### Solving analytical queries on Redshift Cluster

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 **Top 10 ATMs with the highest total amount withdrawn throughout the year**

SELECT

a.atm\_number,

a.atm\_manufacturer,

l.location,

SUM(f.transaction\_amount) as total\_amount\_withdrawn

FROM

etl\_atm\_data.DIM\_ATM as a,

etl\_atm\_data.FACT\_ATM\_TRANS as f,

etl\_atm\_data.DIM\_LOCATION as l

WHERE

a.atm\_id = f.atm\_id

AND a.atm\_location\_id = l.location\_id

GROUP BY

a.atm\_number,

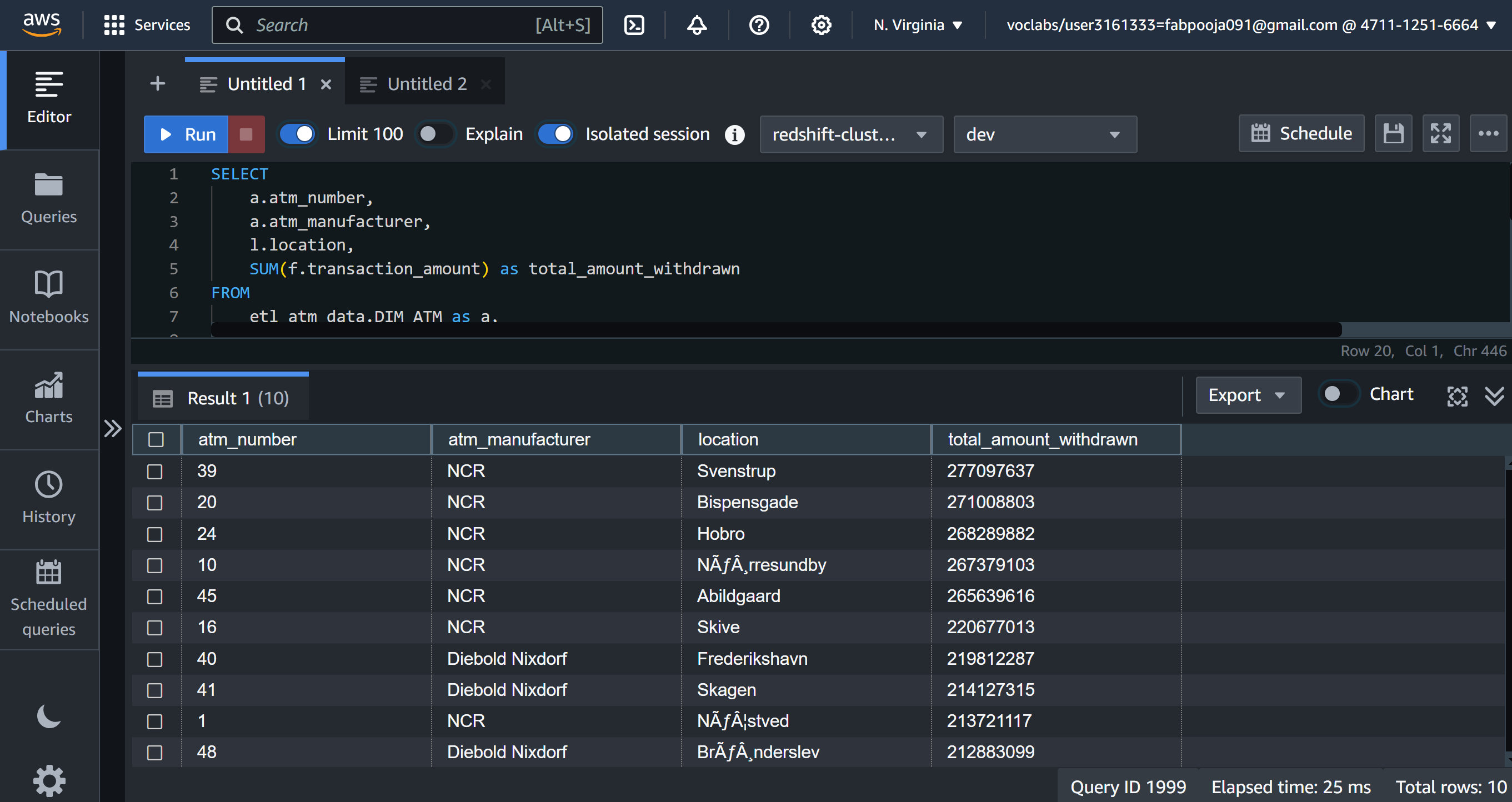
a.atm\_manufacturer,

l.location

ORDER BY

total\_amount\_withdrawn DESC

LIMIT 10;

****

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

Query

SELECT ct.card\_type,

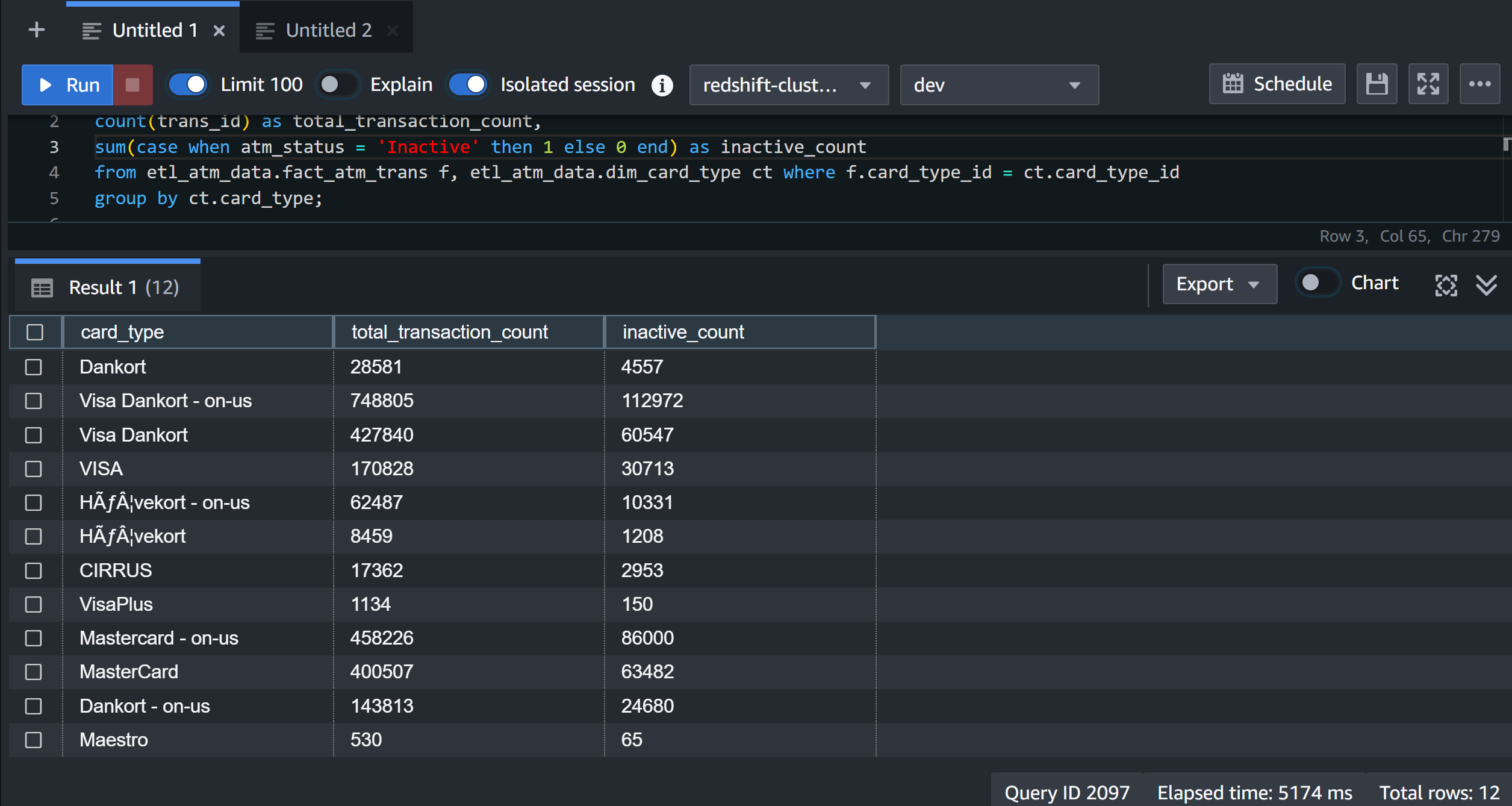
COUNT(trans\_id) as total\_transaction\_count,

SUM(case when atm\_status = 'Inactive' then 1 else 0 end) as inactive\_count

FROM etl\_atm\_data.fact\_atm\_trans f, etl\_atm\_data.dim\_card\_type ct

WHERE f.card\_type\_id = ct.card\_type\_id

GROUP BY ct.card\_type;

****

1. **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

a.atm\_number,

a.atm\_manufacturer,

l.location,

CASE

WHEN d.weekday IN ('Saturday', 'Sunday') THEN 1

ELSE 0

END AS weekend\_flag,

COUNT(f.trans\_id) AS total\_transaction\_count

FROM

etl\_atm\_data.fact\_atm\_trans f

JOIN

etl\_atm\_data.dim\_atm a ON f.atm\_id = a.atm\_id

JOIN

etl\_atm\_data.dim\_location l ON a.atm\_location\_id = l.location\_id

JOIN

etl\_atm\_data.dim\_date d ON f.date\_id = d.date\_id

GROUP BY

a.atm\_number,

a.atm\_manufacturer,

l.location,

weekend\_flag

ORDER BY

a.atm\_number,

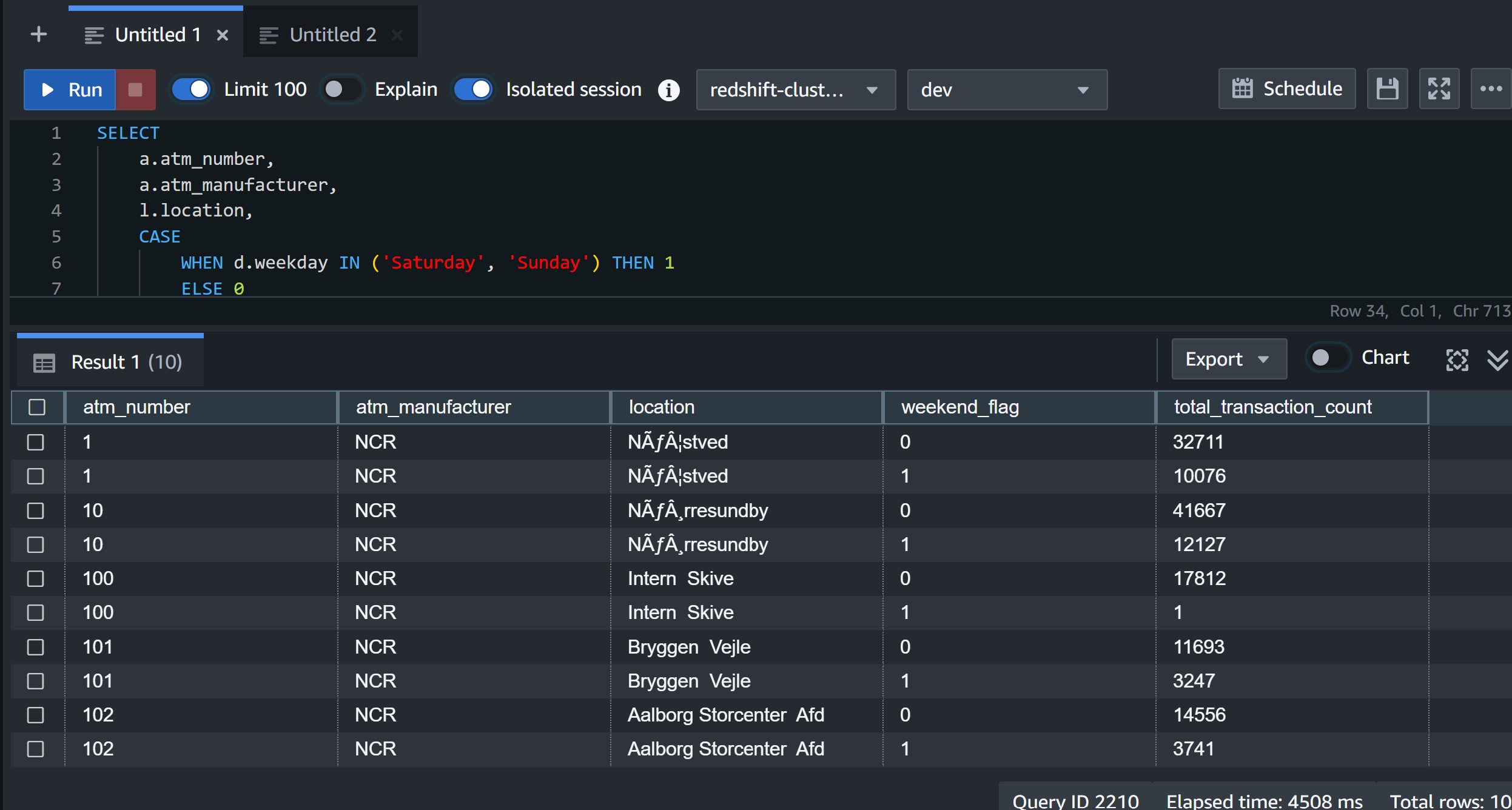
a.atm\_manufacturer,

l.location,

weekend\_flag,

total\_transaction\_count DESC

LIMIT 10;

****

1. **Most active day in each ATMs from location "Vejgaard"**

Query

WITH daily\_transactions AS (

SELECT

a.atm\_number,

a.atm\_manufacturer,

l.location,

d.weekday,

COUNT(f.trans\_id) AS transaction\_count

FROM

etl\_atm\_data.fact\_atm\_trans f

INNER JOIN

etl\_atm\_data.dim\_atm a ON f.atm\_id = a.atm\_id

INNER JOIN

etl\_atm\_data.dim\_location l ON a.atm\_location\_id = l.location\_id

INNER JOIN

etl\_atm\_data.dim\_date d ON f.date\_id = d.date\_id

WHERE

l.location = 'Vejgaard'

GROUP BY

a.atm\_number,

a.atm\_manufacturer,

l.location,

d.weekday

),

ranked\_transactions AS (

SELECT

\*,

ROW\_NUMBER() OVER (PARTITION BY atm\_number ORDER BY transaction\_count DESC) AS rank

FROM

daily\_transactions

)

SELECT

atm\_number,

atm\_manufacturer,

location,

weekday,

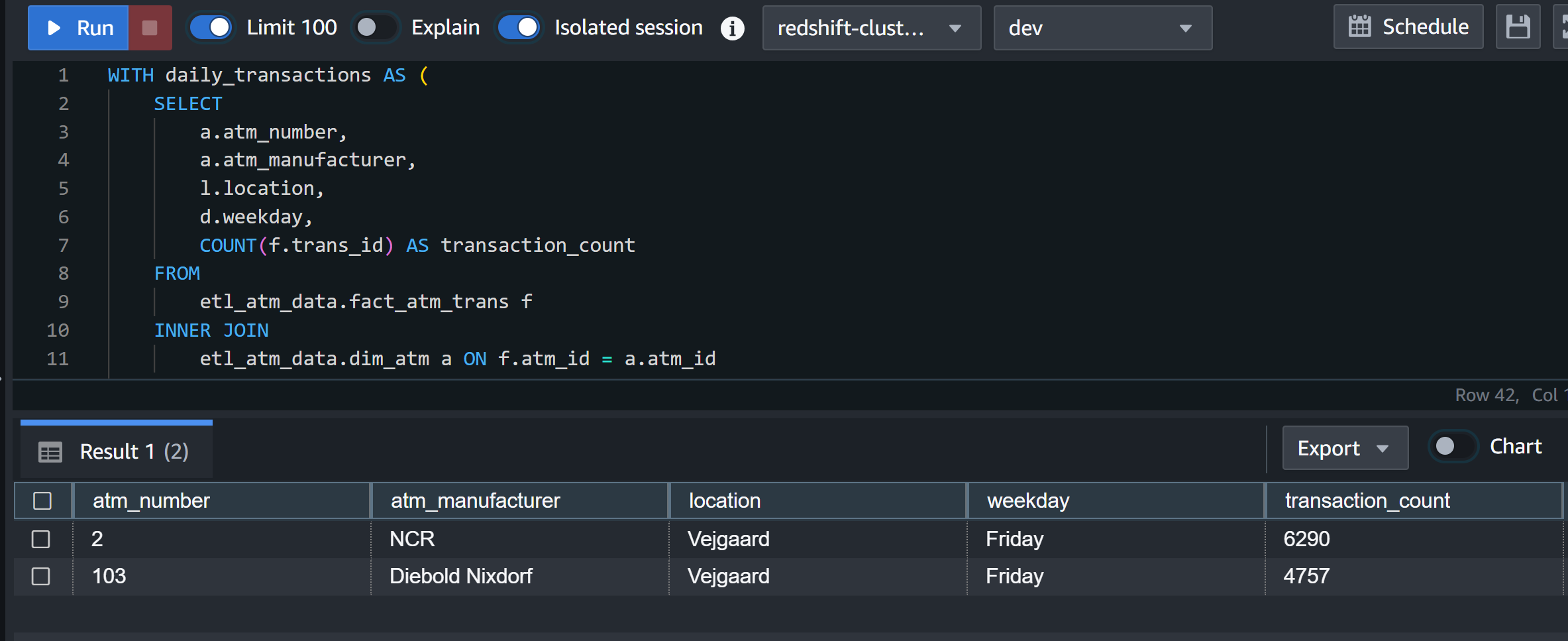
transaction\_count

FROM

ranked\_transactions

WHERE

rank = 1;

****