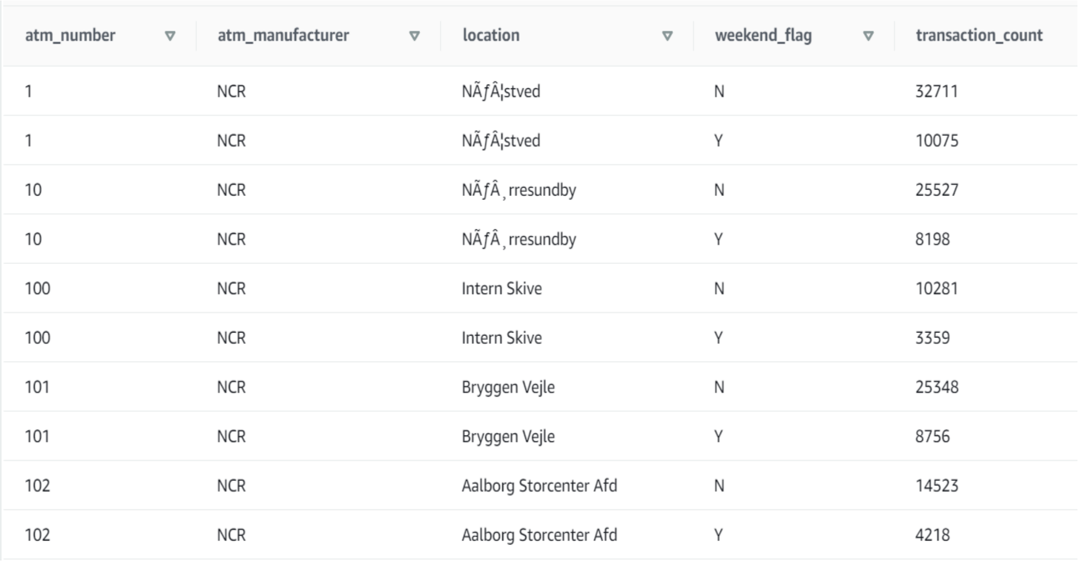
```
select
    a.card_type, a. Total_transaction_count , b. Inactive_transaction_count, ROUND(b.Inactive_transaction_count * 100.00 /
a.Total_transaction_count, 4 )  as Inactive_Percentage
    from
    (select
        b.card_type ,count(1) as Total_transaction_count
    From
        DE_ASSIGN.FACT_ATM_TRANS a left
    join
        DE_ASSIGN.DIM_CARD_TYPE B on a.card_type_id = b.card_type_id
    group
        by  b.card_type
    ) a inner join
    (  select
        b.card_type, count(1) as Inactive_transaction_count
     From
        DE_ASSIGN.FACT_ATM_TRANS a left
    join
        DE_ASSIGN.DIM_CARD_TYPE B on a.card_type_id = b.card_type_id
    where  a.atm_status='Inactive'
    group       by  b.card_type
    ) b
on (
    a. card_type = b.card_type
)  order by Inactive_Percentage desc;
```



card_type	total_transaction_count	inactive_transaction_count	inactive_percentage
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	400506	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

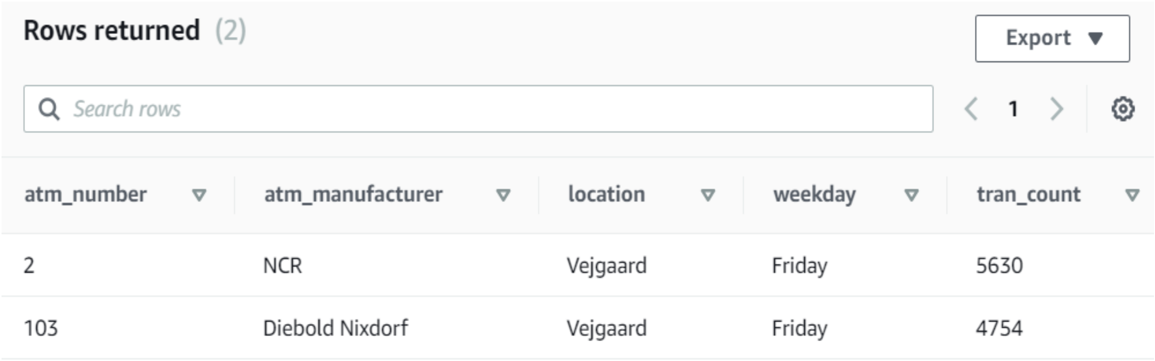
```
 select * from (
select atm_number,atm_manufacturer, d.location, 'Y' as Weekend_flag, count(1) as Transaction_count
  From
        DE_ASSIGN.FACT_ATM_TRANS a
    inner join  DE_ASSIGN.DIM_DATE B
         on a.DATE_ID = b.DATE_ID
     inner join  DE_ASSIGN.DIM_ATM C
          on a.atm_id = c.atm_id
     inner join  DE_ASSIGN.DIM_LOCATION D
         on c.atm_location_id = d.location_id
      where b.weekday in ('Sunday','Saturday')
  group by atm_number,atm_manufacturer, d.location
 union all
select atm_number,atm_manufacturer, d.location, 'N' as Weekend_flag, count(1) as Transaction_count
  From
        DE_ASSIGN.FACT_ATM_TRANS a
    inner join  DE_ASSIGN.DIM_DATE B
         on a.DATE_ID = b.DATE_ID
     inner join  DE_ASSIGN.DIM_ATM C
          on a.atm_id = c.atm_id
     inner join  DE_ASSIGN.DIM_LOCATION D
         on c.atm_location_id = d.location_id
      where b.weekday not in ('Sunday','Saturday')
  group by atm_number,atm_manufacturer, d.location )
   order by 1,2,3,4,5;
```



atm_number ▽	atm_manufacturer ▽	location ▽	weekend_flag ▽	transaction_count
1	NCR	NÃƒÂ¦stved	N	32711
1	NCR	NÃƒÂ¦stved	Y	10075
10	NCR	NÃƒÂ¸rresundby	N	25527
10	NCR	NÃƒÂ¸rresundby	Y	8198
100	NCR	Intern Skive	N	10281
100	NCR	Intern Skive	Y	3359
101	NCR	Bryggen Vejle	N	25348
101	NCR	Bryggen Vejle	Y	8756
102	NCR	Aalborg Storcenter Afd	N	14523
102	NCR	Aalborg Storcenter Afd	Y	4218

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

```
/* Created temporary table with counts for each weekdays */
  create table de_assign.temp_analysis_Vejgaard as
  select Weekday,sum(tran_count) as total_count from (
  select c.atm_number,c.atm_manufacturer, d.location, b.Weekday,count(1) as tran_count
  From
        DE_ASSIGN.FACT_ATM_TRANS a
    inner join   DE_ASSIGN.DIM_DATE B
         on a.DATE_ID = b.DATE_ID
     inner join  DE_ASSIGN.DIM_ATM C
          on a.atm_id = c.atm_id
     inner join  DE_ASSIGN.DIM_LOCATION D
         on c.atm_location_id = d.location_id
      where  d.location ='Vejgaard'
  group by  c.atm_number,c.atm_manufacturer, d.location, b.Weekday)
  group by Weekday;
;
/* Claculated the count for most active weekday */
select c.atm_number,c.atm_manufacturer, d.location, b.Weekday,count(1) as tran_count
  From
        DE_ASSIGN.FACT_ATM_TRANS a
    inner join   DE_ASSIGN.DIM_DATE B
         on a.DATE_ID = b.DATE_ID
     inner join  DE_ASSIGN.DIM_ATM C
          on a.atm_id = c.atm_id
     inner join  DE_ASSIGN.DIM_LOCATION D
         on c.atm_location_id = d.location_id
      where  d.location ='Vejgaard'
      and b.Weekday=(select Weekday from (select ROw_number() over (Order by total_count desc) as rank, Weekday From
de_assign.temp_analysis_Vejgaard)  where rank =1)
  group by  c.atm_number,c.atm_manufacturer, d.location, b.Weekday;
```



atm_number	atm_manufacturer	location	weekday	tran_count
2	NCR	Vejgaard	Friday	5630
103	Diebold Nixdorf	Vejgaard	Friday	4754