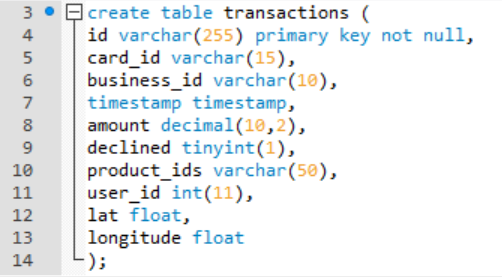
**SPRINT 4:**

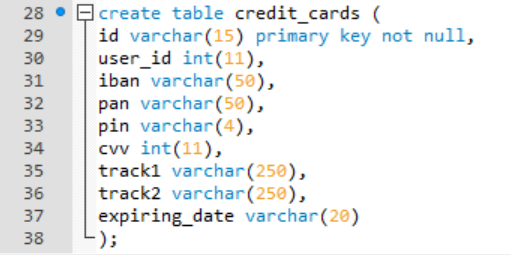
**Nivell 1:**

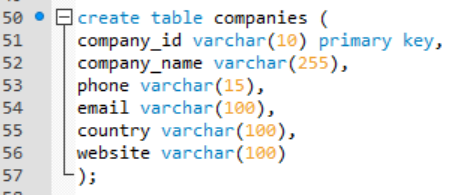
Primer de tot hem creat la base de dades:

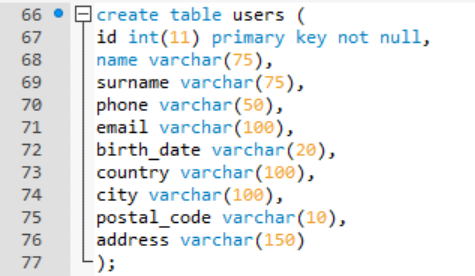


Tot seguit hem creat les taules transaction, companyies, credit\_cards i users.



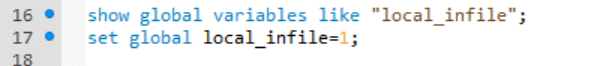


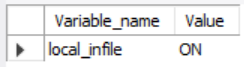




A cada taula ja hem determinat la primary key de cada una i li hem afegit el NOT NULL per determinar que la relació ha de ser obligatòria. A la taula companies no li hem afegit el NOT NULL ja que pot ser que una companyia no hagi fet cap transacció.

Per importar les dades des dels csv en primer lloc hem hagut d’activar el local\_infile de la següent manera:



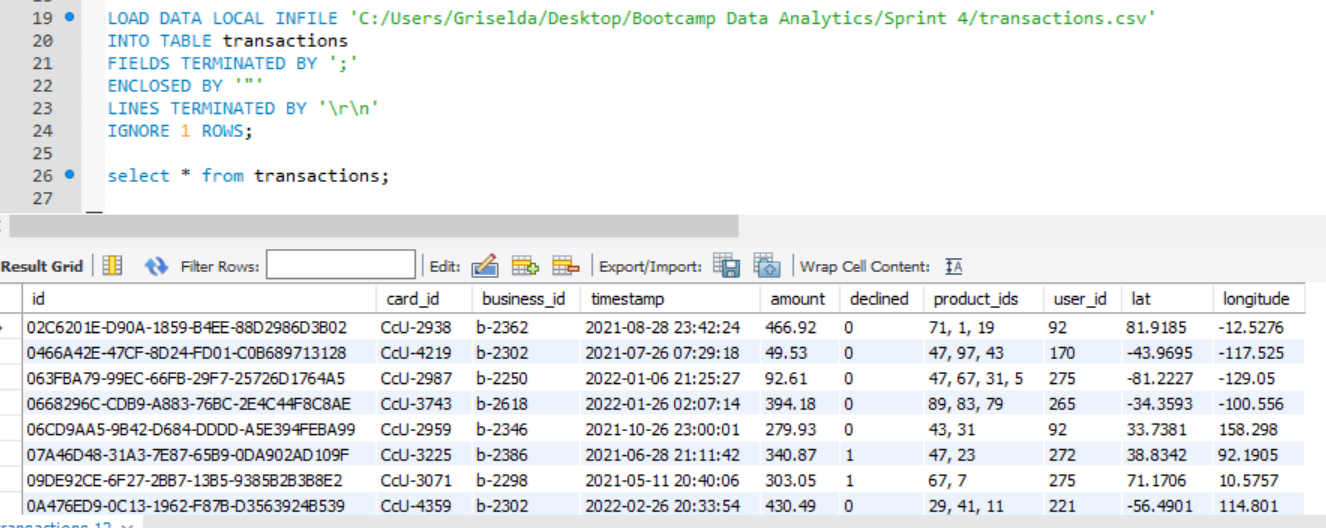


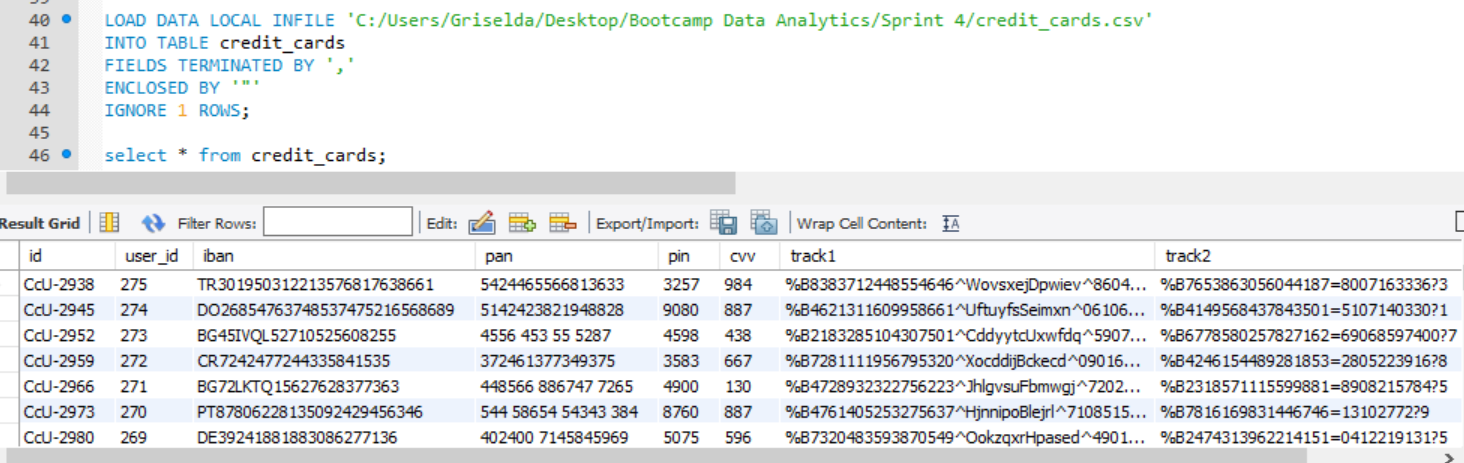
Al continuar sense funcionar hem afegit al fitxer my.ini de mysql el següent:

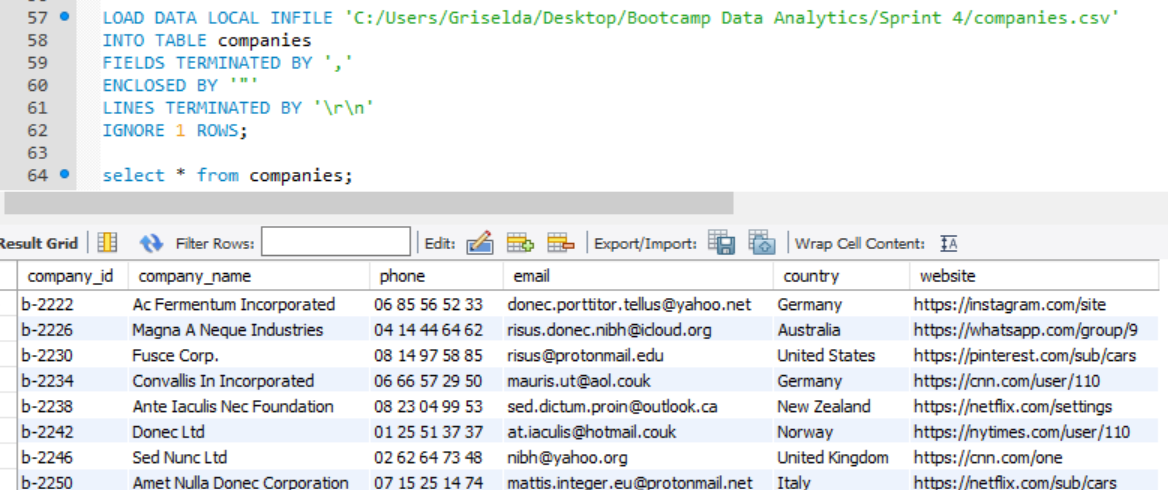
[mysqld]

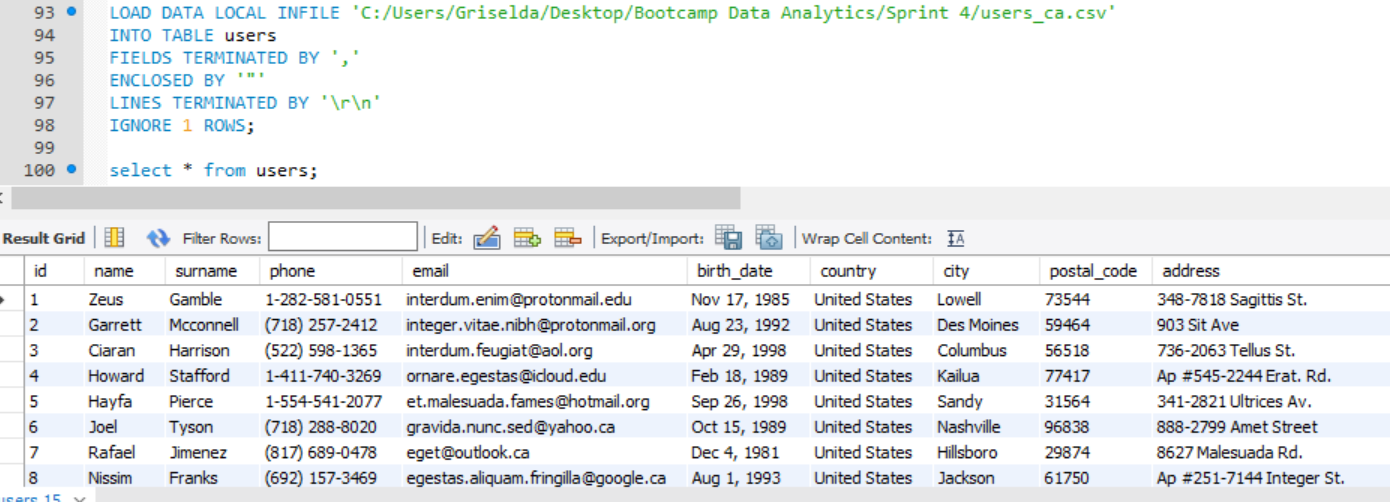
secure-file-priv=""

Un cop fets aquests canvis ja hem pogut importar les dades sense cap problema:



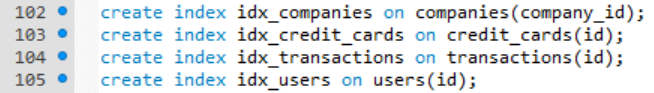


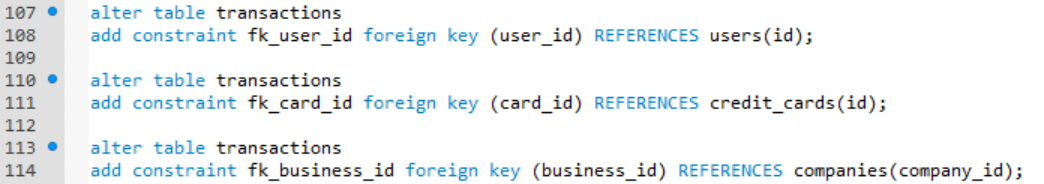




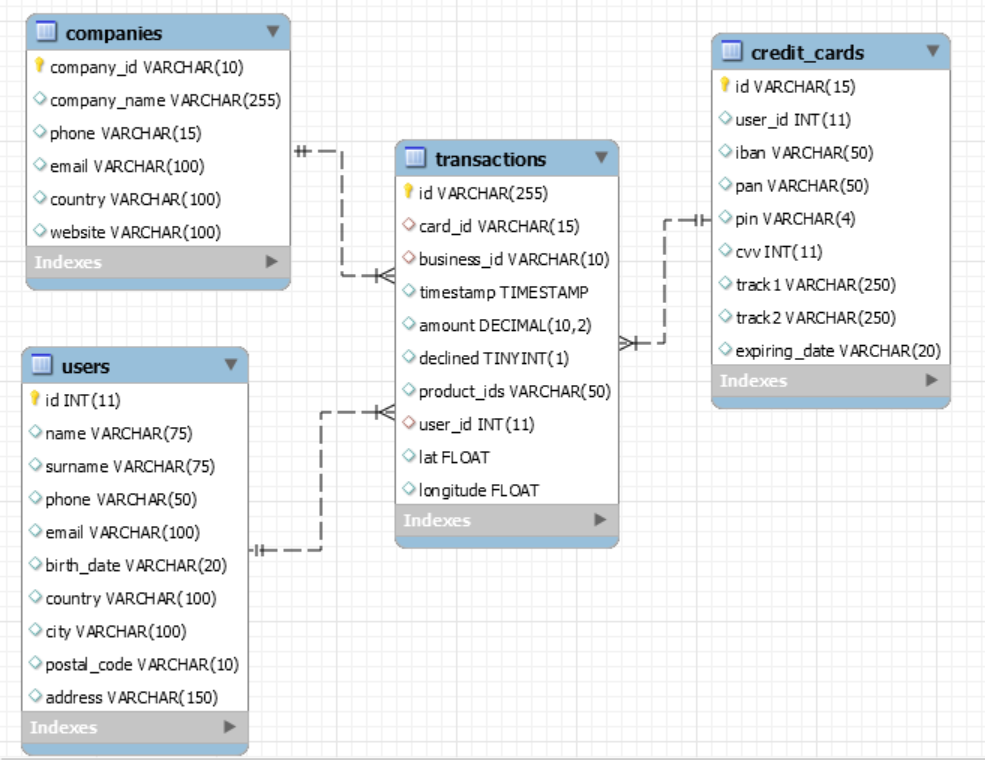
Amb users hem repetit el procés 3 cops per importar els tres fitxers de csv (users\_usa, users\_uk i users\_ca). Només canvia el nom del fitxer en qüestió.

Un cop importats tots els fitxer, hem procedit primer a crear un índex per cada taula per optimitzar el rendiment i després a fer les unions:

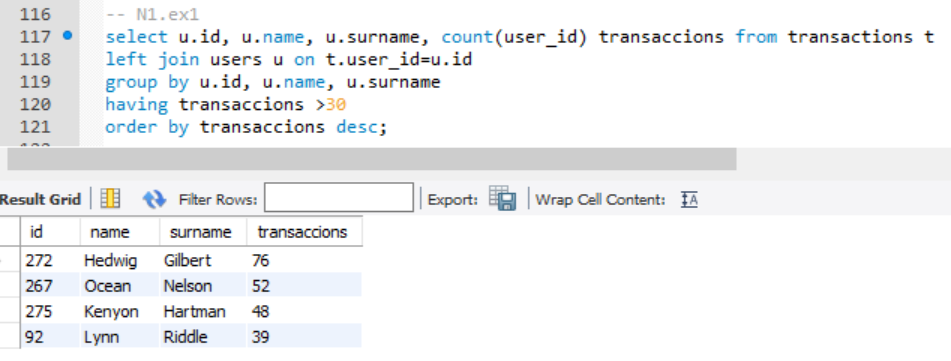




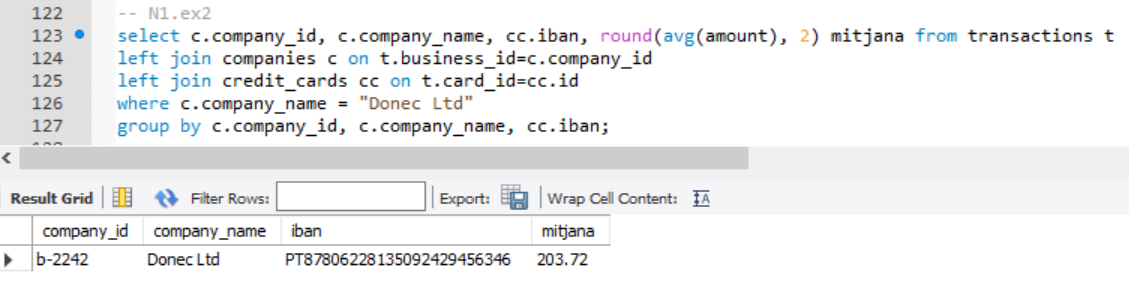
Amb els canvis ens queda el següent diagrama:



* Exercici 1: Realitza una subconsulta que mostri tots els usuaris amb més de 30 transaccions utilitzant almenys 2 taules.



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

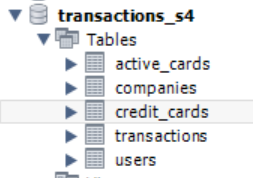


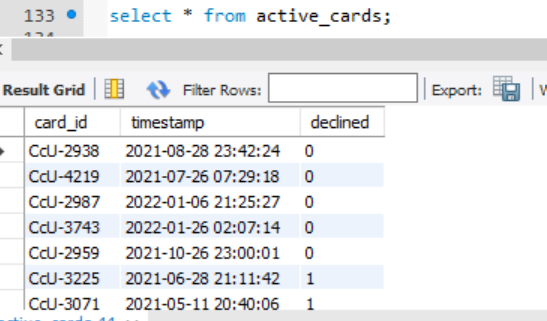
**Nivell 2:**

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:

Per crear la taula independent utilitzem el codi:

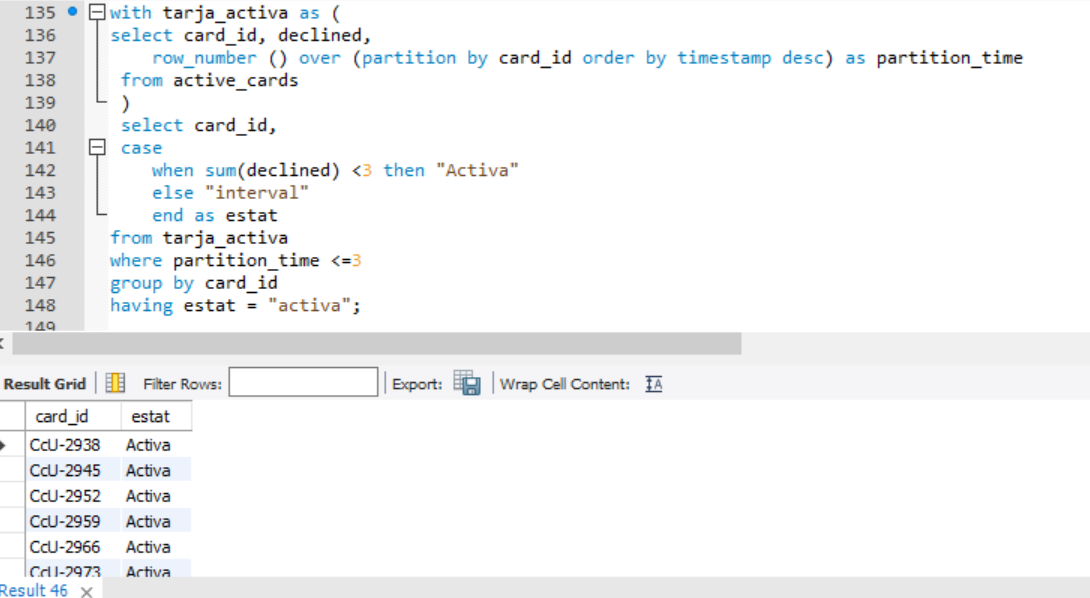






* Exercici 1: Quantes targetes estan actives?

Per saber quantes targetes estan actives amb les condicions de l’enunciat utilitzem el codi:



En primer lloc creem una taula provisional amb el nom de tarja\_activa. Aquesta taula ens mostra les columnes card\_id i declined i amb la funció row\_number creem una nova columna que ens numera cada fila a partir de card\_id, és a dir cada card\_id va de l’1 a n, ordenat pel timestamp desc, perquè les 3 últimes transaccions ens surtin les primeres de cada card\_id.

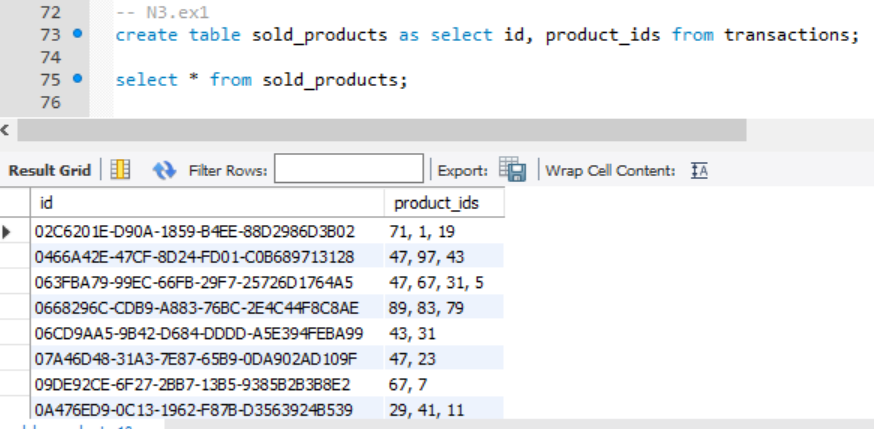
Un cop tenim aquests filtres a través de la funció de case li diem que si la suma de les 3 últimes transaccions de cada card\_id és inferior a 3, ens retorni “activa”, ja que voldrà dir que hi ha com a mínim un moviment acceptat dins les tres últims transaccions. En cas contrari (si la suma és 3 o més) ens retornarà “inactiva”.

Finalment filtrem amb un having by perquè ens mostri només les files de les targetes actives, que son un total de 275, és a dir, totes.

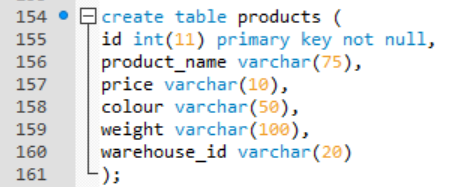
**Nivell 3:**

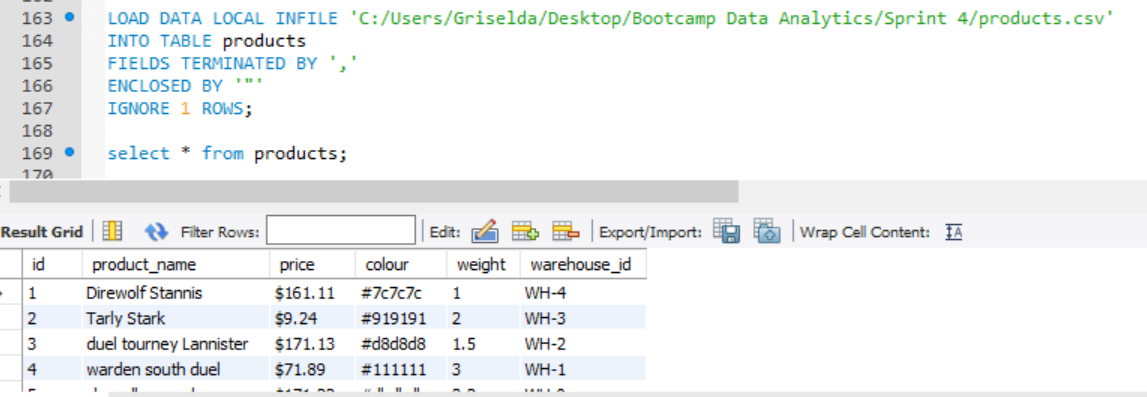
Crea una taula amb la qual puguem unir les dades del nou arxiu products.csv amb la base de dades creada, tenint en compte que des de transaction tens product\_ids. Genera la següent consulta:

Creem una taula amb les columnes id i product\_ids de transaction i creem la taula sold\_products:

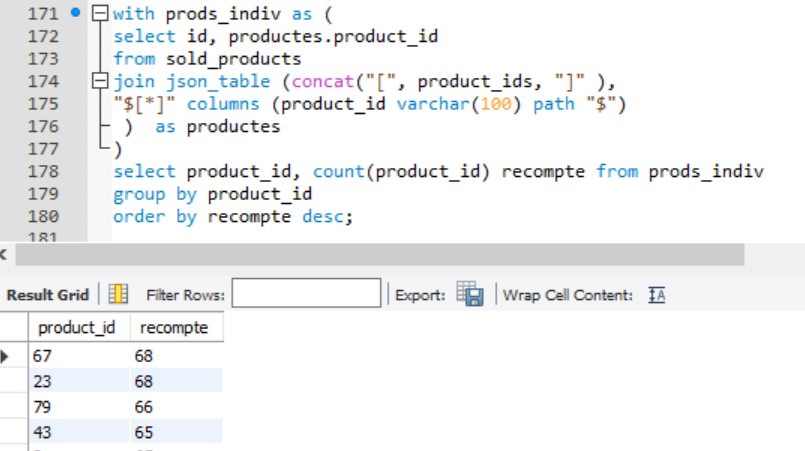


Creem la taula products i pugem les dades:





* Exercici 1: Necessitem conèixer el nombre de vegades que s'ha venut cada producte.



Per poder fer el recompte, primer hem de separar els productes en files diferents. Per fer-ho utilitzem json\_table per convertir les cel·les de la columna product\_ids en format json. Perquè faci la separació correcta, hem d’afegir el $[\*] que fa que separi cada número a través de la funció columns. Un cop ho hem creat, amb la taula auxiliar prods\_indiv ja podem fer el recompte de cada producte.