Roser Blasco
Sprint 4
17 Desembre 2024
Corregit per Vanessa Marina Detto

Base de dades

Partint d'alguns arxius CSV dissenyaràs i crearàs la teva base de dades.

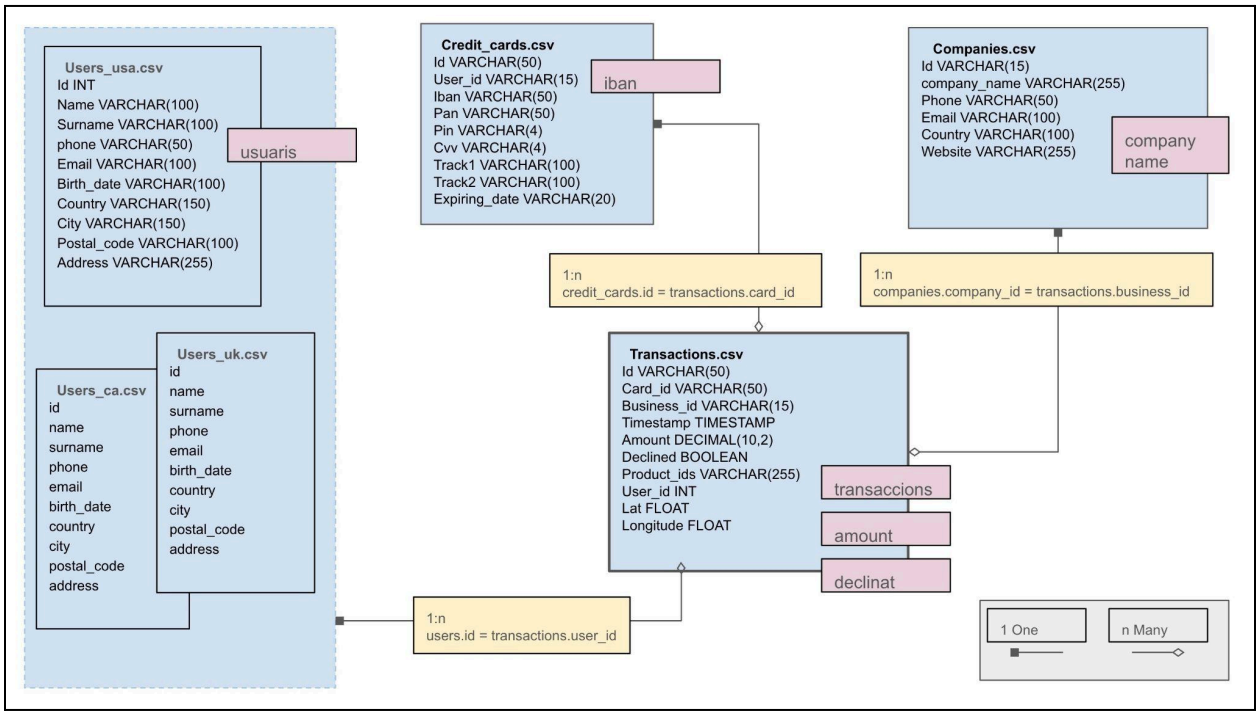
Nivell 1

Descàrrega els arxius CSV, estudia'ls i dissenya una base de dades amb un esquema d'estrella que contingui, almenys 4 taules de les quals puguis realitzar les següents consultes:

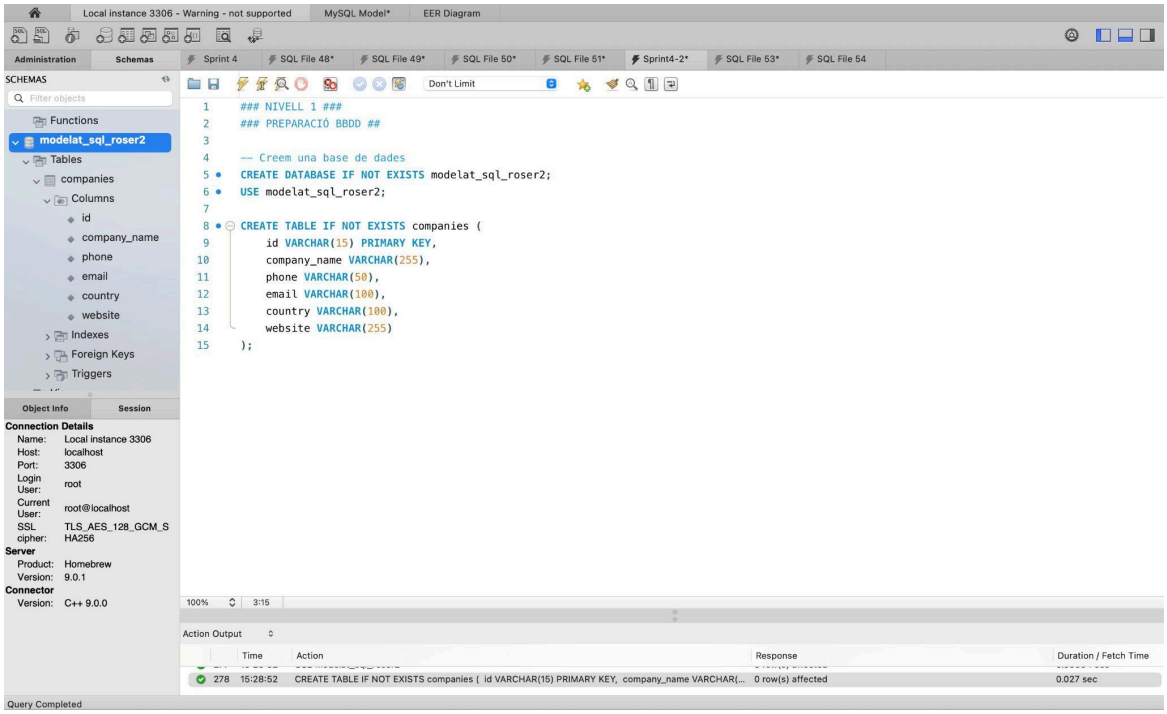
De moment montarem la següent base de dades amb 4 taules amb un esquema d'estrella, on la taula de transaccions serà la taula de fets. Juntarem les taules de users per a simplificar la relació amb credit card i que a més a més, quan es crein nous usuaris, evitar duplicats en el id d'usuari. El usuari de la credit card de moment no el relacionarem perquè no ens interessa de quin usuari és realment la targeta, sinò quin usuari fa la compra.

Deixarem de banda la taula de producte perquè és una relació molts a molts i ho solucionarem en el nivell 3.

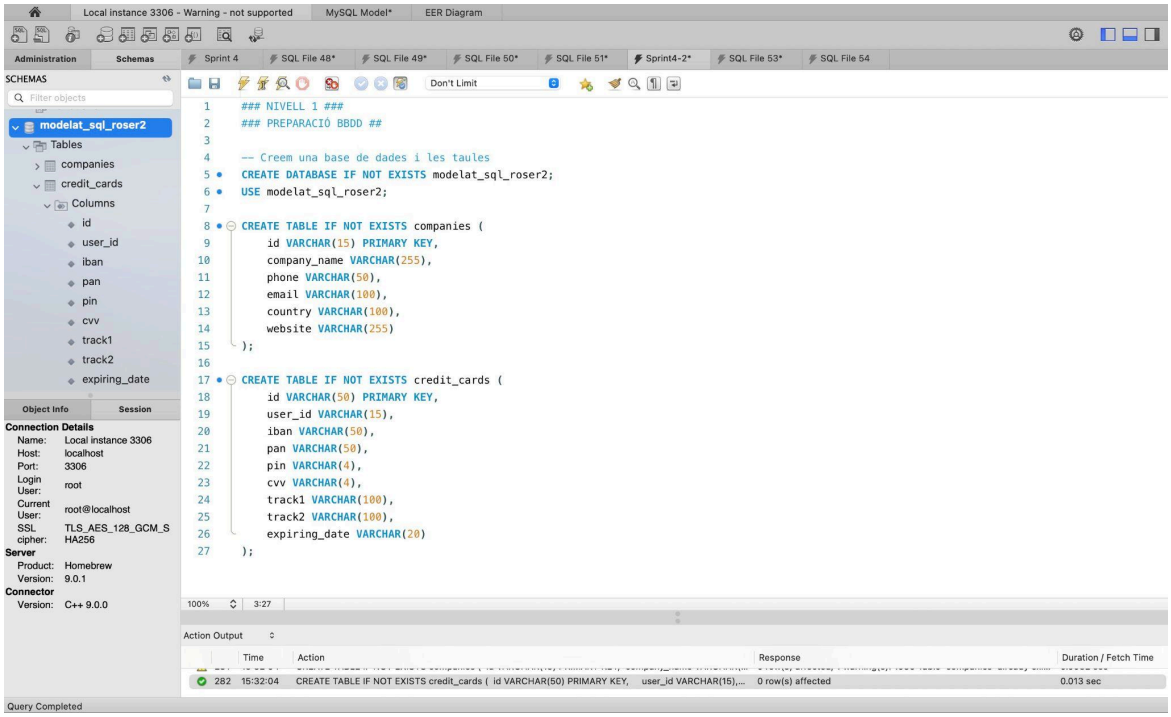
Afegim en rosa les dades que ens demanaran després en els exercicis i en groc la relació entre taules. Hem fet el grafic en power point, per això hem hagut d'inventar els simbols per a 1 i n. Un usuari pot tenir moltes compres, una targeta de crèdit es pot fer servir multiples vegades i les empreses poden tenir moltes vendes. Per això les relacions amb la taula transactions són de 1 a molts.



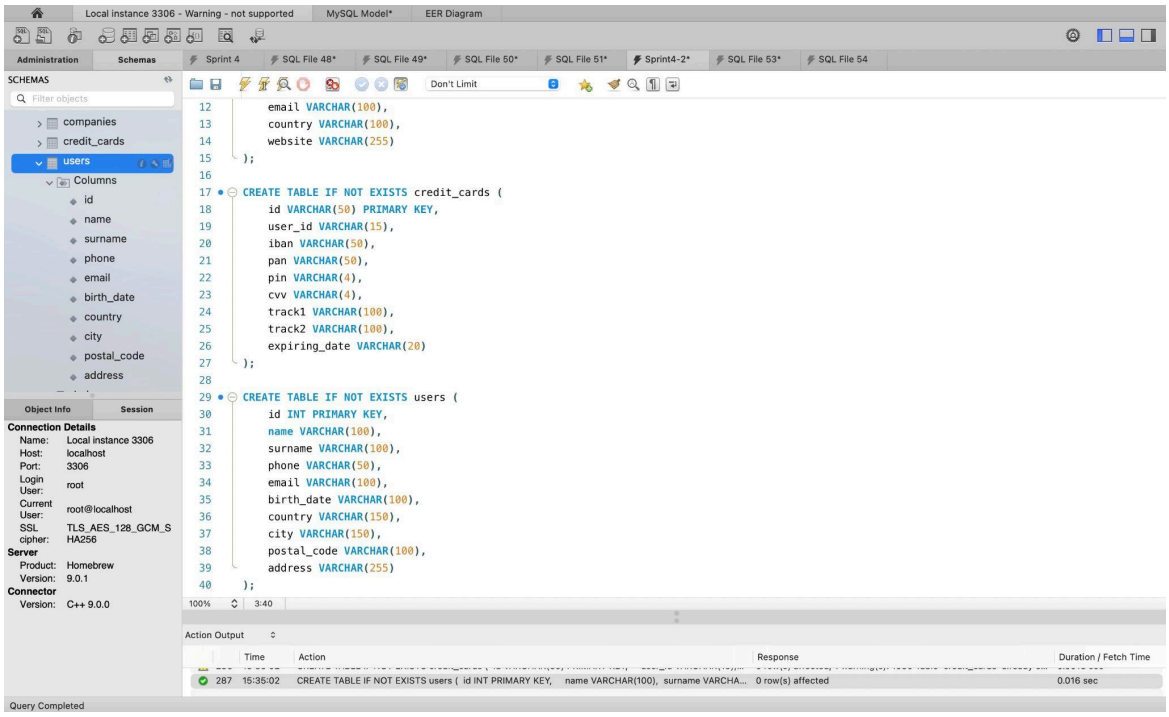
Començarem per crear la base de dades i la taula companies


Local instance 3306 - Warning - not supported
MySQL Model*
EER Diagram
Administration
Schemas
Sprint 4
SQL File 48*
SQL File 49*
SQL File 50*
SQL File 51*
Sprint4-2*
SQL File 53*
SQL File 54
SCHEMAS
Filter objects
Functions
modelat_sql_roser2
Tables
companies
Columns
id
company_name
phone
email
country
website
Indexes
Foreign Keys
Triggers
Object Info
Session
Connection Details
Name: Local instance 3306
Host: localhost
Port: 3306
Login User: root
Current User: root@localhost
SSL cipher: TLS_AES_128_GCM_SHA256
Server
Product: Homebrew
Version: 9.0.1
Connector
Version: C++ 9.0.0
Don't Limit
1 ### NIVELL 1 ###
2 ### PREPARACIÓ BBDD ##
3
4 -- Creem una base de dades
5 CREATE DATABASE IF NOT EXISTS modelat_sql_roser2;
6 USE modelat_sql_roser2;
7
8 CREATE TABLE IF NOT EXISTS companies (
9 id VARCHAR(15) PRIMARY KEY,
10 company_name VARCHAR(255),
11 phone VARCHAR(50),
12 email VARCHAR(100),
13 country VARCHAR(100),
14 website VARCHAR(255)
15);
100%
3:15
Action Output
Time
Action
Response
Duration / Fetch Time
278 15:28:52 CREATE TABLE IF NOT EXISTS companies (id VARCHAR(15) PRIMARY KEY, company_name VARCHAR(... 0 row(s) affected 0.027 sec
Query Completed

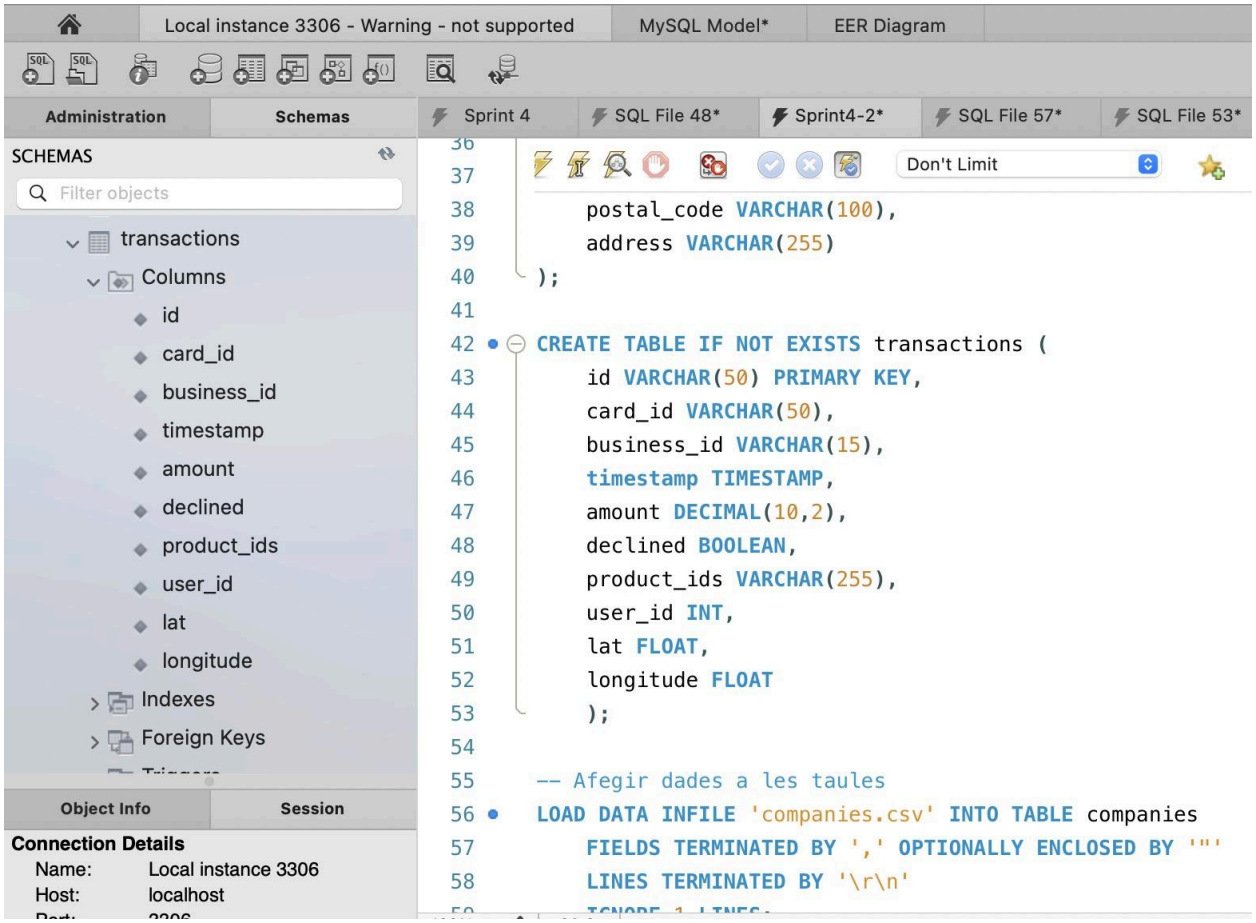
La taula credit cards



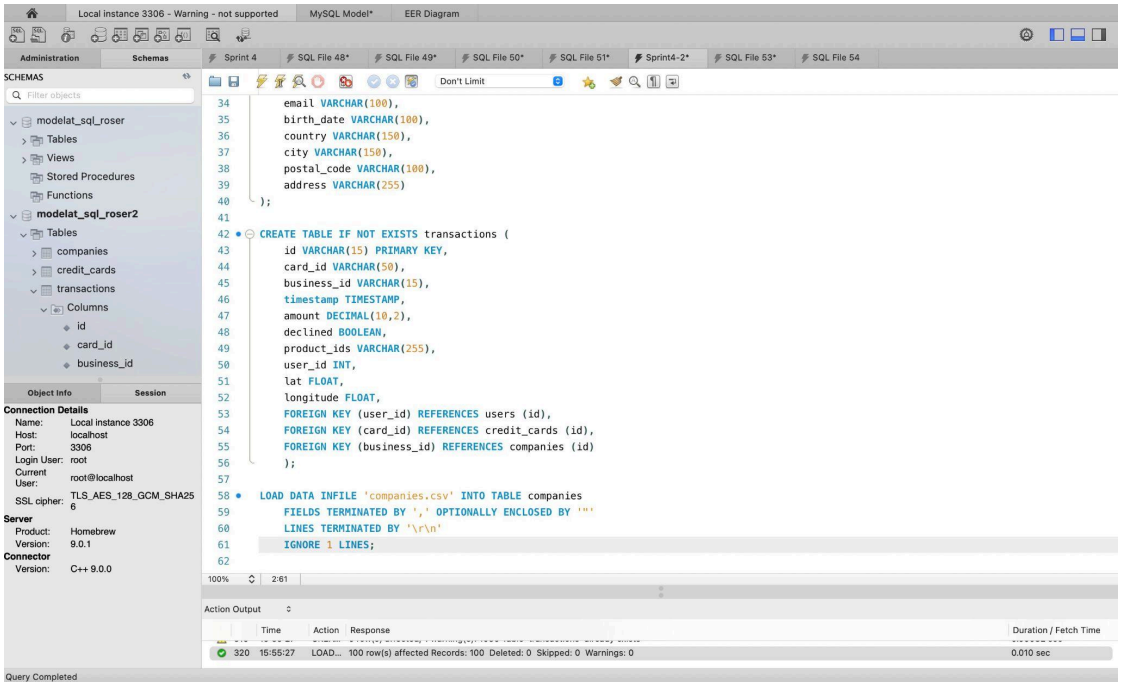
La taula users



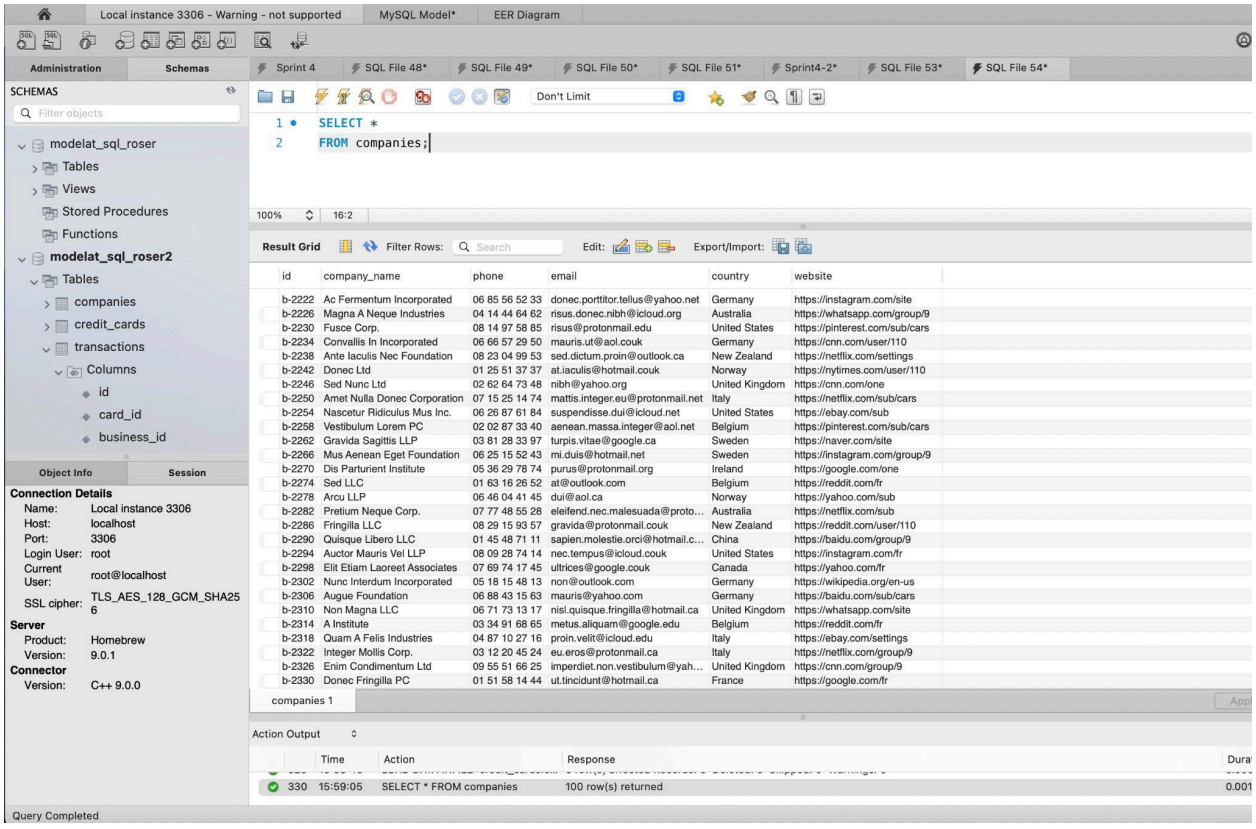
La taula transactions



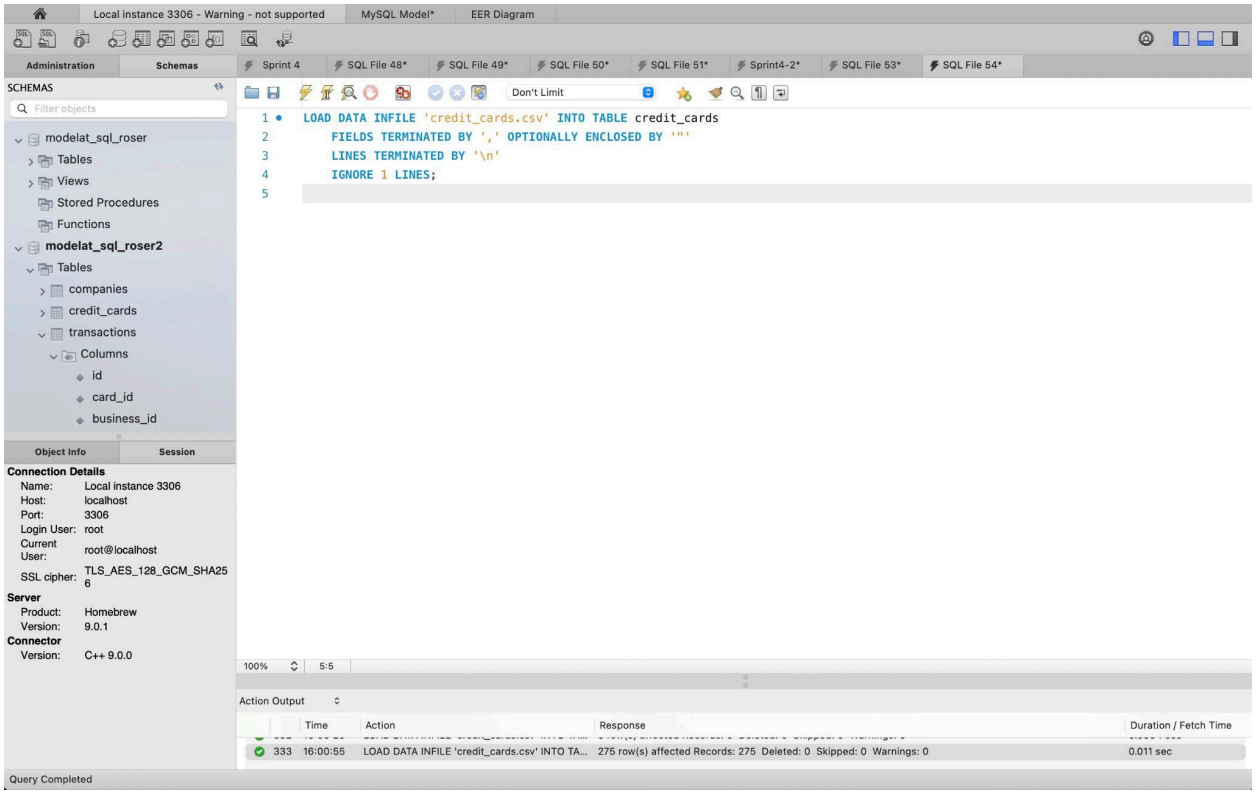
Afegim les dades a la taula companies



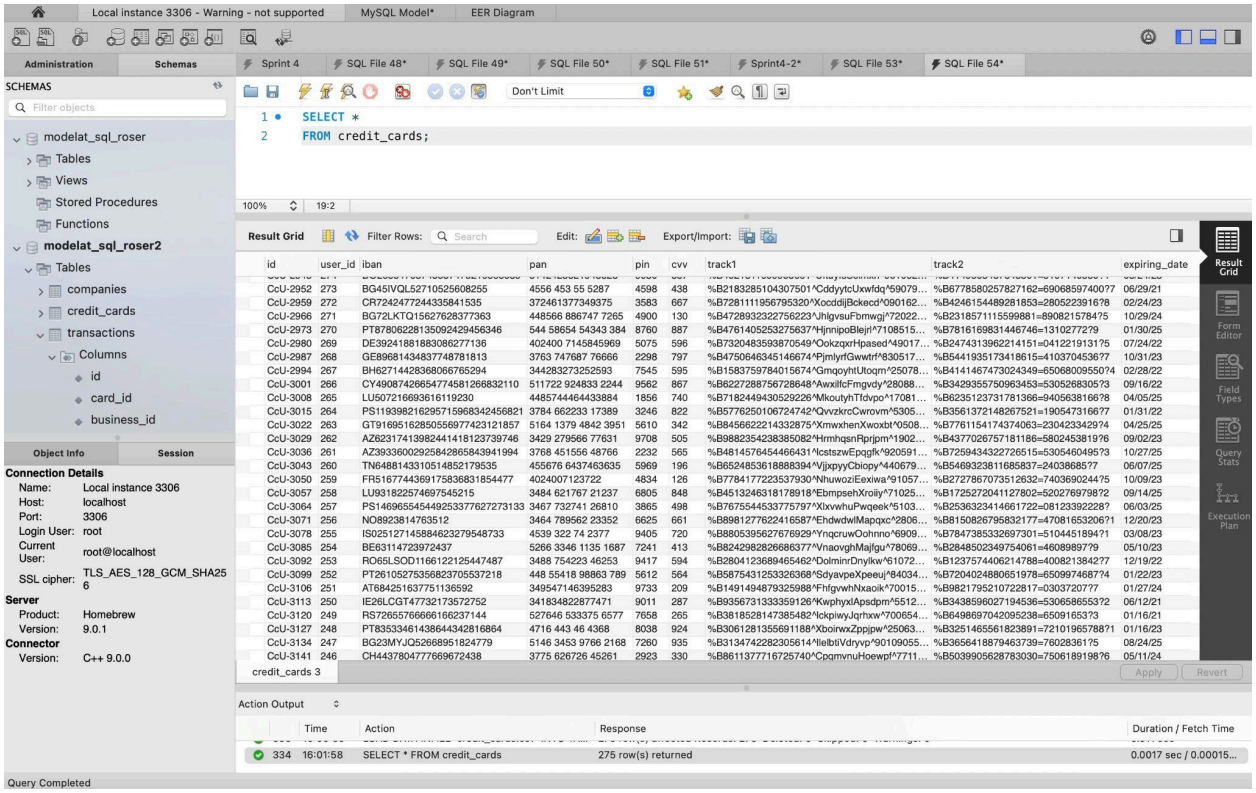
I mirem que estigui bé



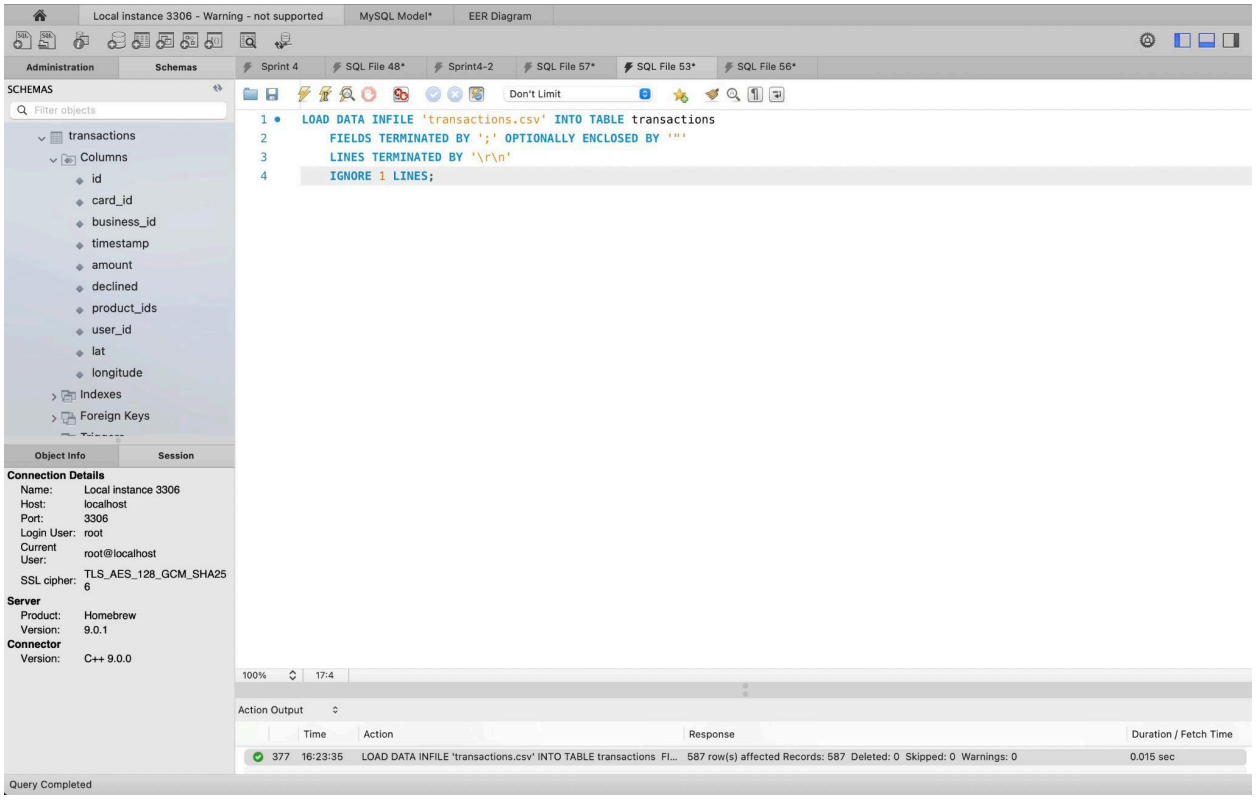
A la taula credit cards, en aquest cas només es carrega sense la r



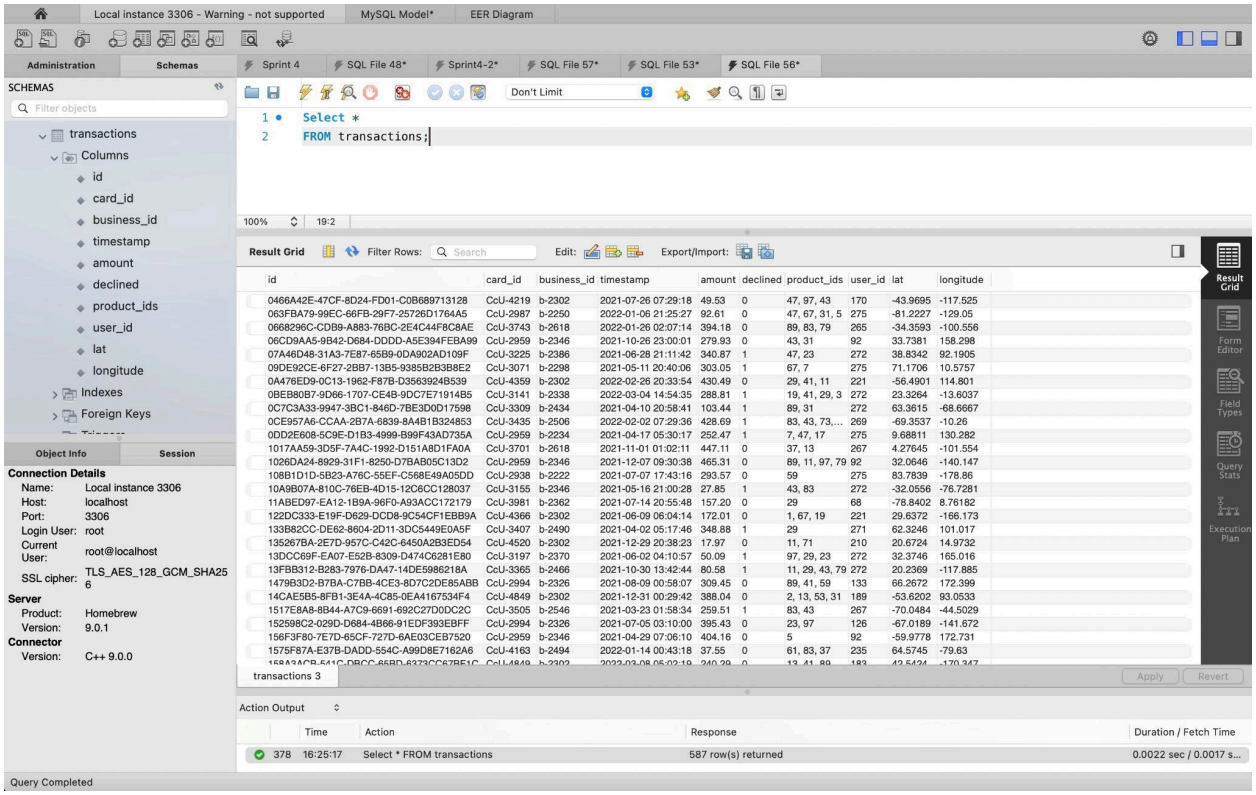
I mirem que estan bé els resultats



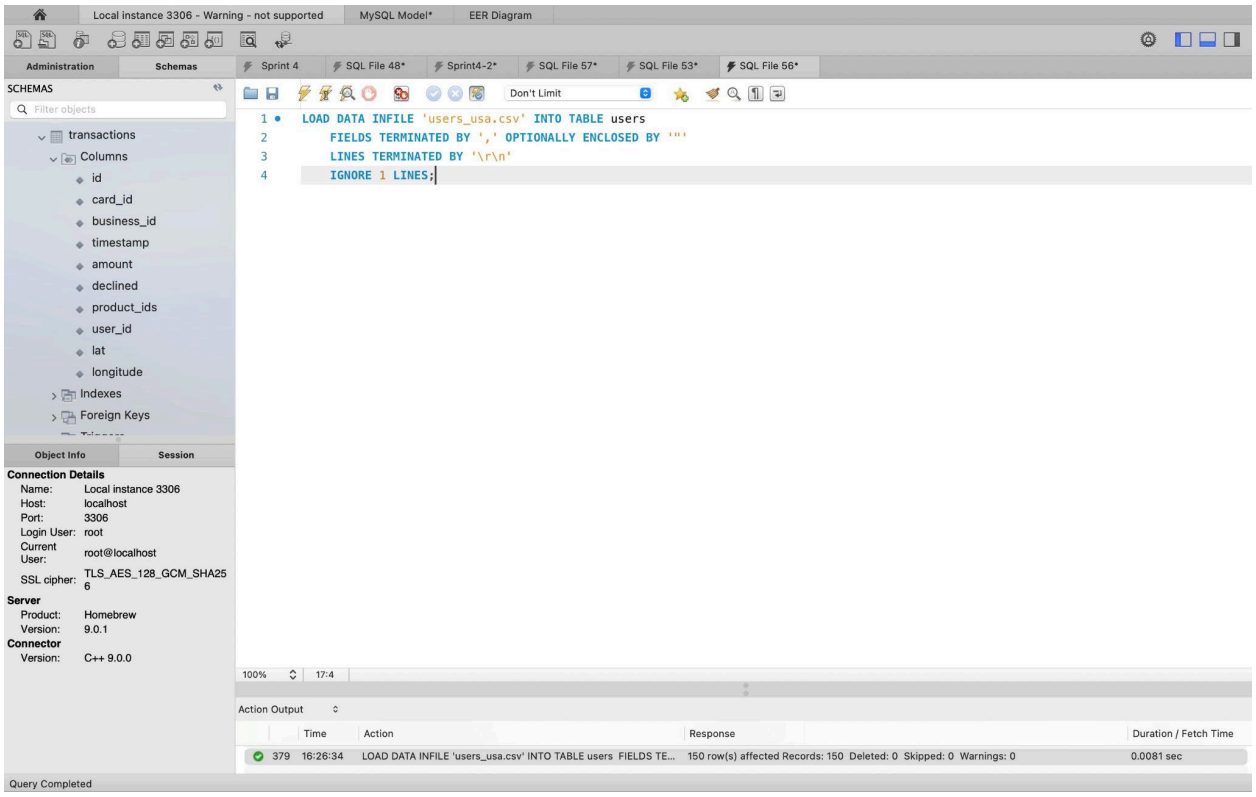
Carreguem la taula transactions. Amb compte que aquí els valors no estan separats per una coma sinò per punt i coma



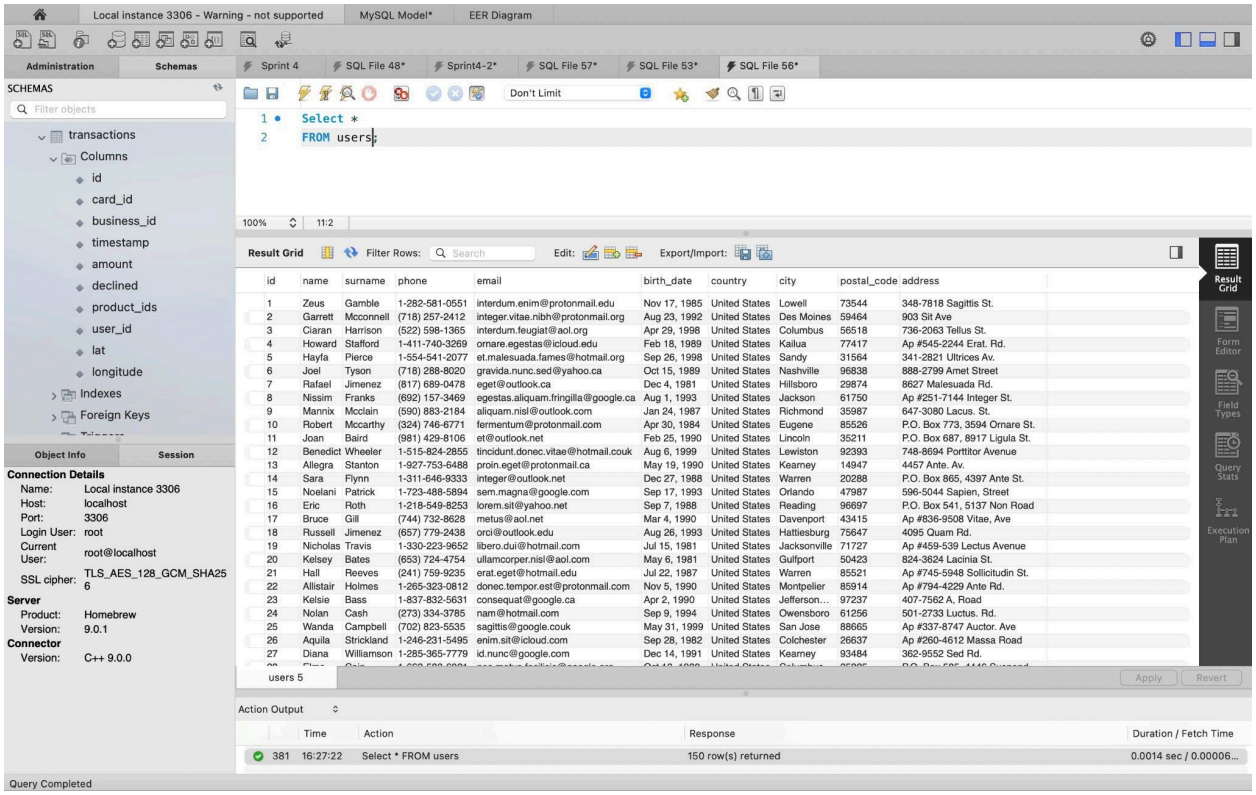
Mirem que estigui bé



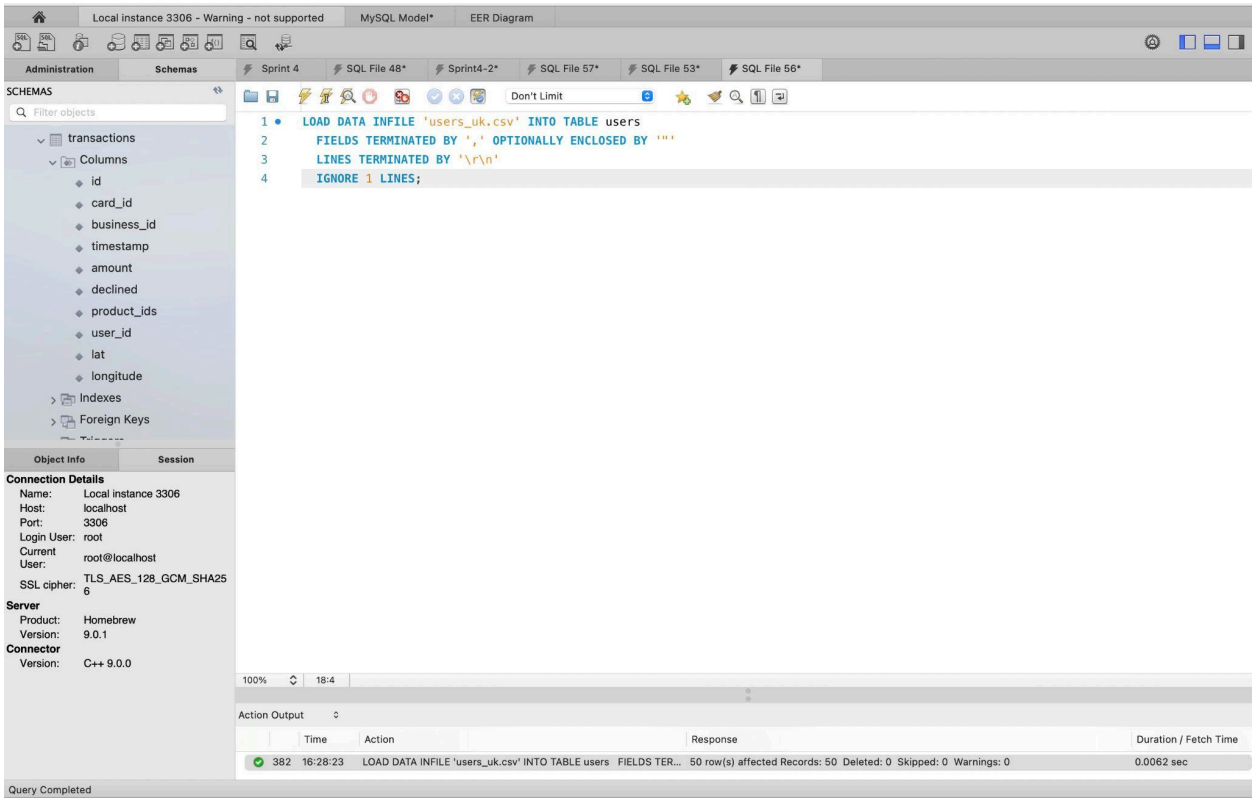
I les dades de la taula users. Primer els de USA



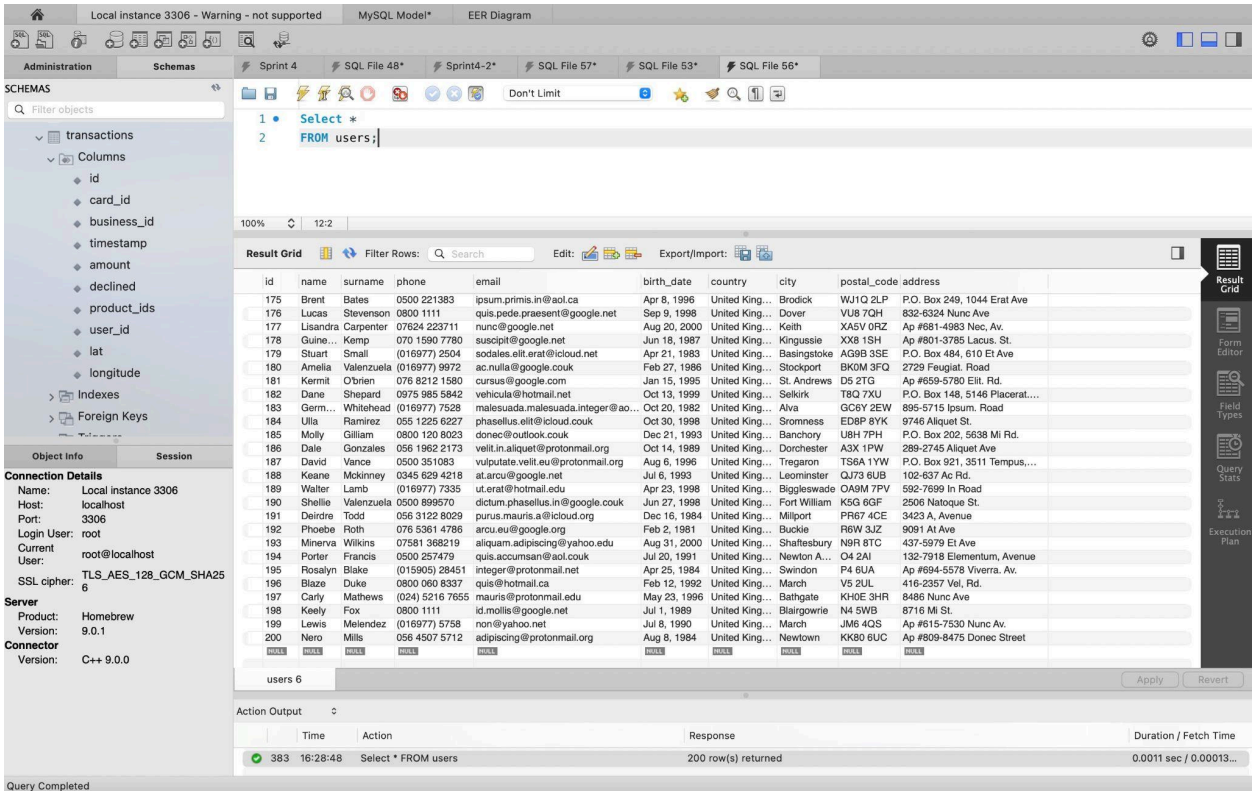
Mirem que estan bé



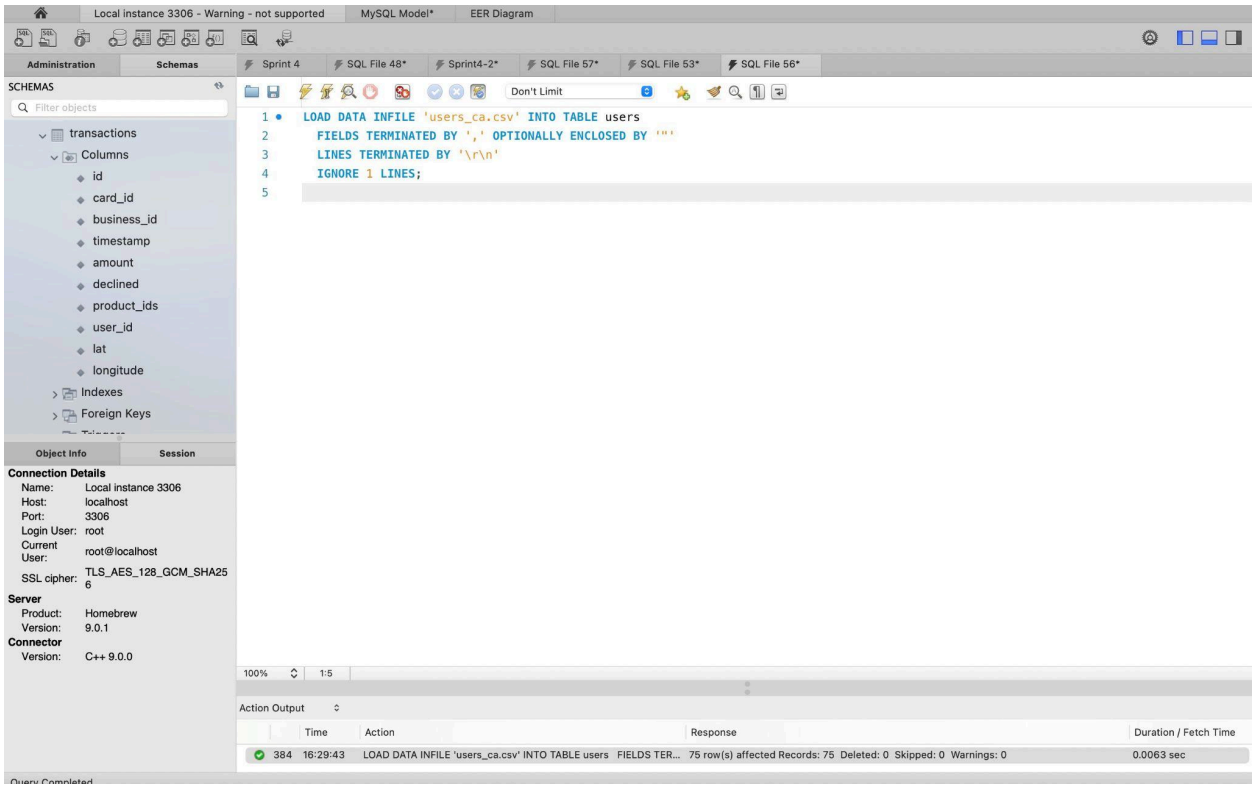
Li afegim les dades de UK



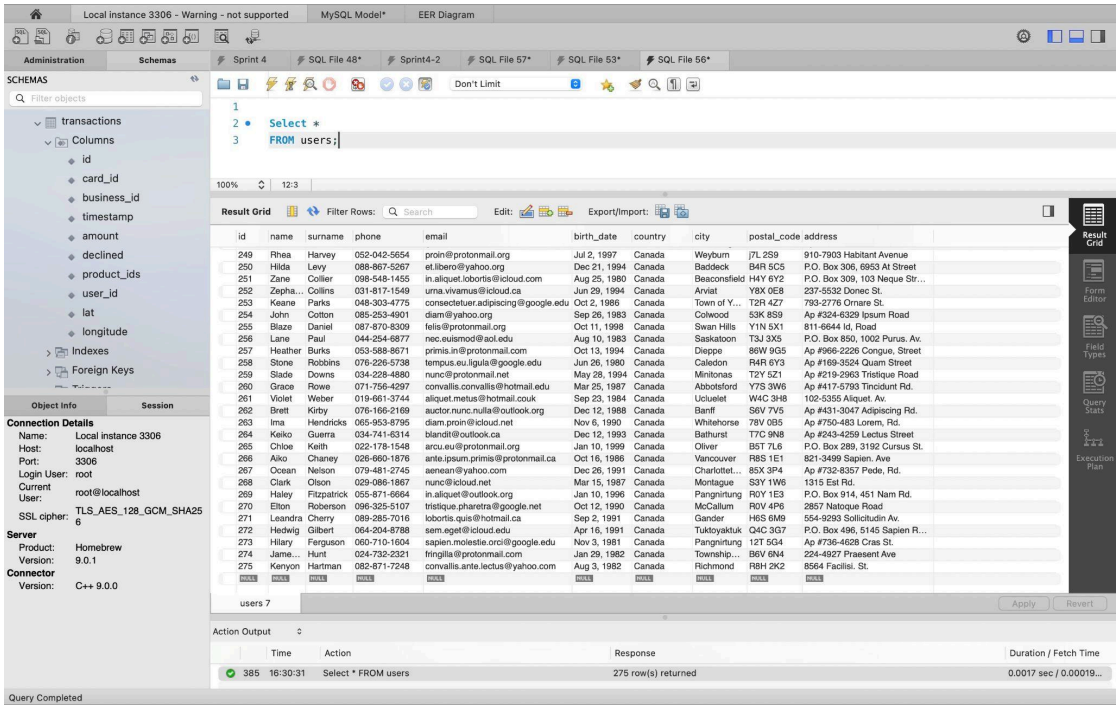
I mirem que estan bé



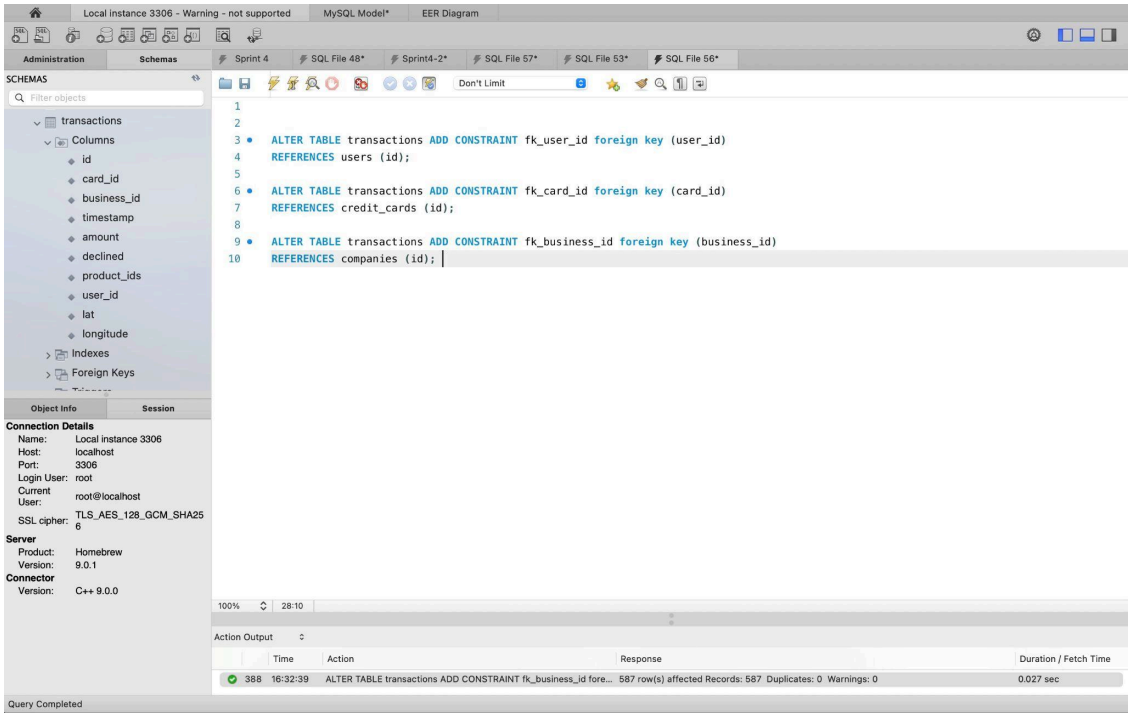
Els de Canadà



I mirem que està bé



Ara afegim les relacions



Li demanem que ens monti un grafic per a comprovar-ho

- Exercici 1: 30 transaccions

Realitza una subconsulta que mostri tots els usuaris amb més de 30 transaccions utilitzant almenys 2 taules.

```sql
-- Usuaris amb més de 30 transaccions
SELECT users.id, users.name AS nom, users.surname AS cognom, COUNT(transactions.id) as numero_transaccions
FROM users
JOIN transactions ON users.id = transactions.user_id
GROUP BY user_id
HAVING numero_transaccions > 30
ORDER BY numero_transaccions DESC;
```

100% 35:7

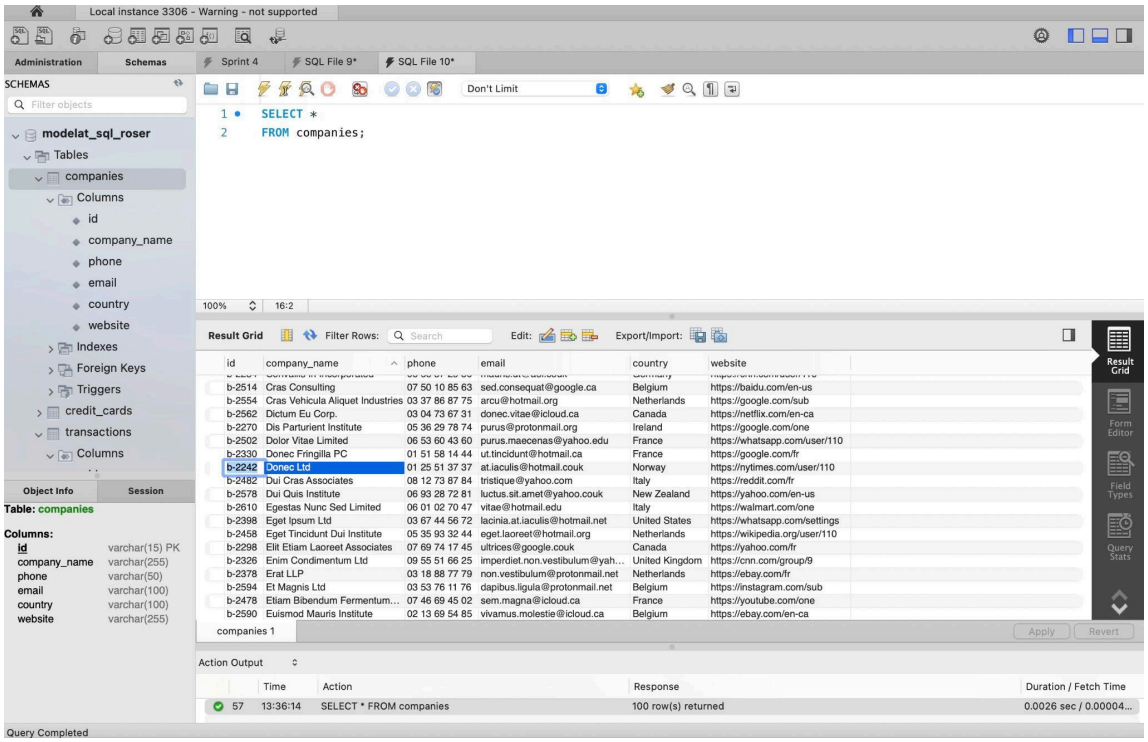
Result Grid Filter Rows: Search Export:

id	nom	cognom	numero_transaccions
272	Hedwig	Gilbert	76
267	Ocean	Nelson	52
275	Kenyon	Hartman	48
92	Lynn	Riddle	39

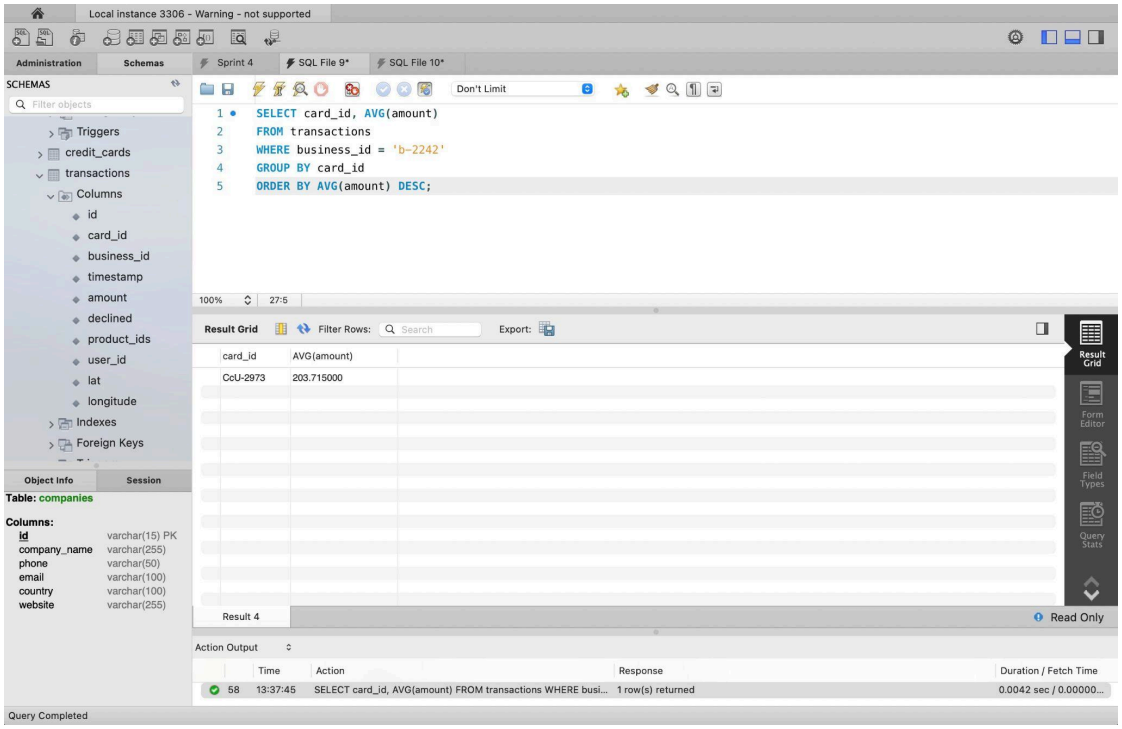
- Exercici 2: mitjana per iban

Mostra la mitjana d'amount per IBAN de les targetes de crèdit a la companyia Donec Ltd, utilitza almenys 2 taules.

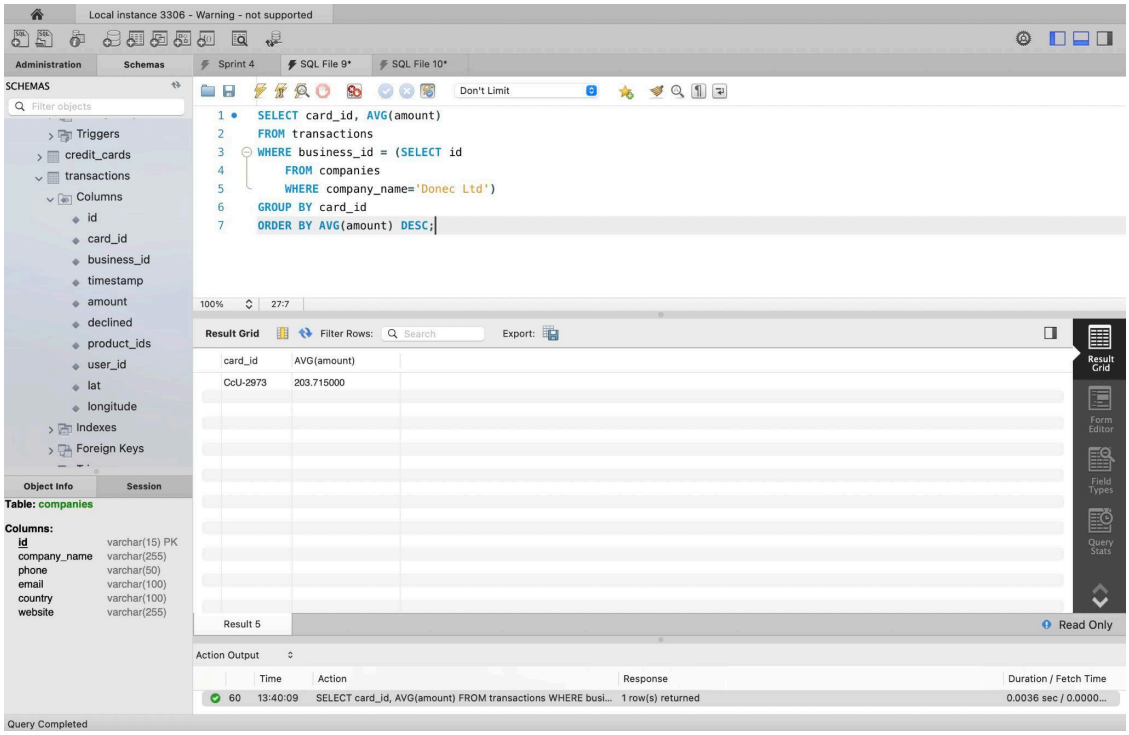
Busquem el id de la companyia Donec, i és b-2242



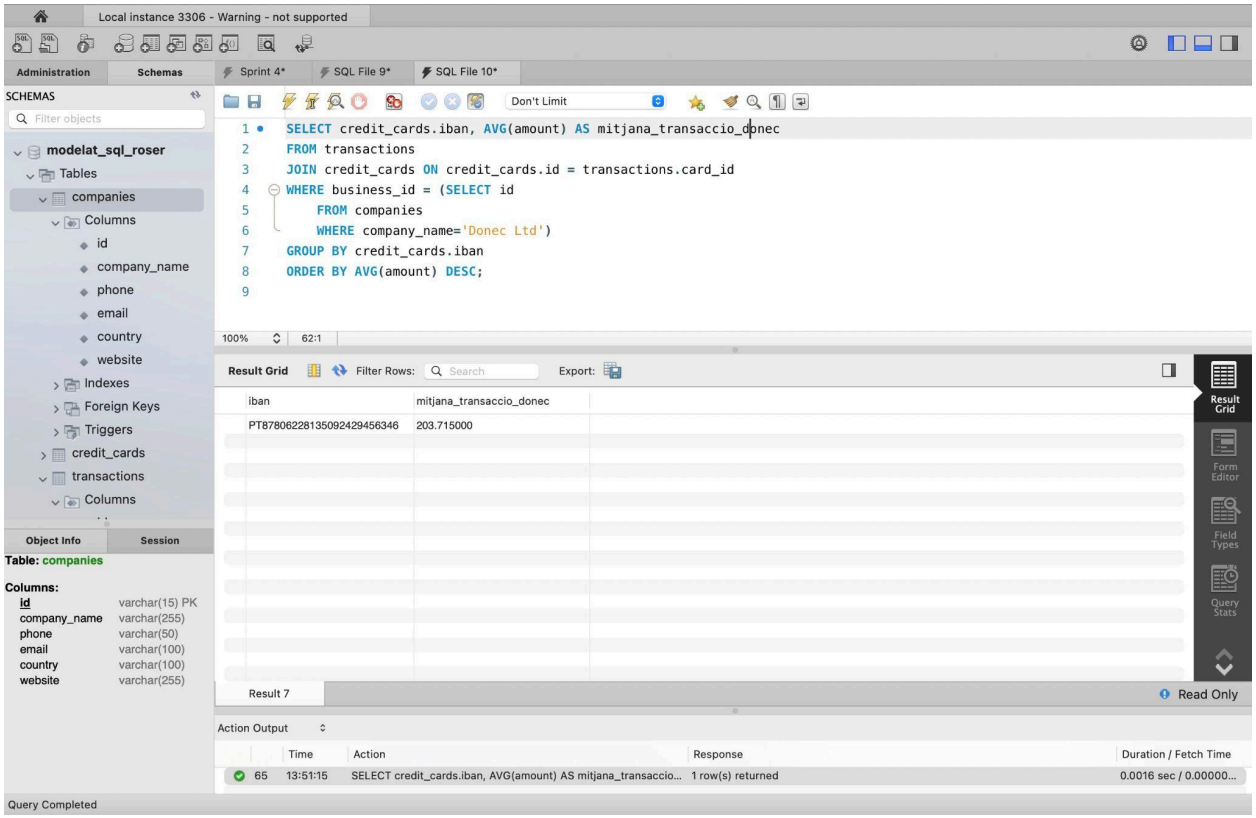
I filtrem els resultats



Ara reemplacem el ID per una subquery. M'ha agradat més fer-ho així que amb una join, em sembla que queda més clar.



Li afegim els ibans



I ho posem bonic

```sql
-- Mitjana d'amount per IBAN de targetes a Donec Ltd
SELECT credit_cards.iban, ROUND(AVG(amount)) AS mitjana_transaccio_donec
FROM transactions
JOIN credit_cards ON credit_cards.id = transactions.card_id
WHERE business_id = (SELECT id
    FROM companies
    WHERE company_name='Donec Ltd')
GROUP BY credit_cards.iban
ORDER BY mitjana_transaccio_donec DESC;
```

100% 40:9

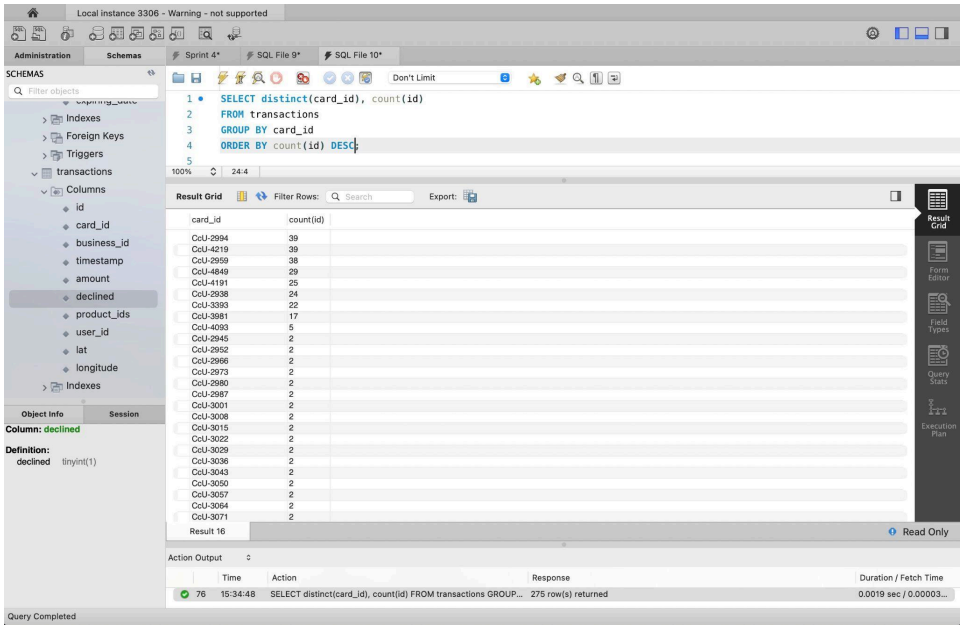
Result Grid Filter Rows: Search Export:

iban	mitjana_transaccio_donec
PT87806228135092429456346	204

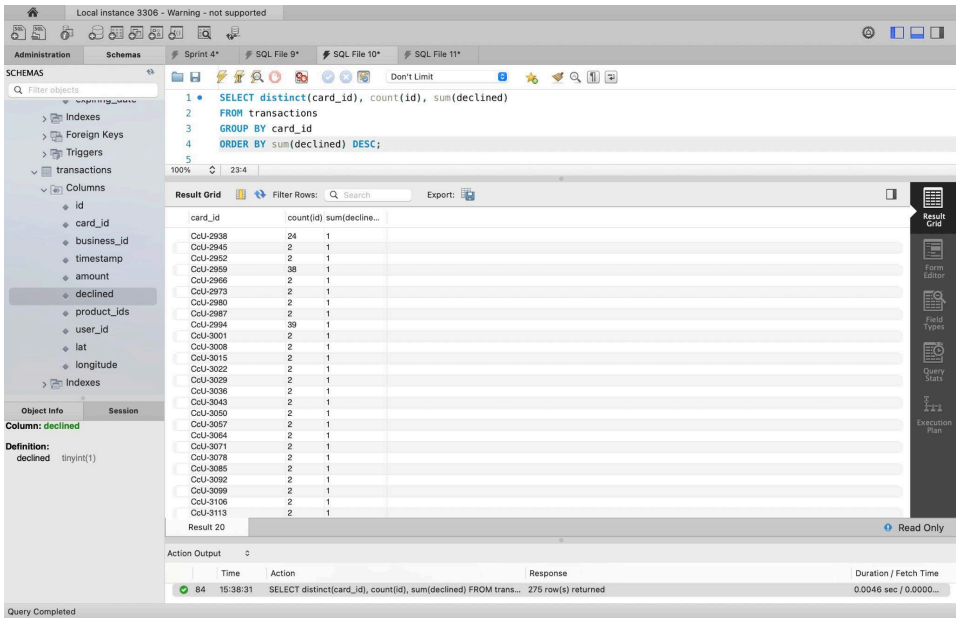
Nivell 2

Crea una nova taula que reflecteixi l'estat de les targetes de crèdit basat en si les últimes tres transaccions van ser declinades i genera la següent consulta:

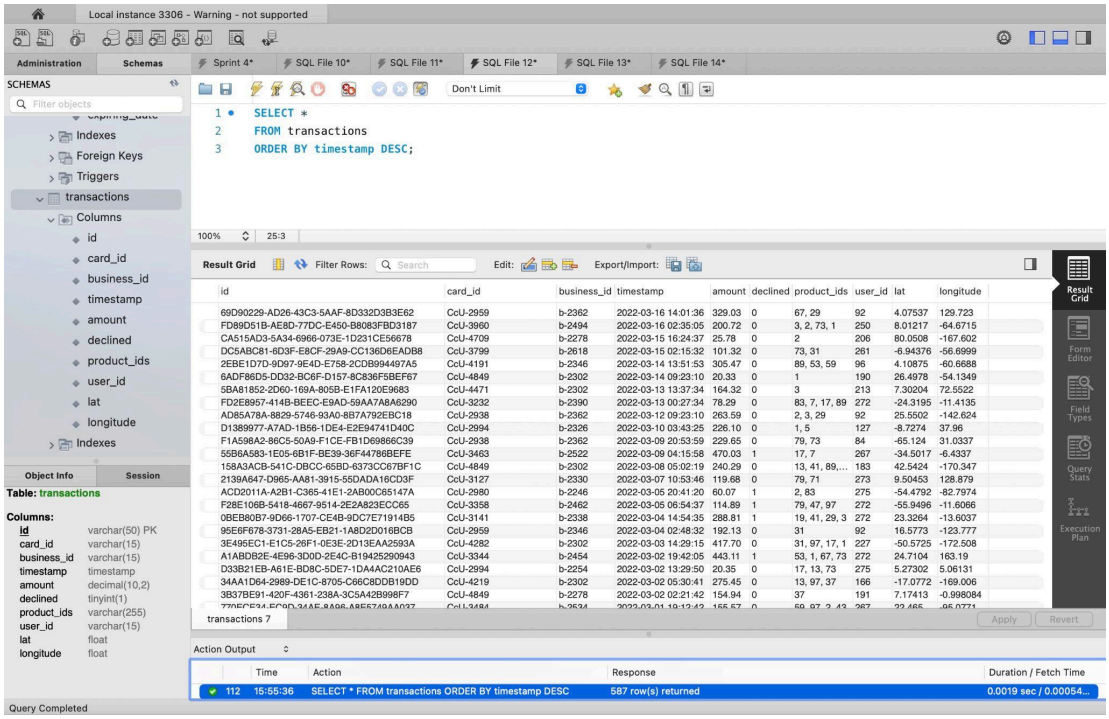
Entenem que les targetes que han segut declinades les últimes tres transaccions són tgt inactives i que la resta són actives. Comencem mirant quantes transaccions ha tingut cada targeta.



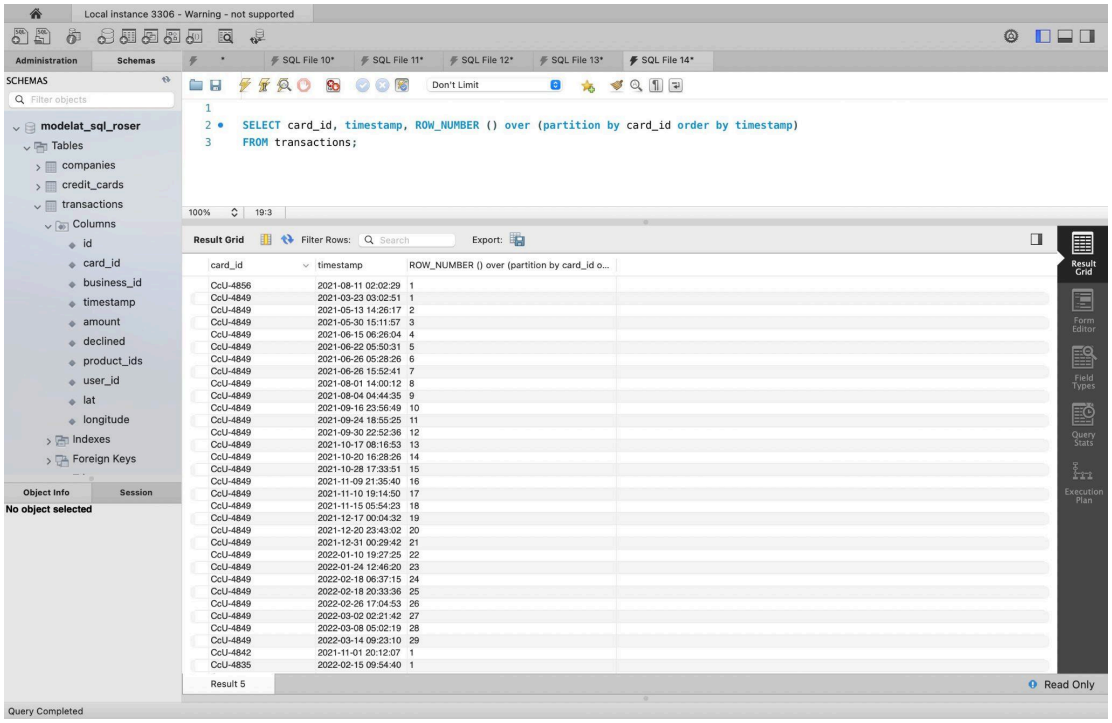
I quantes vegades en total s'ha declinat cada targeta, que no semblen moltes.
Cap targeta s'hagi declinat més d'una vegada.



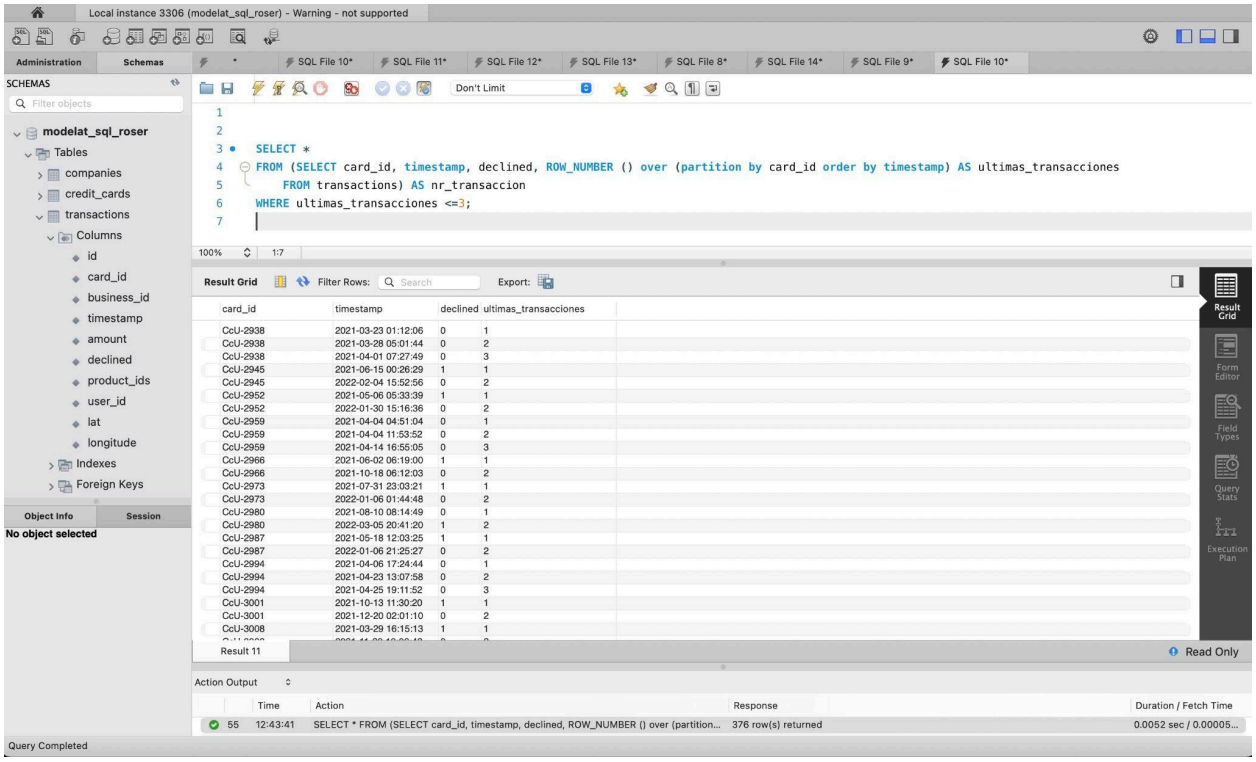
En qualsevol cas, seguim. Ordenem les transaccions de més recents a més antigues



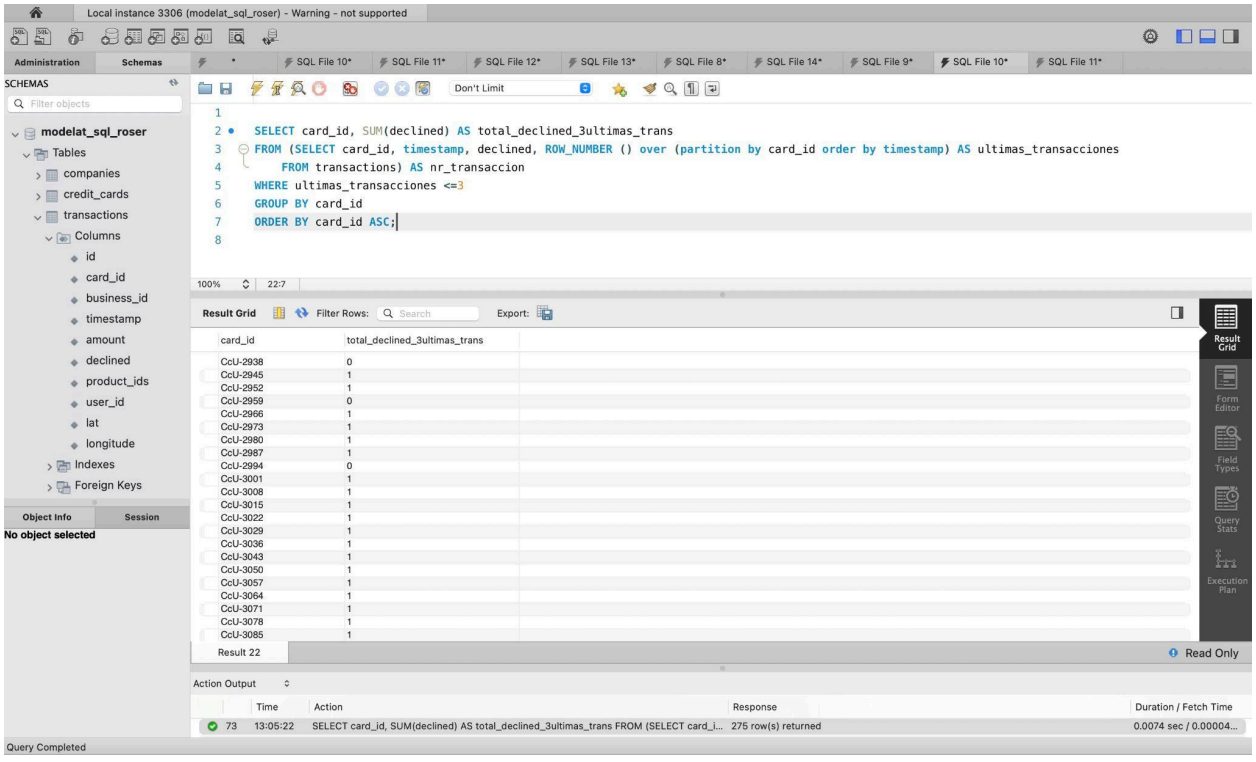
Però ens cal partir açò per targeta.
Experimentem una mica amb ROW_NUMBER i amb el partition by.



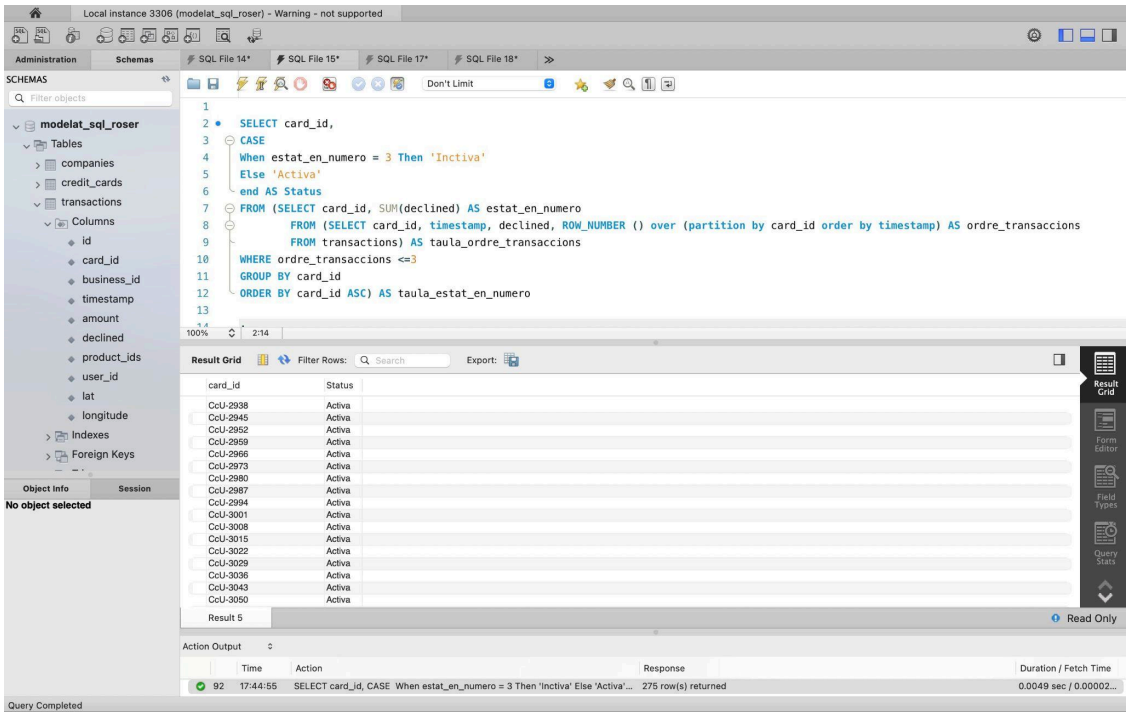
De la llista anterior, només ens interessen les tres últimes transaccions. Això vols dir que tenen un ROW NUMBER 3 o menor.



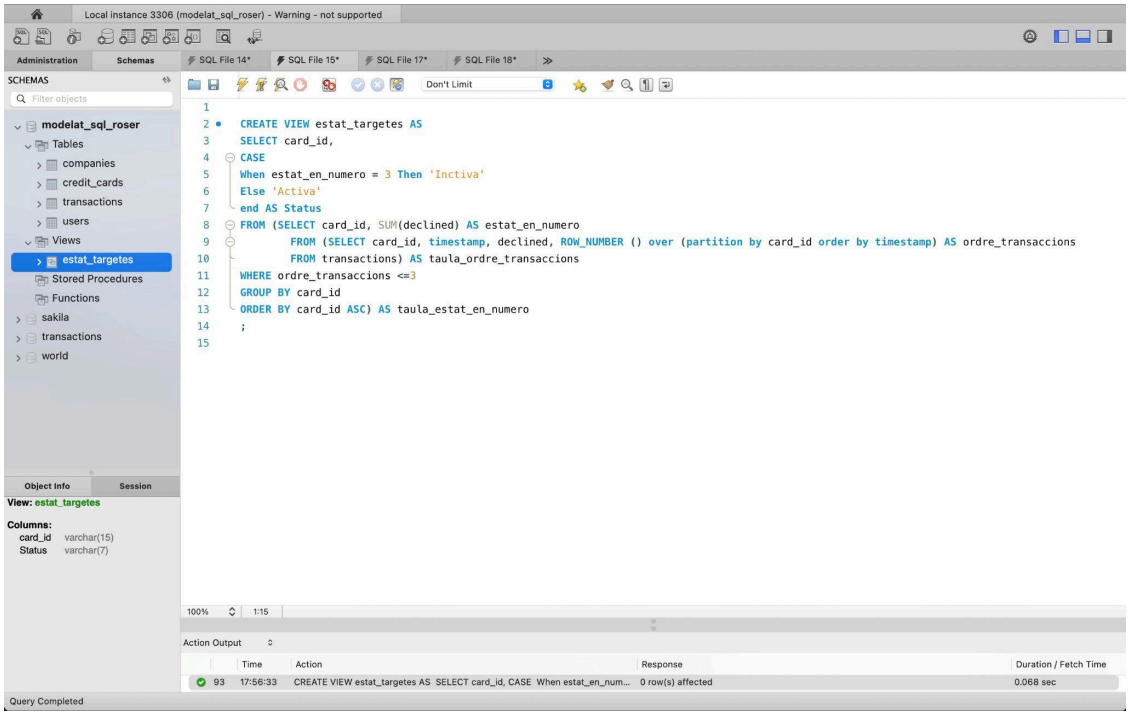
D'aquesta llista, sumarem total dels resultats de declined per a cada targeta. Les que tinguin un 3 estaran Inactives i les que tinguen un nombre menor (totes) estan Actives.



Ara reemplacem la columna amb l'estat en número per una columna amb l'estat com a paraula, utilitzant el case.



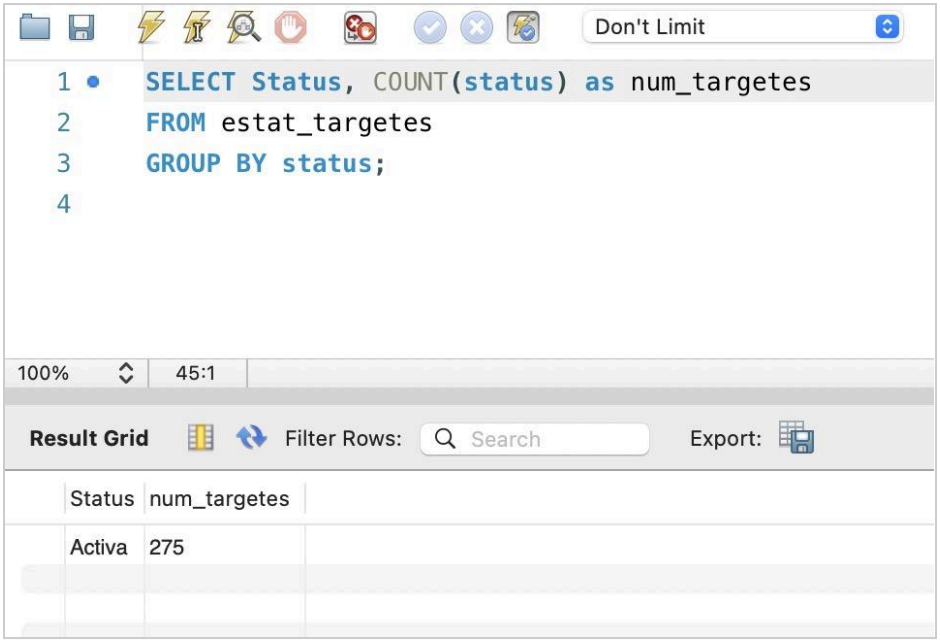
Finalment, ho convetirem en una vista en comptes de taula.
Així, si s'afegeixen transaccions s'actualitzaran els estatus automàticament



- Exercici 1: targetes actives

Quantes targetes estan actives?

Totes les targetes estan actives, 275.

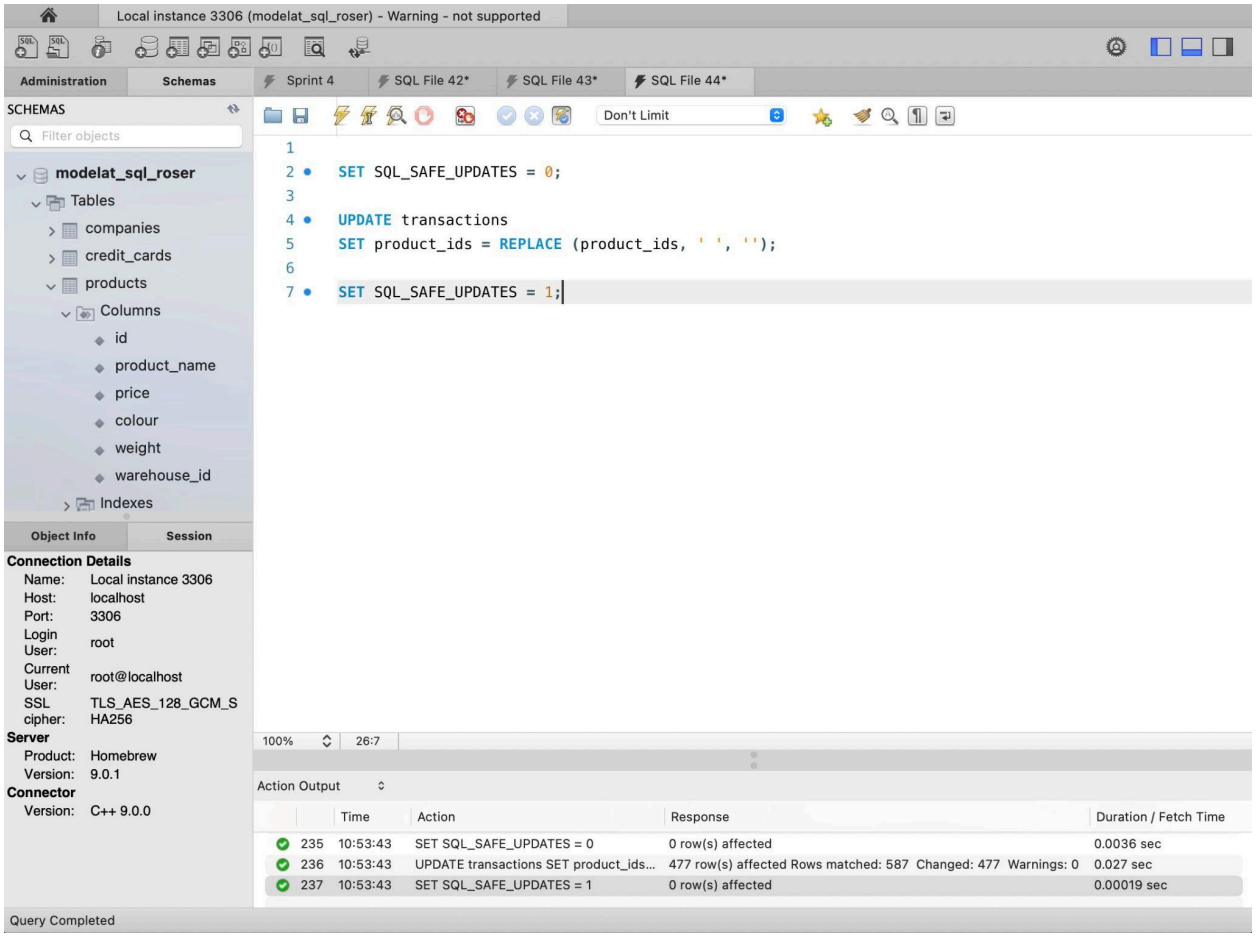


Nivell 3

Crea una taula amb la qual puguem unir les dades del nou arxiu products.csv amb la base de dades creada, tenint en compte que des de transaction tens product_ids. Genera la següent consulta:

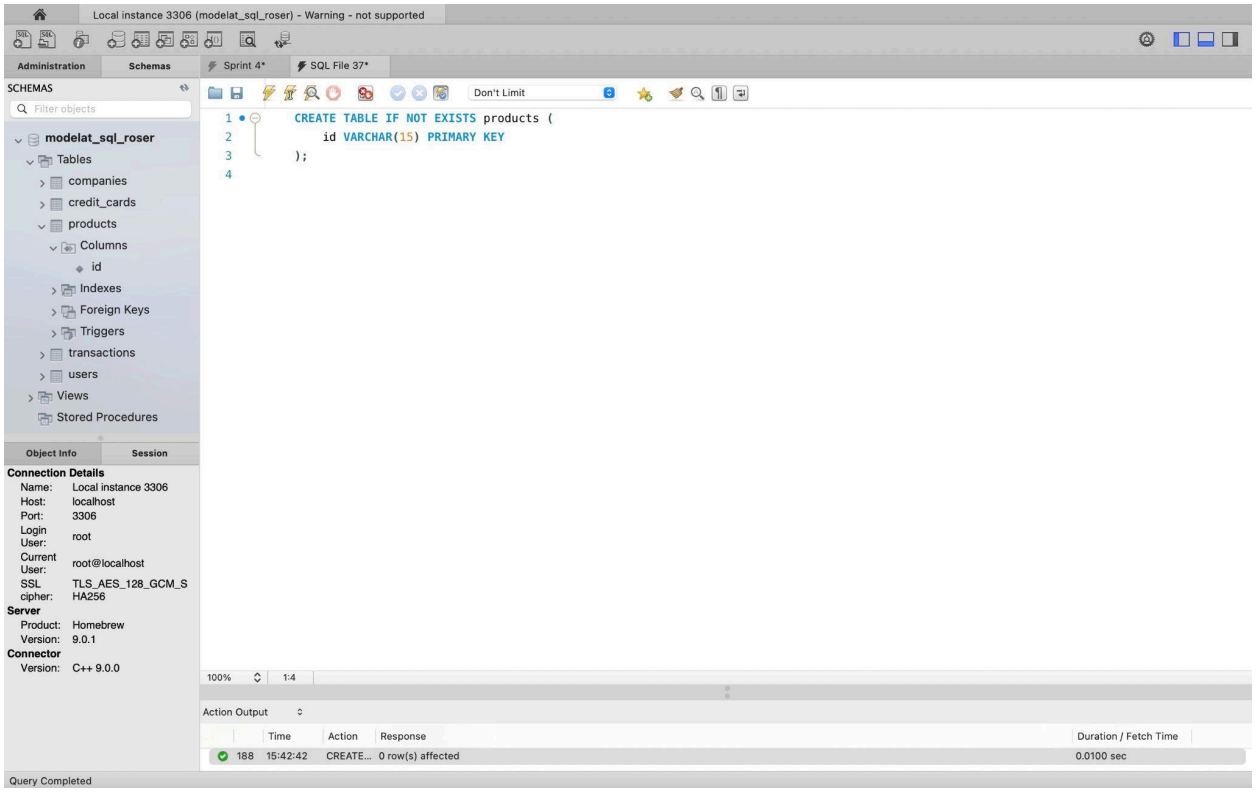
Per a poder unir el nou arxiu, necessitem una taula pont que indexi les relacions entre transaccions i productes.

- Reemplaçar espais

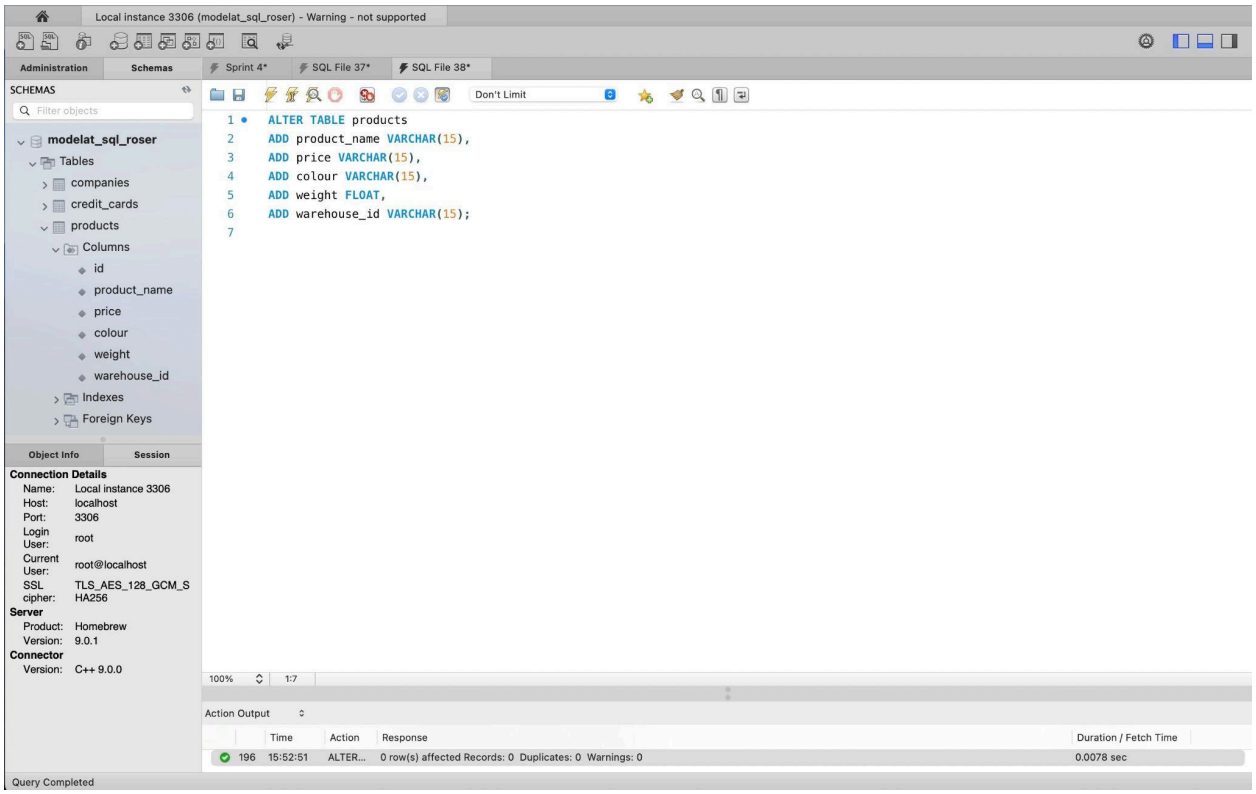


- Carregar taula productes

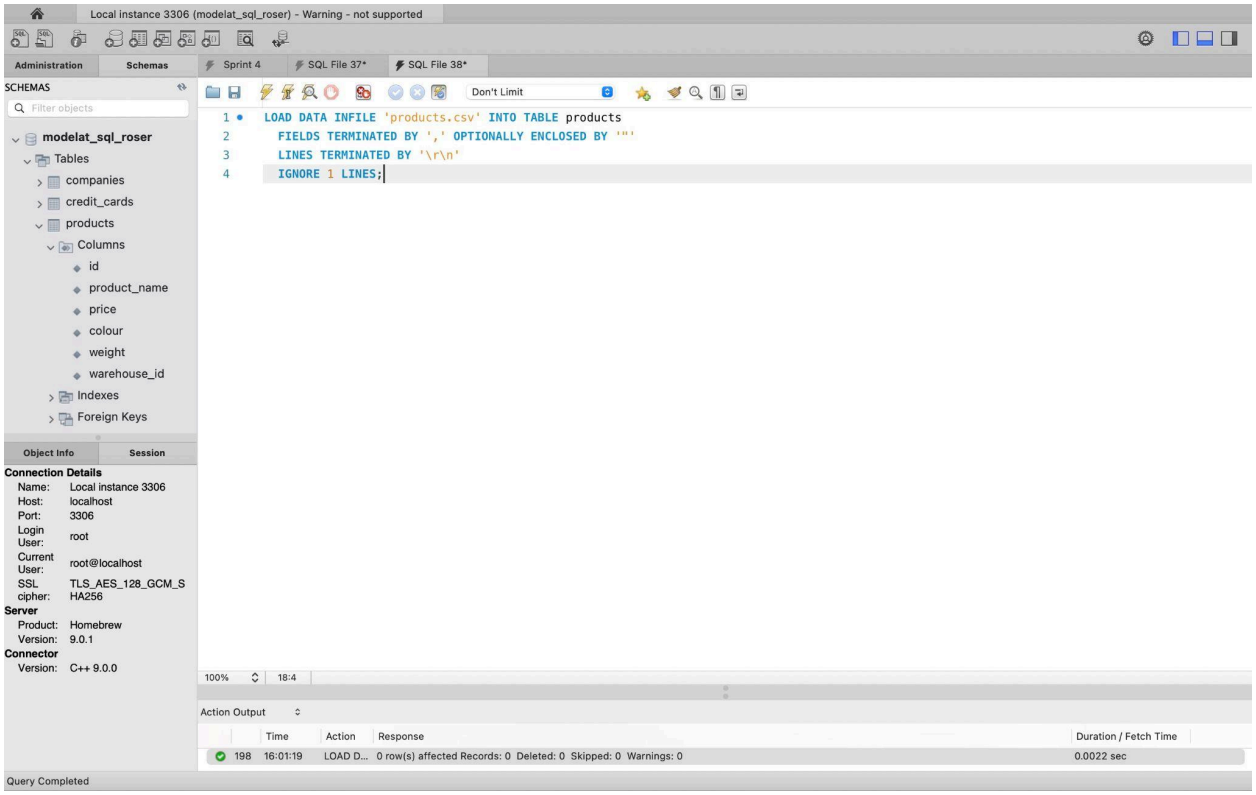
Creem la taula productes



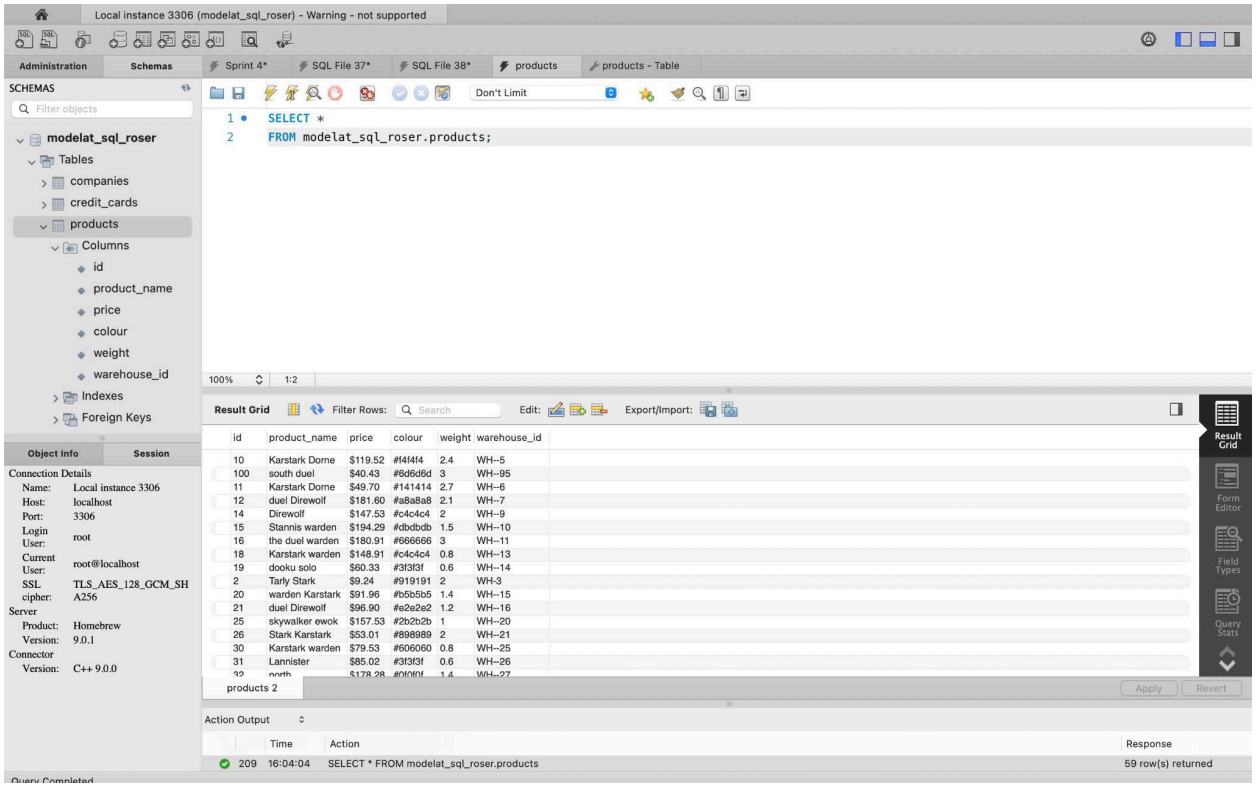
Li afegim les columnes



I insertem les dades

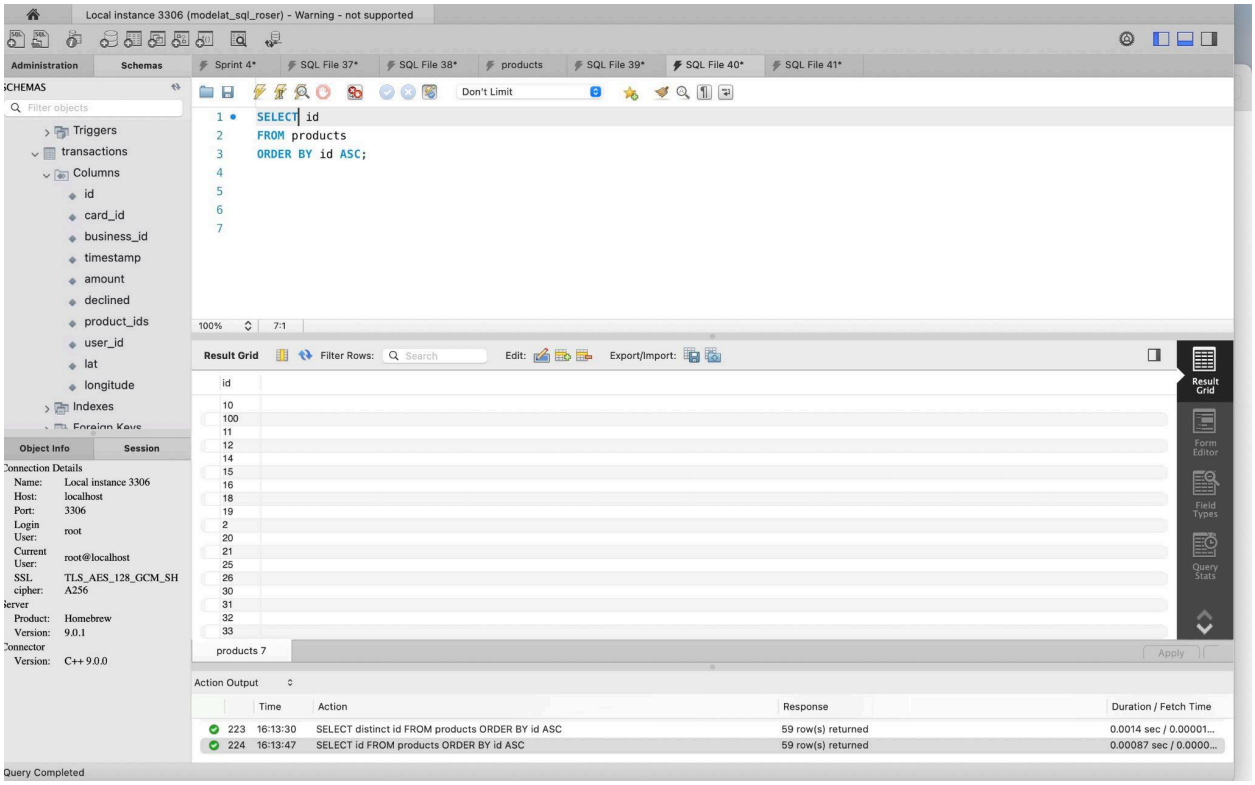


Veiem que estiguin bé

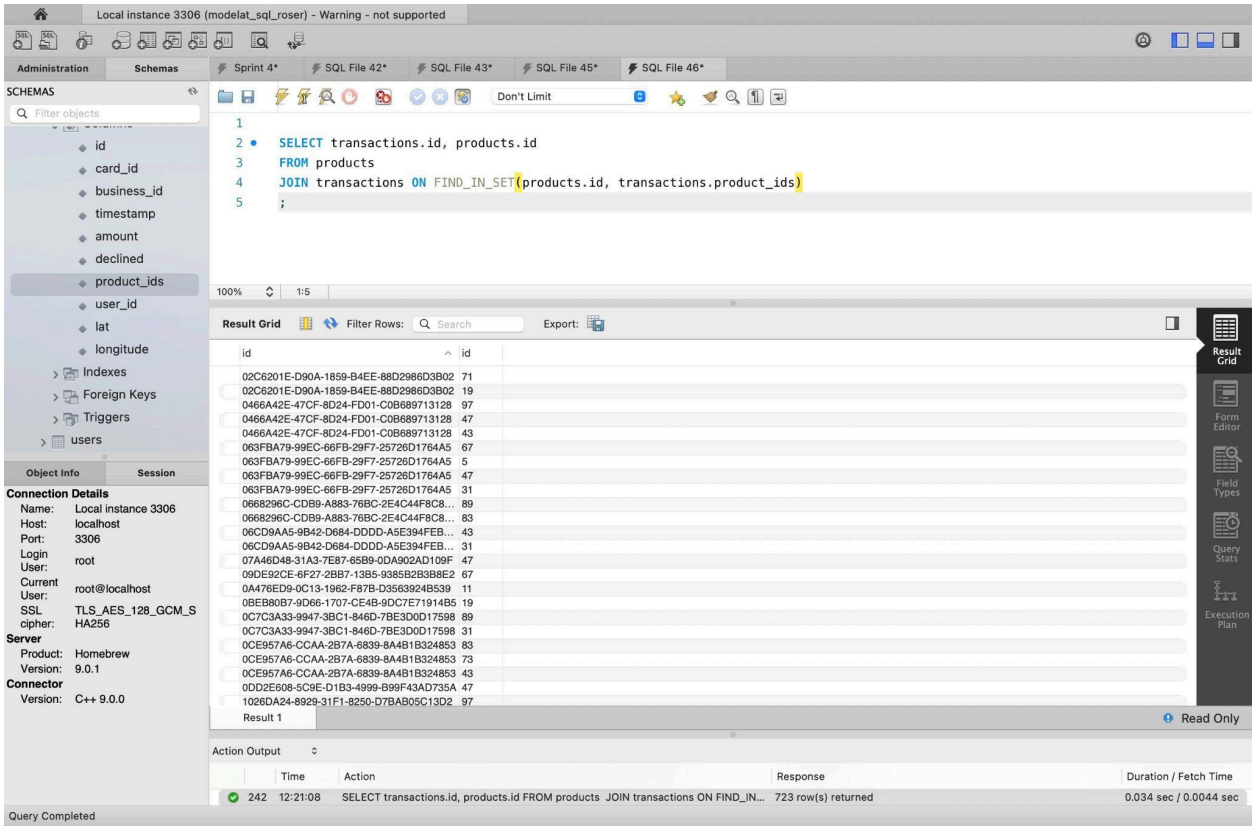


- Llista de productes

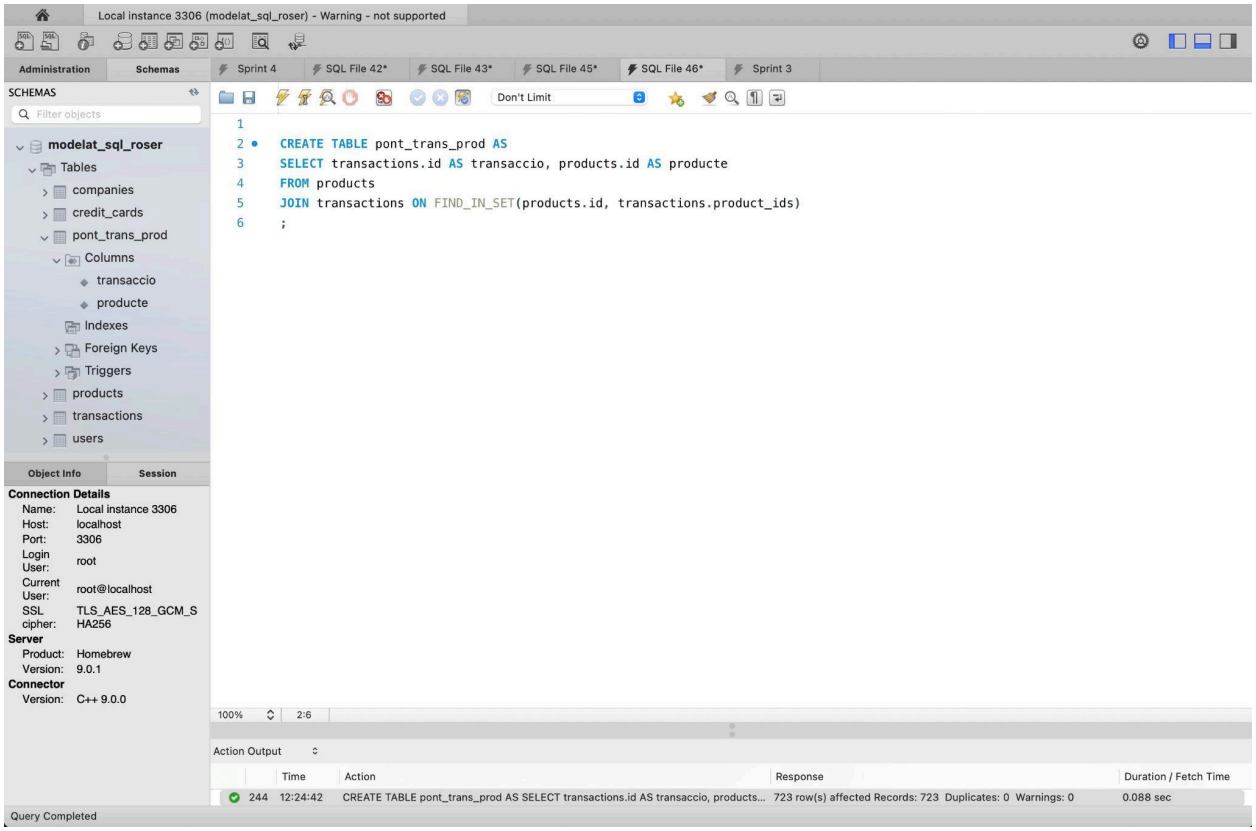
Treiem la llista de productes, hi ha 59 i no estan repetits



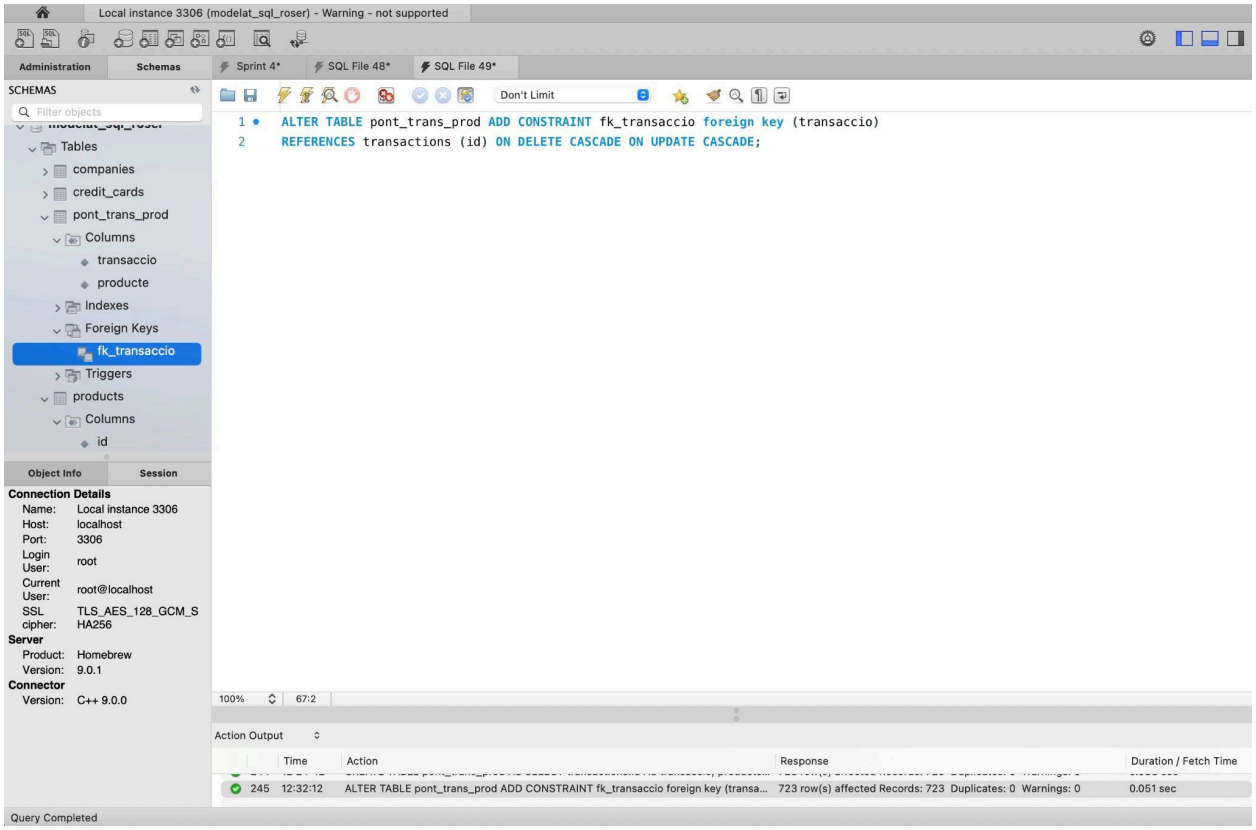
Ara buscarem aquests IDs en la taula de transactions. Utilitzem FIND IN SET Joining Tables Based on Comma-Separated Values https://five.co/blog/mysql-find-in-set/

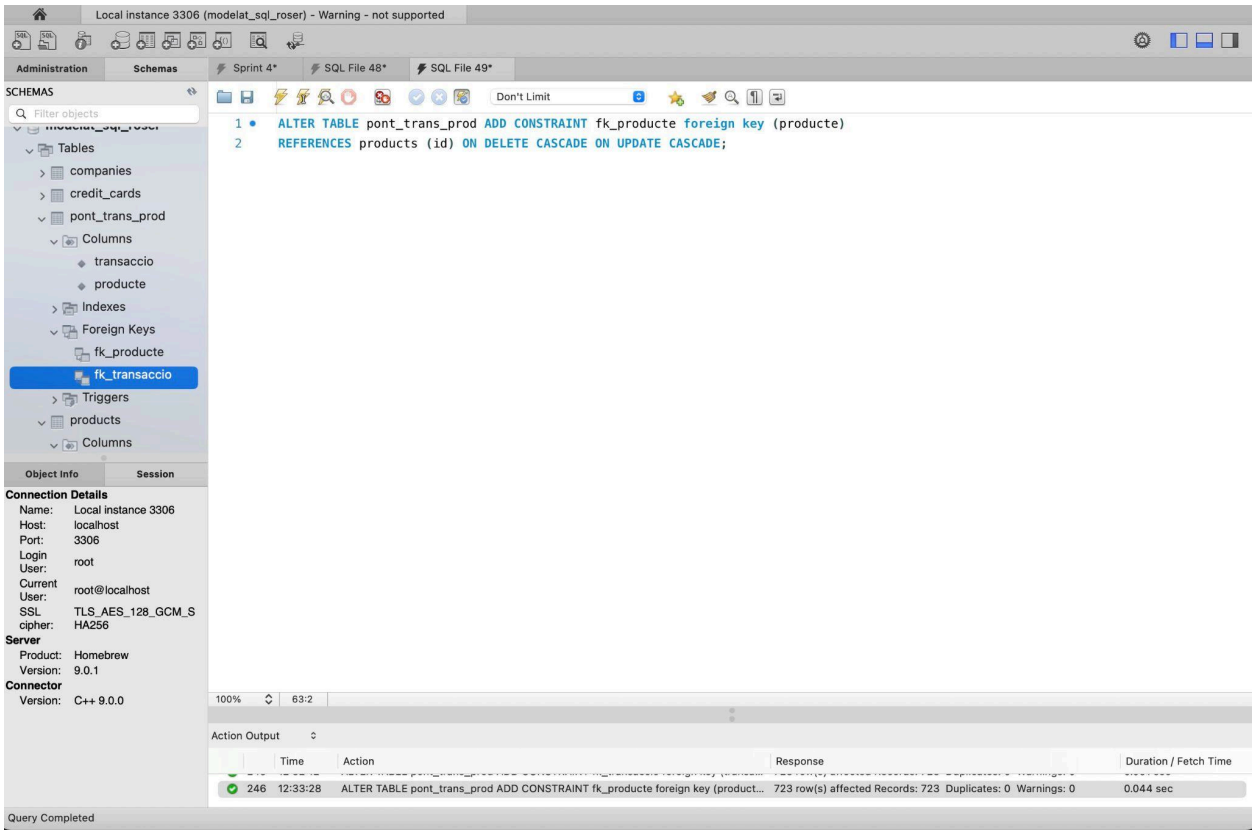


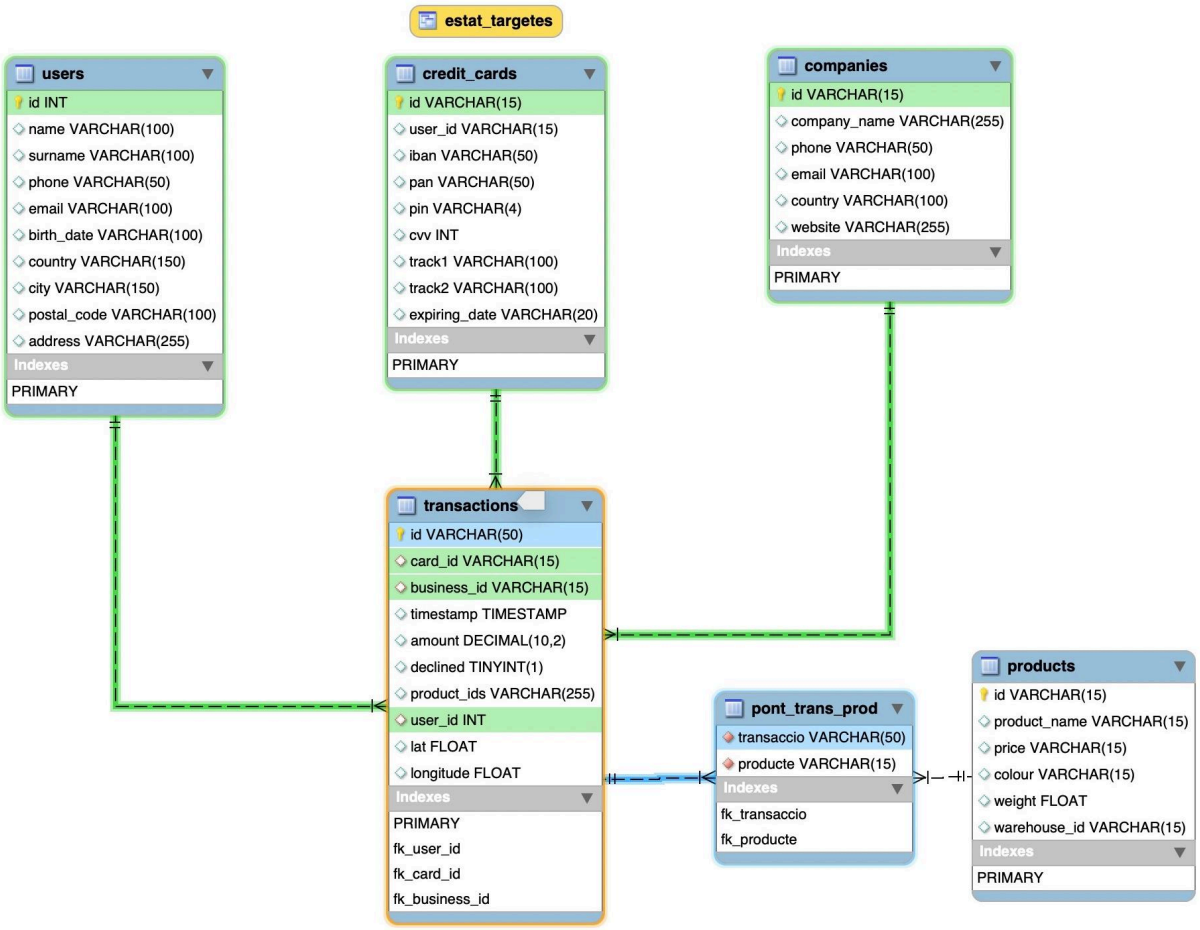
I creem una taula del resultat



Afegim les relacions a les altres taules

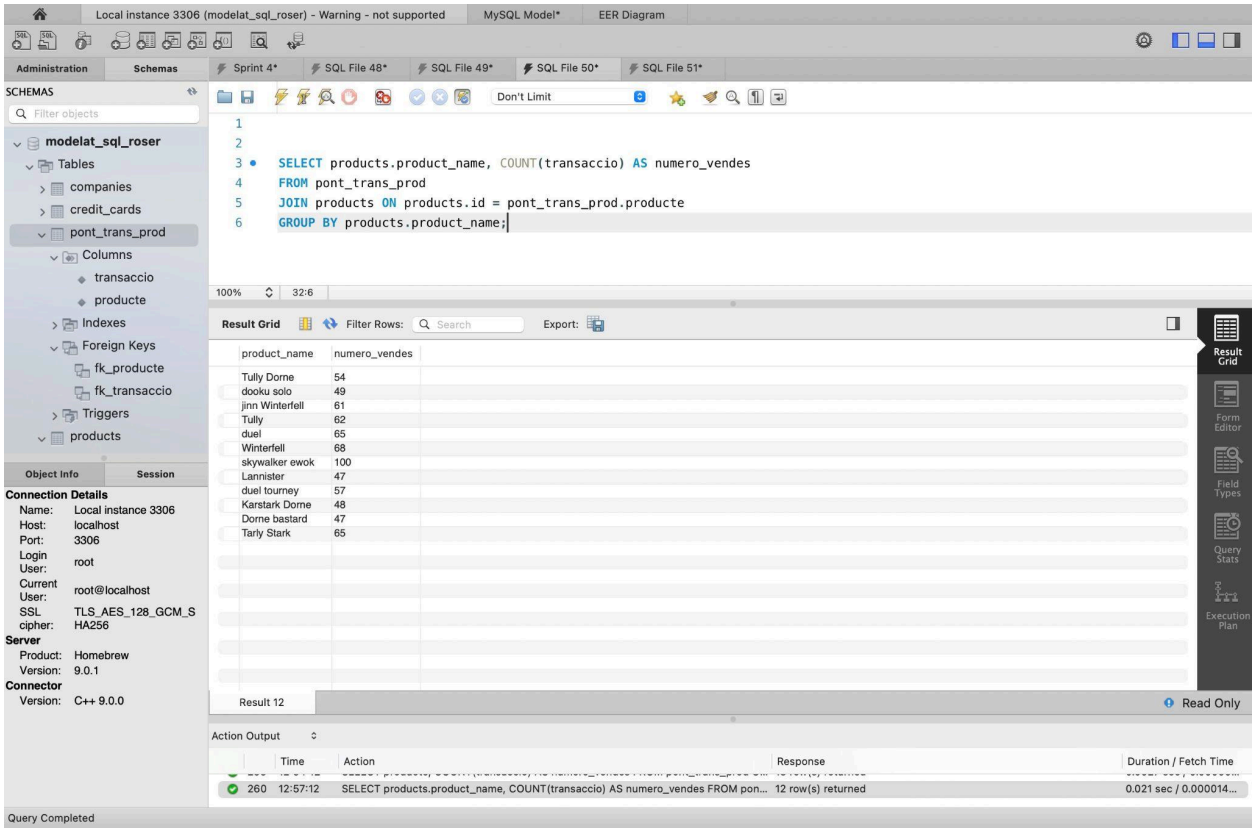

Local instance 3306 (modelat_sql_roser) - Warning - not supported
Administration
Schemas
Sprint 4*
SQL File 48*
SQL File 49*
SCHEMAS
Filter objects
Tables
companies
credit_cards
pont_trans_prod
Columns
transaccio
producte
Indexes
Foreign Keys
fk_transaccio
Triggers
products
Columns
id
Object Info
Session
Connection Details
Name: Local instance 3306
Host: localhost
Port: 3306
Login User: root
Current User: root@localhost
SSL cipher: TLS_AES_128_GCM_SHA256
Server
Product: Homebrew
Version: 9.0.1
Connector
Version: C++ 9.0.0
Don't Limit
1 • ALTER TABLE pont_trans_prod ADD CONSTRAINT fk_transaccio foreign key (transaccio)
2 REFERENCES transactions (id) ON DELETE CASCADE ON UPDATE CASCADE;
100%
67:2
Action Output
Time
Action
Response
Duration / Fetch Time
245 12:32:12 ALTER TABLE pont_trans_prod ADD CONSTRAINT fk_transaccio foreign key (transa... 723 row(s) affected Records: 723 Duplicates: 0 Warnings: 0 0.051 sec
Query Completed


Local instance 3306 (modelat_sql_roser) - Warning - not supported
Administration
Schemas
Sprint 4*
SQL File 48*
SQL File 49*
SCHEMAS
Filter objects
Tables
companies
credit_cards
pont_trans_prod
Columns
transaccio
producte
Indexes
Foreign Keys
fk_producte
fk_transaccio
Triggers
products
Columns
Object Info
Session
Connection Details
Name: Local instance 3306
Host: localhost
Port: 3306
Login User: root
Current User: root@localhost
SSL cipher: TLS_AES_128_GCM_SHA256
Server
Product: Homebrew
Version: 9.0.1
Connector
Version: C++ 9.0.0
Don't Limit
1 • ALTER TABLE pont_trans_prod ADD CONSTRAINT fk_producte foreign key (producte)
2 REFERENCES products (id) ON DELETE CASCADE ON UPDATE CASCADE;
100%
63:2
Action Output
Time
Action
Response
Duration / Fetch Time
246 12:33:28 ALTER TABLE pont_trans_prod ADD CONSTRAINT fk_producte foreign key (product... 723 row(s) affected Records: 723 Duplicates: 0 Warnings: 0 0.044 sec
Query Completed



Exercici 1: vendes per producte

Necessitem conèixer el nombre de vegades que s'ha venut cada producte.


Local instance 3306 (modelat_sql_roser) - Warning - not supported
MySQL Model*
EER Diagram
Administration
Schemas
Sprint 4*
SQL File 48*
SQL File 49*
SQL File 50*
SQL File 51*
SCHEMAS
Filter objects
modelat_sql_roser
Tables
companies
credit_cards
pont_trans_prod
Columns
transaccio
producte
Indexes
Foreign Keys
fk_producte
fk_transaccio
Triggers
products
Object Info
Session
Connection Details
Name: Local instance 3306
Host: localhost
Port: 3306
Login User: root
Current User: root@localhost
SSL cipher: TLS_AES_128_GCM_SHA256
Server
Product: Homebrew
Version: 9.0.1
Connector
Version: C++ 9.0.0
Don't Limit
1
2
3 SELECT products.product_name, COUNT(transaccio) AS numero_vendes
4 FROM pont_trans_prod
5 JOIN products ON products.id = pont_trans_prod.producte
6 GROUP BY products.product_name;
100%
32:6
Result Grid
Filter Rows:
Search
Export:
product_name
numero_vendes
Tully Dorne 54
dooku solo 49
jinn Winterfell 61
Tully 62
duel 65
Winterfell 68
skywalker ewok 100
Lannister 47
duel tourney 57
Karstark Dorne 48
Dorne bastard 47
Tarly Stark 65
Result Grid
Form Editor
Field Types
Query Stats
Execution Plan
Result 12
Read Only
Action Output
Time
Action
Response
Duration / Fetch Time
260 12:57:12 SELECT products.product_name, COUNT(transaccio) AS numero_vendes FROM pon... 12 row(s) returned 0.021 sec / 0.000014...
Query Completed