

Creación de Base de Datos - SQL

Tabla de contenido

| | |
|---|----|
| Nivel 1 | 2 |
| Creo yactivo la base de datos | 2 |
| Creo tabla companies | 2 |
| Intento insertar los datos del csv por código | 3 |
| Inserto datos desde el menú "table data import wizard":..... | 3 |
| Creo la tabla credit_cards, inserto los datos y compruebo | 5 |
| Creo 3 tablas de usuarios: users_usa, users_uk, users_ca | 6 |
| Inserto los datos desde el wizard y compruebo que se hayan cargado | 6 |
| Creo la tabla users, inserto los datos mediante una UNION, agrego columna "tipo" que distinga el origen de datos, y compruebo. | 7 |
| Elimino las tablas auxiliares de usuarios | 7 |
| Creo tabla transactions, agrego las FK que vinculan con otras tablas | 8 |
| Creo índices en las tablas de dimensiones usando los campos clave (PK) de las tablas de dimensiones..... | 9 |
| - Ejercicio 1..... | 10 |
| - Ejercicio 2 | 11 |
| Nivel 2 | 12 |
| - Ejercicio 1..... | 13 |
| Nivel 3 | 14 |
| - Ejercicio 1..... | 17 |

Nivel 1

Descarga los archivos CSV, estudiales y diseña una base de datos con un esquema de estrella que contenga, al menos 4 tablas de las que puedas realizar las siguientes consultas:

Creo yactivo la base de datos

Creo tabla companies

```
1      -- creo y activo la base de datos
2 •  create database institute;
3 •  use institute;
4
5      -- creo una tabla
6 • ⊖create table companies(
7      company_id varchar(10) primary key,
8      company_name varchar(100),
9      phone varchar(20),
10     email varchar(100),
11     country varchar(20),
12     website varchar(100)
13 );
14
```

100% 21:2

Action Output

| | Time | Action | Response |
|-------|----------|----------------------|-------------------|
| ✓ 159 | 11:30:13 | create database i... | 1 row(s) affected |
| ✓ 160 | 11:30:17 | use institute | 0 row(s) affected |
| ✓ 161 | 11:31:27 | create table com... | 0 row(s) affected |

Intento insertar los datos del csv por código de diferentes maneras:

- Intento moviendo el archivo CSV a una ubicación permitida: cambio de ruta el csv, lo pongo donde tengo la base de datos, en la carpeta temp, en la carpeta dónde tengo instalado el sql.
- Intento modificar el archivo de configuración de MySQL (my.cnf o my.ini) y reiniciando el servidor MySQL. No tengo estos archivos, intento crearlos con instrucciones que encuentro en foros y páginas web.

A pesar de todos estos intentos que me llevan más de un día (4hs) el código para importar el csv sigue dando error de privacidad en mi Mac. Entonces, luego de hablarlo con mi tutora, validamos la opción de hacerlo por el wizard del workbench.

```
-- intento insertar los datos del csv por código de diferentes maneras

-- muestra la ruta dónde está el sql
show variables where variable_name = 'datadir';
-- ubico el csv en esta ruta: /usr/local/var/mysql/ -- e intento cargar los datos
20 • LOAD DATA INFILE '/usr/local/var/mysql/companies.csv'
21 INTO TABLE companies
22 FIELDS TERMINATED BY ','
23 ENCLOSED BY '\"'
24 IGNORE 1 LINES;
25 -- me sale un error de privacidad que intenté sortear de distintas maneras durante un día completo y fue imposible:
26 -- Error Code: 1290. The MySQL server is running with the --secure-file-priv option so it cannot execute this statement
27 -- Viendo imposible sortear la privacidad de mi Mac para insertar datos del csv por código,
28

100% ◇ 93:27 |
```

| Action | Output | Duration / Fetch Time |
|--------|---|-----------------------|
| 161 | Time: 11:33:27 Create table companies: 0 row(s) affected | 0.002 sec |
| 162 | Time: 11:33:29 show variables w... 1 row(s) returned | 0.0076 sec / 0.0000 |
| 163 | Time: 11:35:44 LOAD DATA INFI... Error Code: 1290. The MySQL server is running with the --secure-file-priv option so it cannot execute this statement | 0.0019 sec |

Inserto datos desde el menú "table data import wizard":

TASCA SPRINT 4

The screenshot shows the MySQL Workbench interface. In the top navigation bar, the schema is set to 'pruebas csv'. The left sidebar shows the 'SCHEMAS' tree with 'escuela2', 'institute', and 'new_schema' selected. Under 'Tables', 'companies' is selected. The main pane displays the SQL code for creating the 'companies' table:

```

6 • create table companies(
7   company_id varchar(10) primary key,
8   company_name varchar(100),
9   phone varchar(20),
10  email varchar(100),
11  website varchar(100),
12  address varchar(20),
13  city varchar(100)
14 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
15
16 -- intento insertar los datos del csv por código de diferentes maneras
17
18 1 variables where variable_name = 'datadir';
19 2 ibérico el csv en esta ruta: /usr/local/var/mysql/ -- e intento cargar los datos
20
21 3 ! DATA INFILE '/usr/local/var/mysql/companies.csv'
22 4 ! TABLE companies
23 5 ! FIELDS TERMINATED BY ','
24 6 ! ENCLOSED BY '\"'
25 7 ! IGNORE 1 LINES;
26 8 ! -- me sale un error de privacidad que intenté sortear de distintas maneras durante un día completo y fue imposible:
27 9 ! Error Code: 1290. The MySQL server is running with the --secure-file-priv option so it cannot execute this statement
28 10 ! -- Viendo imposible sortear la privacidad de mi Mac para insertar datos del csv por código,
29 11 ! -- inserto datos desde el menú "table data import wizard"
30 12 • select * from companies;
31
32
33

```

The status bar at the bottom indicates '-- incerto datos desde el'.

The 'Action Output' pane shows the following log entries:

| Action | Time | Response | Duration / Fetch Time |
|---------------------|----------|--|-------------------------|
| use institute | 11:30:17 | 0 row(s) affected | 0.0000 sec |
| create table com... | 11:31:27 | 0 row(s) affected | 0.052 sec |
| show variables w... | 11:33:29 | 1 row(s) returned | 0.0076 sec / 0.00001... |
| LOAD DATA INFL... | 11:35:44 | Error Code: 1290. The MySQL server is running with the --secure-file-priv option so it cannot execute this statement | 0.0019 sec |

Observamos los datos insertados

The screenshot shows the MySQL Workbench interface with the schema set to 'pruebas csv'. The left sidebar shows the 'SCHEMAS' tree with 'escuela2', 'institute', and 'new_schema' selected. Under 'Tables', 'companies' is selected. The main pane displays the 'Result Grid' for the 'companies' table:

| company_id | company_name | phone | email | country | website |
|------------|------------------------------|----------------|----------------------------------|----------------|---|
| b-2222 | Ac Fermentum Incorporated | 06 85 56 52 33 | donec.porttitor.tellus@yahoo.net | Germany | https://instagram.com/site |
| b-2226 | Magna A Neque Industries | 04 14 44 64 62 | risus.donec.nibh@icloud.org | Australia | https://whatsapp.com/group/9 |
| b-2230 | Fusce Corp. | 08 14 97 58 85 | risus@protonmail.edu | United States | https://pinterest.com/sub/cars |
| b-2234 | Convallis In Incorporated | 06 66 57 29 50 | mauris.ut@aol.co.uk | Germany | https://cnn.com/user/110 |
| b-2238 | Ante Iaculis Nec Foundation | 08 23 04 99 53 | sed.dictum.pron@outlook.ca | New Zealand | https://netflix.com/settings |
| b-2242 | Donec Ltd | 01 25 51 37 37 | at.iaculis@hotmail.co.uk | Norway | https://nytimes.com/user/110 |
| b-2246 | Sed Nunc Ltd | 02 82 64 73 48 | nibh@yahoo.org | United Kingdom | https://cnn.com/one |
| b-2250 | Amet Nulla Donec Corporation | 07 15 25 14 74 | mattis.integer.eu@protonmail.net | Italy | https://netflix.com/sub/cars |
| b-2254 | Nascetur Ridiculus Mus Inc. | 06 26 87 61 84 | suspendisse.dui@icloud.net | United States | https://ebay.com/sub |
| b-2258 | Vestibulum Lorem PC | 02 02 87 33 40 | aenean.massa.integer@aol.net | Belgium | https://pinterest.com/sub/cars |
| b-2262 | Gravida Sagittis LLP | 03 81 28 33 97 | turpis.vitae@google.ca | Sweden | https://haver.com/site |

The status bar at the bottom indicates 'companies 3'.

The 'Action Output' pane shows the following log entries:

| Action | Time | Response |
|---------------------|----------|---------------------|
| PREPARE SELECT ... | 12:02:10 | OK |
| DEALLOCATE PREPARE | 12:02:16 | OK |
| select * from co... | 12:02:21 | 100 row(s) returned |

Creo la tabla credit_cards, inserto los datos y compruebo

```

33 • create table credit_cards(
34   id varchar(10) primary key,
35   user_id varchar(5),
36   iban varchar(100),
37   pan varchar(50),
38   pin varchar(20),
39   cvv varchar(20),
40   track1 varchar(100),
41   track2 varchar(100),
42   expiring_date varchar(20)
43 );
44 • select * from credit_cards;

```

100% 28:44

| Result Grid | | | | | | | | | |
|--|----------|---------|------------------------------|------------------|------|-----|------------------------------|-------------------|---------------|
| <input type="button" value="Edit"/> <input type="button" value="Print"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/> <input type="button" value="Import"/> Export/Import: <input type="button" value="Print"/> <input type="button" value="Copy"/> | | | | | | | | | |
| | id | user_id | iban | pan | pin | cvv | track1 | track2 | expiring_date |
| ▶ | CcU-2938 | 275 | TR301950312213576817638661 | 5424465566813633 | 3257 | 984 | %B8383712448554646^Wovs... | %B765386305604... | 10/30/22 |
| ▶ | CcU-2945 | 274 | DO26854763748537475216568689 | 5142423821948828 | 9080 | 887 | %B4621311609958661^Uftyf... | %B414956843784... | 08/24/23 |
| ▶ | CcU-2952 | 273 | BG45IVQL52710525608255 | 4556 453 55 5287 | 4598 | 438 | %B2183285104307501^Cddy... | %B677858025782... | 06/29/21 |
| ▶ | CcU-2959 | 272 | CR7242477244335841535 | 372461377349375 | 3583 | 667 | %B7281111956795320^Xocddi... | %B424615448928... | 02/24/23 |
| credit_cards 4 | | | | | | | | | |

Action Output

| | Time | Action | Response |
|---|--------------|---|---------------------|
| ✓ | 172 12:55:42 | create table credit_cards(id varchar(10) primary key, user_id varchar(5), iban varchar(100), pan v... | 0 row(s) affected |
| ✓ | 173 12:56:30 | SHOW SESSION VARIABLES LIKE 'lower_case_table_names' | OK |
| ✓ | 174 12:56:30 | SHOW DATABASES | OK |
| ✓ | 175 12:56:31 | SHOW SESSION VARIABLES LIKE 'lower_case_table_names' | OK |
| ✓ | 176 12:56:31 | SHOW COLUMNS FROM `institute`.`credit_cards` | OK |
| ✓ | 177 12:56:33 | PREPARE stmt FROM 'INSERT INTO `institute`.`credit_cards` ('`id`, `user_id`, `iban`, `pan`, `pin`, `... | OK |
| ✓ | 178 12:56:34 | DEALLOCATE PREPARE stmt | OK |
| ✓ | 179 12:58:49 | select * from credit_cards LIMIT 0, 1000 | 275 row(s) returned |

Creo 3 tablas de usuarios: users_usa, users_uk, users_ca.

```

postal_code varchar(30),
address varchar(50)
);

select * from users_usa;

60
61 • ⏺ create table users_uk(
62     id int primary key,
63     name varchar(30),
64     surname varchar(30),
65     phone varchar(30),
66     email varchar(50),
67     birth_date varchar(30),
68     country varchar(30),
69     city varchar(30),
70     postal_code varchar(30),
71     address varchar(50)
72 );
73
74 •   select * from users_uk;
75
76 • ⏺ create table users_ca(
77     id int primary key,
78     name varchar(30),

```

Action Output

| | Time | Action | Response |
|-------|----------|---|----------|
| ✓ 180 | 13:06:08 | create table users_usa(id int primary key, name varchar(30), surname varchar(30), phone varchar... 0 row(s) affected | |
| ✓ 181 | 13:06:23 | create table users_uk(id int primary key, name varchar(30), surname varchar(30), phone varchar(... 0 row(s) affected | |
| ✓ 182 | 13:06:27 | create table users_ca(id int primary key, name varchar(30), surname varchar(30), phone varchar... 0 row(s) affected | |

Inserto los datos desde el wizard y compruebo que se hayan cargado

Para poder importarlo tuve que editar el csv Qébec por Qebec sin acento.

Administration Schemas

SCHEMAS

- > escuela2
- < institute
 - Tables
 - > companies
 - > credit_cards
 - > **users_ca**
 - > users_uk
 - > users_usa
 - Views
 - Stored Procedures
 - Functions
 - new schema

Object Info Session

Table: **users_ca**

Columns:

| id | int PK |
|---------|-------------|
| name | varchar(30) |
| surname | varchar(30) |

Table Data Import

Import Results

File /Users/rags/Desktop/formaciones/Data Analytics IT Academy/scripts/sprint4/datos/users_ca.csv was imported in 0.425 s

Table institute.users_ca has been used

75 records imported

TASCA SPRINT 4

Creo la tabla users, inserto los datos mediante una UNION, agrego columna "tipo" que distinga el origen de datos, y compruebo.

The screenshot shows the MySQL Workbench interface. At the top, there's a toolbar with various icons. Below it is a code editor window containing SQL code. The code creates a table 'users' with columns: birth_date (varchar(30)), country (varchar(30)), city (varchar(30)), postal_code (varchar(30)), address (varchar(50)), and tipo (varchar(20)). It then inserts data from three tables: 'users_usa', 'users_uk', and 'users_ca' into the 'users' table, adding a 'tipo' column that distinguishes the origin of each row. Finally, it selects all data from the 'users' table.

```

99 birth_date varchar(30),
100 country varchar(30),
101 city varchar(30),
102 postal_code varchar(30),
103 address varchar(50),
104 tipo varchar(20)
105 );
106
107 -- inserto datos en users mediante una Union, agrego columna "tipo" que distinga el origen de datos
108 • insert into users
109 select *, 'users_usa' as tipo from users_usa
110 union
111 select *, 'users_uk' from users_uk
112 union
113 select *, 'users_ca' from users_ca;
114
115 • select * from users;
116
117
118
119
120
121
  
```

Result Grid:

| id | name | surname | phone | email | birth_date | country | city | postal_code | address | tipo |
|----|---------|-----------|----------------|-----------------------------------|--------------|---------------|------------|-------------|-----------------------|-----------|
| 1 | Zeus | Gamble | 1-282-581-0551 | interdum.enim@protonmail.edu | Nov 17, 1985 | United States | Lowell | 73544 | 348-7818 Sagittis St. | users_usa |
| 2 | Garrett | Mcconnell | (718) 257-2412 | integer.vitae.nibh@protonmail.org | Aug 23, 1992 | United States | Des Moines | 59464 | 903 Sit Ave | users_usa |
| 3 | Ciaran | Harrison | (522) 598-1365 | interdum.feugiat@aol.org | Apr 29, 1998 | United States | Columbus | 56518 | 736-2063 Tellus St. | users_usa |

Action Output:

| Time | Action | Response | Duration |
|--------------|--|--|----------|
| 209 13:50:33 | create table users (id int primary key, name varchar(30), surname varchar(30), phone varchar(30), email varchar(50), birth_date varchar(30), country varchar(30), city varchar(30), postal_code varchar(30), address varchar(50), tipo varchar(20)); | 0 row(s) affected | 0.051 s |
| 210 13:51:02 | insert into users select *, 'users_usa' as tipo from users_usa union select *, 'users_uk' from users_uk union select *, 'users_ca' from users_ca; | 275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0 | 0.054 s |
| 211 13:51:06 | select * from users LIMIT 0, 1000 | 275 row(s) returned | 0.0016 s |

Elimino las tablas auxiliares de usuarios

```

117 • drop table users_usa;
118 • drop table users_uk;
119 • drop table users_ca;
120
121
  
```

The screenshot shows the MySQL Workbench interface. It displays the execution of three 'drop table' statements: 'users_usa', 'users_uk', and 'users_ca'. The 'Action Output' section shows the results of these operations.

Action Output:

| Time | Action |
|--------------|----------------------|
| 213 13:58:26 | drop table users_usa |
| 214 13:58:29 | drop table users_uk |
| 215 13:58:31 | drop table users_ca |

TASCA SPRINT 4

Creo tabla transactions, agrego las FK que vinculan con otras tablas

```

122  create table transactions(
123      id varchar(100) primary key,
124      card_id varchar(10),
125      business_id varchar(10),
126      fecha_hora timestamp,
127      amount decimal(20,2),
128      declined int,
129      product_ids varchar(100),
130      user_id int,
131      lat varchar(10),
132      longitude varchar(10)
133  );
134 • alter table transactions
135     add foreign key(card_id) references credit_cards(id),
136     add foreign key(business_id) references companies(company_id),
137     add foreign key(user_id) references users(id);
138
  
```

Result Grid

| Field | Type | Null | Key | Default | Extra |
|-------------|--------------|------|-----|---------|-------|
| id | varchar(100) | NO | PRI | NULL | |
| card_id | varchar(10) | YES | MUL | NULL | |
| business_id | varchar(10) | YES | MUL | NULL | |
| fecha_hora | timestamp | YES | | NULL | |

Action Output

| Time | Action | Response |
|--------------|--|----------------------------|
| 217 20:54:59 | create table transactions(id varchar(100) primary key, card_id varchar(10), business_id varchar(10), fecha_hora timestamp, amount decimal(20,2), declined int, product_ids varchar(100), user_id int, lat varchar(10), longitude varchar(10)); | 0 row(s) affected |
| 218 20:55:04 | alter table transactions add foreign key(card_id) references credit_cards(id), add foreign key(business_id) references companies(company_id), add foreign key(user_id) references users(id); | 0 row(s) affected Response |
| 219 20:55:22 | desc transactions | 10 row(s) returned |

Inserto los datos del csv de transactios y consulto a modo de chequeo

```

138
139 • desc transactions;
140 • select * from transactions;
141
  
```

Result Grid

| id | card_id | business_id | fecha_hora | amount | declined | product_ids | user_id | lat | longitude |
|--------------------------------------|----------|-------------|---------------------|--------|----------|---------------|---------|----------------|-----------------|
| 02C6201E-D90A-1859-B4EE-88D2986D3B02 | CcU-2938 | b-2362 | 2021-08-28 23:42:24 | 466.92 | 0 | 71, 1, 19 | 92 | 81.9184589824 | -12.5275561984 |
| 0466A42E-47CF-8D24-FD01-C0B689713128 | CcU-4219 | b-2302 | 2021-07-26 07:29:18 | 49.53 | 0 | 47, 97, 43 | 170 | -43.9694885888 | -117.5251835904 |
| 063FBA79-99EC-66FB-29F7-25726D176A5 | CcU-2987 | b-2250 | 2022-01-06 21:25:27 | 92.61 | 0 | 47, 67, 31, 5 | 275 | -81.222680576 | -129.049879552 |
| 066829AC-CD90-A883-76BC-2EAC44E8C8AE | CcU-3743 | b-2618 | 2022-01-26 02:07:14 | 394.18 | 0 | 89, 82, 79 | 265 | -34.3593055232 | -100.555928064 |

Action Output

| Time | Action | Response | Dur |
|--------------|--|---------------------|------|
| 227 20:59:31 | DEALLOCATE PREPARE stmt | OK | 0.00 |
| 228 21:03:06 | select * from transactions LIMIT 0, 1000 | 587 row(s) returned | 0.00 |

SCHEMAS

Import Results

File /Users/rags/Desktop/formaciones/Data Analytics I'

Table institute.transactions has been used

587 records imported

Creo índices en las tablas de dimensiones usando los campos clave (PK) de las tablas de dimensiones

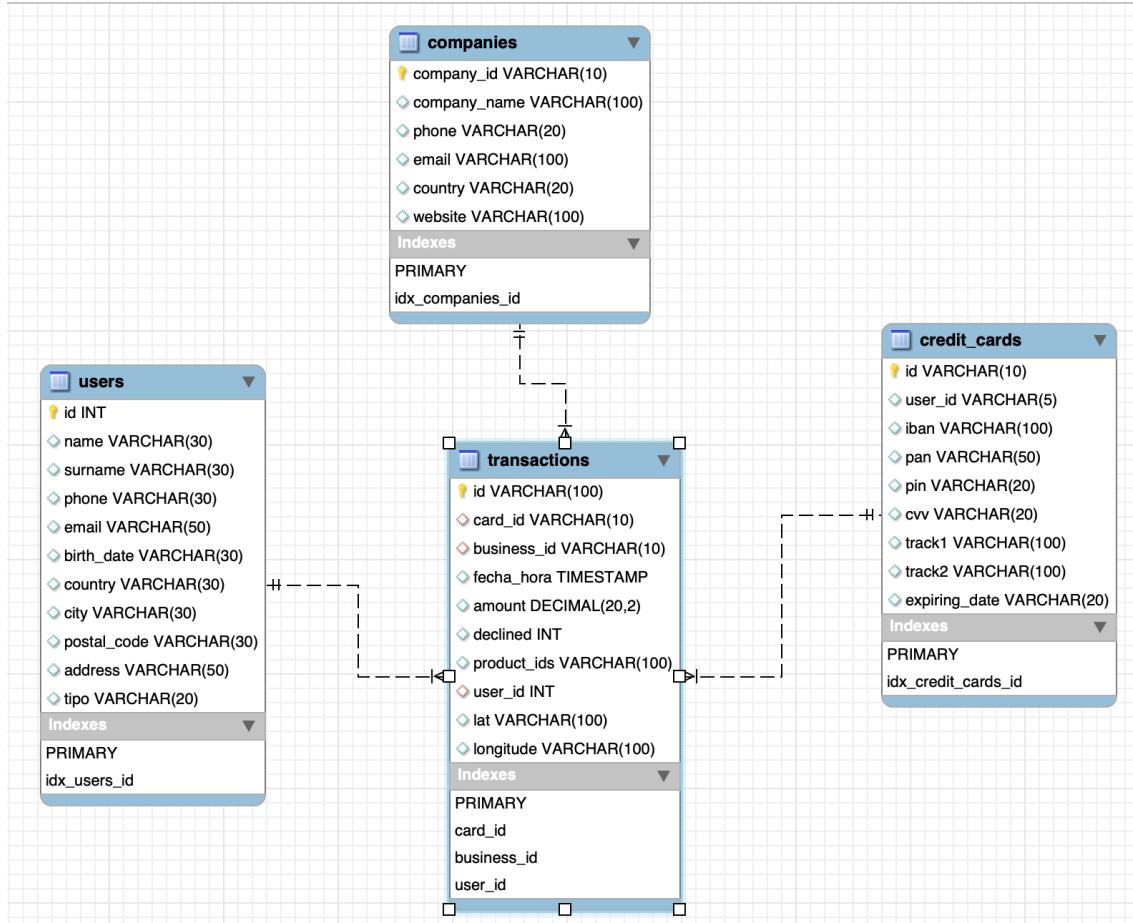
```

142      -- Índice para la tabla companies
143 •  CREATE INDEX idx_companies_id ON companies(company_id);
144
145      -- Índice para la tabla users
146 •  CREATE INDEX idx_users_id ON users(id);
147
148      -- Índice para la tabla credit_cards
149 •  CREATE INDEX idx_credit_cards_id ON credit_cards(id);
150

```

| Time | Action |
|--------------|--|
| 229 21:06:08 | CREATE INDEX idx_companies_id ON companies(company_id) |
| 230 21:06:13 | CREATE INDEX idx_users_id ON users(id) |
| 231 21:06:16 | CREATE INDEX idx_credit_cards_id ON credit_cards(id) |

Mostramos el esquema de estrella que diseñamos que contiene 4 tablas, con transactions como tabla de hechos.



- Ejercicio 1

Realiza una subconsulta que muestre a todos los usuarios con más de 30 transacciones utilizando al menos 2 tablas.

Para resolver esta consulta hacemos una query de todos aquellos usuarios de la tabla transacciones que tengan más de 30 repeticiones en esa tabla, es decir usuarios con más de 30 transacciones.

Luego para obtener el nombre de usuario y los demás datos de estos usuarios hago una consulta sobre la tabla users que filtre aquellos usuarios que obtuve el id explicados en el primer párrafo (quedará como una subquery).

Por último, agrego a los datos de usuario la cantidad de transacciones para que se pueda verificar que todos superan la cantidad de 30 que pide el enunciado. También los ordenamos descendenteamente por cantidad de transacciones.

Aclaramos que para resolver este planteo consideramos TODAS las transacciones, estén o no declinadas.

```

152  -- nl ex1
153 •  select *, (select count(user_id) from transactions where transactions.user_id = users.id) as q_transactions
154   from users
155   where id in(
156     select user_id
157     from transactions
158     group by user_id
159     having count(user_id) > 30
160     order by count(user_id) desc
161   )
162   order by q_transactions desc;
163
100% 13:158 |
```

Result Grid Filter Rows: Search Export:

| | id | name | surname | phone | email | birth_date | country | city | postal_code | address | tipo | q_transaksi |
|---|-----|--------|---------|----------------|---------------------------------|--------------|---------------|---------------|-------------|--------------------------------|-----------|-------------|
| ▶ | 272 | Hedwig | Gilbert | 064-204-8788 | sem.eget@icloud.edu | Apr 16, 1991 | Canada | Tuktoyaktuk | Q4C 3G7 | P.O. Box 496, 5145 Sapien Road | users_ca | 76 |
| | 267 | Ocean | Nelson | 079-481-2745 | aenean@yahoo.com | Dec 26, 1991 | Canada | Charlottetown | 85X 3P4 | Ap #732-8357 Pede, Rd. | users_ca | 52 |
| | 275 | Kenyon | Hartman | 082-871-7248 | convallis.ante.lectus@yahoo.com | Aug 3, 1982 | Canada | Richmond | R8H 2K2 | 8564 Facilisi, St. | users_ca | 48 |
| | 92 | Lynn | Riddle | 1-387-885-4057 | vitae.aliquet@outlook.edu | Sep 21, 1984 | United States | Bozeman | 61871 | P.O. Box 712, 7907 Est St. | users_usa | 39 |

Result 13

Action Output

| Time | Action | Response | Duration / Fetch |
|--------------|--|-------------------|------------------|
| 237 11:35:52 | select *, (select count(user_id) from transactions where transactions.user_id = users.id) as q_transaksi | 4 row(s) returned | 0.0073 sec / 0 |

- Ejercicio 2

Muestra la media de amount por IBAN de las tarjetas de crédito en la compañía Donec Ltd., utiliza por lo menos 2 tablas.

Para resolver este pedido uno la tabla transactions con companies y credit_cards. (joins). Filtro la compañía que deseo y agrupo por iban de tarjetas de crédito y compañía. Finalmente armo en el select el promedio de importes de las transacciones, agrupado por iban de las tarjetas.

Al igual que para el ejercicio 1, consideramos TODAS las transacciones, tanto las declinadas como las no declinadas. En este caso lo hacemos debido a que sólo tiene 2 transacciones y una está declinada. Preferimos dejarla para mostrar el funcionamiento del cálculo del promedio.

```
--  
173 •   select  
174     c.company_name,  
175     cr.iban,  
176     round(avg(t.amount),2)  
177   from transactions t  
178   inner join companies c on t.business_id = c.company_id  
179   inner join credit_cards cr on cr.id = t.card_id  
180   where c.company_name = 'Donec Ltd'  
181   group by c.company_name, cr.iban;  
100% ◊ 14:245 | 1 error found  
  
Result Grid Filter Rows: Search Export:  


| company_name | iban                      | round(avg(t.amount)... |
|--------------|---------------------------|------------------------|
| Donec Ltd    | PT87806228135092429456346 | 203.72                 |

  
Result 15  
  
Action Output Action Response  
262 19:37:35 select c.company_name, cr.iban, round(avg(t.amount),2) from transactions t inner join companie... 1 row(s) returned
```

Nivel 2

Crea una nueva tabla que refleje el estado de las tarjetas de crédito basado en si las últimas tres transacciones fueron declinadas y genera la siguiente consulta:

Intentamos dividir el problema en partes, y luego ir anidando las consultas.

Primero, numeramos las transacciones para cada tarjeta desde la más reciente hasta la más antigua, y creamos una nueva columna con esa numeración.

Segundo, filtramos las tres transacciones más recientes para cada tarjeta de crédito

Tercero, agrupamos cada tarjeta por el estado, en una nueva columna estado, según la condición de si la suma de las últimas 3 transacciones para cada tarjeta de crédito es menor a 3 activo, y si suma 3 será que las últimas 3 transacciones han sido declinadas.

Cuarto, creamos la tablas status, la estructura y luego la completamos con la subquery que habíamos generado.

```
190      -- paso 4: Creo la tabla de status
191  ●  create table creditCardStatus (
192      card_id varchar(10) primary key,
193      estado varchar(20)
194      -- foreign key (card_id) references credit_cards(id) -- aquí no me lo permite
195  ) as
196      -- Paso 3: Agrupo cada tarjeta por el estado, en una nueva columna estado
197      select
198          card_id,
199          -- opción con "if"
200          if(sum(declined) < 3, 'activa', 'declinada') as estado
201          -- opción con "case"
202          -- case
203          -- when sum(declined) = 0 then 'activa'
204          -- else 'declinada'
205          -- end as estado
206      from (
207          -- Paso 2: Filtrar las tres transacciones más recientes para cada tarjeta de crédito
208          select *
209          from (
210              -- Paso 1) numeramos las transacciones para cada tarjeta desde la mas reciente hasta la mas antigua, y
211              -- creamos una nueva columna con esa numeración
212              select
213                  card_id,
214                  declined,
215                  row_number() over(partition by card_id order by fecha_hora) as row_num
216                  from transactions
217              ) as cronologic_transactions
218              -- hasta aquí 1)
219              where row_num <= 3
220          ) as last_3
221          -- hasta aquí 2)
222          group by card_id;
223          -- hasta aquí 3)
224          -- hasta aquí 4)
```

'5% 103:188 | 1 error found

Action Output

| | Time | Action | Response |
|-----|----------|---------------------------------------|--|
| 335 | 11:42:55 | create table creditCardStatus (ca... | 275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0 |

TASCA SPRINT 4

Quinto, agregamos la clave foránea e índice para vincular tablas

```
!31 • alter table creditCardStatus
!32   add foreign key (card_id) references credit_cards(id);
!33
!34 • CREATE UNIQUE INDEX idx_card_id_unique ON creditCardStatus(card_id);
```

| Action Output | | | |
|---------------|--------------|--|--|
| | Time | Action | Response |
| ● | 336 11:43:07 | alter table creditCardStatus add fo... | 275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0 |
| ● | 337 11:43:59 | CREATE UNIQUE INDEX idx_card_i... | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0 |

- Ejercicio 1

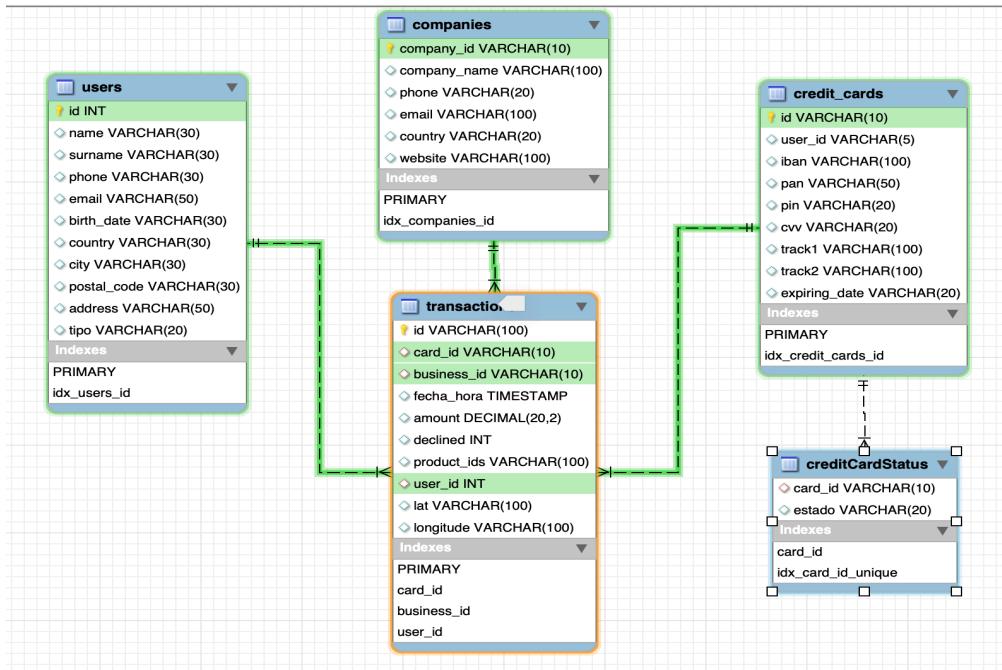
¿Cuántas tarjetas están activas?

Sexto, consultamos la nueva tabla y verificamos que hay 275 tarjetas declinadas con los criterios establecidos (las últimas 3 transacciones declinadas).

```
238 -- paso 6) consulto las tarjetas caducadas
239 • select estado, count(card_id)
240   from CreditCardStatus
241   where estado = 'activa';
242
```

| Result Grid | | Filter Rows: | Search | Export: |
|-------------------------------------|--------------|---------------------------------------|-------------------|---------|
| <code>estado count(card_i...</code> | | | | |
| ▶ | activa | 275 | | |
| Result 23 | | | | |
| Action Output | | | | |
| ● | 340 12:30:28 | select estado, count(card_id) from... | 1 row(s) returned | |

Observamos cómo queda el esquema con la incorporación de la nueva tabla



Nivel 3

Crea una tabla con la que podamos unir los datos del nuevo archivo products.csv con la base de datos creada, teniendo en cuenta que desde transaction tienes product_ids. Genera la siguiente consulta:

Creamos la tabla products, estructura, inserción de datos y verificamos

```

393 • create table products(
394     id varchar(20) primary key,
395     product_name varchar(100),
396     price varchar(30),
397     colour varchar(20),
398     weight varchar(10),
399     warehouse_id varchar(20)
400 );
401
402 • select * from products;
```

100% 15:402

Result Grid

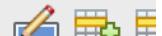


Filter Rows:



Search

Edit:



| | id | product_name | price | colour | weight | warehouse_id | |
|---|-----|------------------|----------|---------|--------|--------------|--|
| ▶ | 1 | Direwolf Stannis | \$161.11 | #7c7c7c | 1 | WH-4 | |
| ◀ | 10 | Karstark Dorne | \$119.52 | #f4f4f4 | 2.4 | WH-5 | |
| ◀ | 100 | south duel | \$40.43 | #6d6d6d | 3 | WH-95 | |
| ◀ | 11 | Karstarks Dorne | \$40.70 | #111111 | 0.7 | WH-6 | |

products 90

Action Output



| | Time | Action | Response |
|-------|----------|----------------|---------------------|
| ✓ 389 | 14:37:20 | create ta... | 0 row(s) affected |
| ✓ 390 | 14:37:41 | SHOW SE... | OK |
| ✓ 391 | 14:37:41 | SHOW DA... | OK |
| ✓ 392 | 14:37:42 | SHOW SE... | OK |
| ✓ 393 | 14:37:42 | SHOW C... | OK |
| ✓ 394 | 14:37:43 | PREPARE... | OK |
| ✓ 395 | 14:37:44 | DEALLOC... | OK |
| ✓ 396 | 14:38:14 | select * fr... | 100 row(s) returned |

Dado que la relación entre products y transactions es de muchos a muchos, debemos crear una tabla intermedia:

Creamos la tabla intermedia entre transacciones y productos con ambos campos como PK y FK

Luego llenamos la tabla insertando los id de transaction y los id de product, filtrando por las transacciones efectivamente realizadas.

TASCA SPRINT 4

Previamente debemos utilizar la función `find_in_set` para enlazar las tablas ya que en transactions existen varios id products separados por comas (',')

```
408 • ⏷ create table trans_products (
409   id_transaction varchar(100),
410   id_product varchar(20),
411   primary key (id_transaction, id_product),
412   foreign key (id_transaction) references transactions(id),
413   foreign key (id_product) references products(id));
414
415 • ⏷ insert into trans_products (
416   id_transaction, id_product)
417   select t.id, p.id
418   from transactions t
419   inner join products p on find_in_set(p.id, replace(t.product_ids, ',', ',', ',')) 
420   where declined = 0;
421
```

100% 1:407

| Action Output | Time | Action | Response |
|---------------|----------|---------------|--|
| 397 | 14:59:39 | create ta... | 0 row(s) affected |
| 398 | 14:59:48 | insert int... | 1236 row(s) affected Records: 1236 Duplicates: 0 Warnings: 0 |

- Ejercicio 1

Necesitamos conocer el número de veces que se ha vendido cada producto.

Vinculamos la nueva tabla para obtener el nombre del producto y hacemos un count agrupado por productos.

```
428 • select p.id, p.product_name, count(id_transaction) as q_sold
429   from trans_products t
430   inner join products p on t.id_product = p.id
431   group by p.id, p.product_name
432   order by q_sold desc;
```

100% 1:427

Result Grid Filter Rows: Search Export:

| | id | product_name | q_sold |
|---|----|------------------|--------|
| ▶ | 23 | riverlands north | 60 |
| ▶ | 67 | Winterfell | 59 |
| ▶ | 2 | Tarly Stark | 56 |

Result 96

Action Output

| | Time | Action | Response |
|---|--------------|---------------|--------------------|
| ✓ | 404 15:34:17 | select p.i... | 26 row(s) returned |

Vemos el diagrama con las nuevas tablas y sus relaciones

