

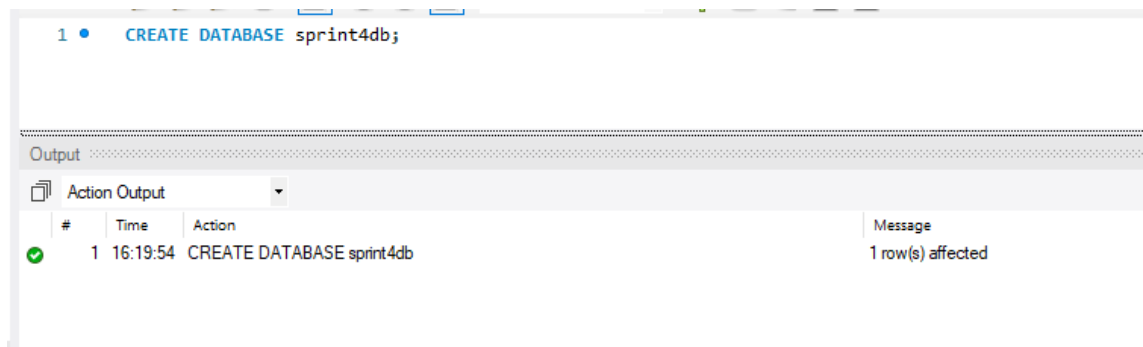
Tarea S4.01. Creación de bases de datos

A partir de algunos archivos CSV, diseñarás y crearás tu base de datos.

NIVEL 1

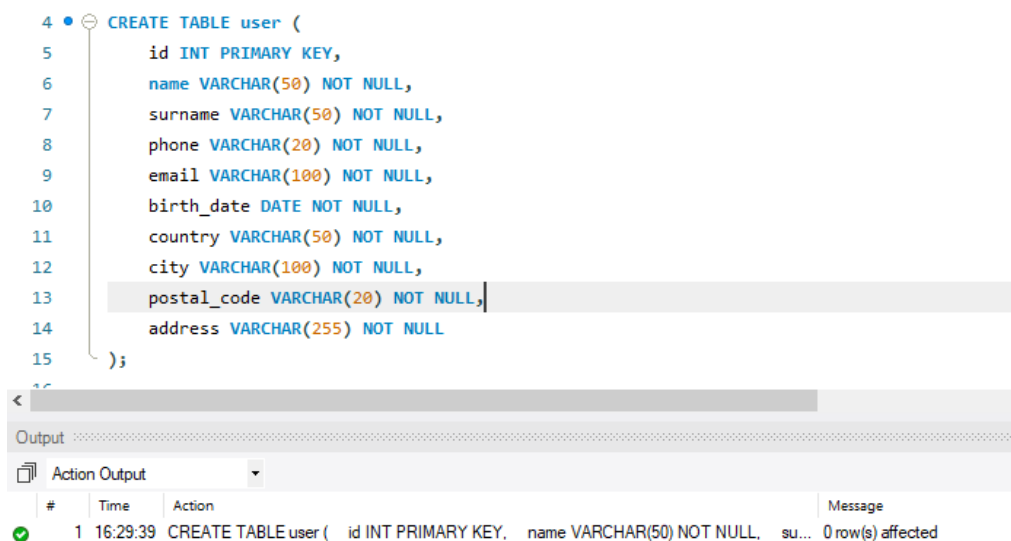
Descarga los archivos CSV, estúdialos y diseña una base de datos con un esquema en estrella que contenga al menos 4 tablas, de las cuales puedas realizar las siguientes consultas:

Creo la base de datos llamada *sprint4db*



A continuación creo la primera tabla, user, creo esta la primera ya que es la más independiente. Esta tabla contendrá los csv de users_uk, users_ca y users_usa ya que los tres csv's tienen los mismos tipos de datos y las mismas columnas.

Contiene un campo country, que diferencia el país de cada registro (cada fichero es de un país). Indicamos como PK el campo id y todos los campos como Not Null.



Tarea S4.01. Creación de bases de datos

Cargo los datos de la tabla user con los datos de los csv's de users_uk, users_ca y users_usa con el comando LOAD DATA INFILE.

Los csvs los he tenido que almacenar en la carpeta uploads del servidor de mysql

Hay que hacer modificación del tipo de dato de birth_date ya que el formato en el que aparece no es DATE y queremos transformarlo por ello utilizamos STR_TO_DATE. El modificador %b indica el nombre del mes abreviado, por ejemplo Enero (Jan),

En cada fichero hay que observar si los saltos de línea son \r\n (CRLF) o \n (LF) dependiendo desde el sistema que nos hayan creado el csv, etc...

```
17 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\users_ca.csv'
18 INTO TABLE user
19 FIELDS TERMINATED BY ','
20 OPTIONALLY ENCLOSED BY ''
21 LINES TERMINATED BY '\\r\\n'
22 IGNORE 1 ROWS
23 (id, name, surname, phone, email, @birth_date_raw, country, city, postal_code, address)
24 SET
25     birth_date = STR_TO_DATE(@birth_date_raw, '%b %d, %Y')
26
27
```

Output

#	Time	Action	Message
1	17:05:00	LOAD DATA INFILE C:\\ProgramData\\MySQL\\MySQL...	75 row(s) affected Records: 75 Deleted: 0 Skipped: 0 Warnings: 0

```
27 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\users_uk.csv'
28 INTO TABLE user
29 FIELDS TERMINATED BY ','
30 OPTIONALLY ENCLOSED BY ''
31 LINES TERMINATED BY '\\r\\n'
32 IGNORE 1 ROWS
33 (id, name, surname, phone, email, @birth_date_raw, country, city, postal_code, address)
34 SET
35     birth_date = STR_TO_DATE(@birth_date_raw, '%b %d, %Y');
36
37
```

Output

#	Time	Action	Message
1	17:06:29	LOAD DATA INFILE C:\\ProgramData\\MySQL\\MySQL...	50 row(s) affected Records: 50 Deleted: 0 Skipped: 0 Warnings: 0

```
37
38 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\users_usa.csv'
39 INTO TABLE user
40 FIELDS TERMINATED BY ','
41 OPTIONALLY ENCLOSED BY ''
42 LINES TERMINATED BY '\\r\\n'
43 IGNORE 1 ROWS
44 (id, name, surname, phone, email, @birth_date_raw, country, city, postal_code, address)
45 SET
46     birth_date = STR_TO_DATE(@birth_date_raw, '%b %d, %Y');
47
48
```

Output

#	Time	Action	Message
1	17:06:51	LOAD DATA INFILE C:\\ProgramData\\MySQL\\MySQL...	150 row(s) affected Records: 150 Deleted: 0 Skipped: 0 Warnings: 0

Tarea S4.01. Creación de bases de datos

Creo la tabla company indicando como Pk el campo id y todos los campos Not Null de tipo Varchar.

```
48 CREATE TABLE company (  
49     id VARCHAR(20) PRIMARY KEY,  
50     name VARCHAR(255) NOT NULL,  
51     phone VARCHAR(50) NOT NULL,  
52     email VARCHAR(255) NOT NULL,  
53     country VARCHAR(100) NOT NULL,  
54     website VARCHAR(255) NOT NULL  
55 );
```

Output

Action Output

#	Time	Action	Message
1	17:09:32	CREATE TABLE company (id VARCHAR(20) PRIMA...	0 row(s) affected

Cargo los datos de la tabla company:

```
56  
57 LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\companies.csv'  
58 INTO TABLE company  
59 FIELDS TERMINATED BY ','  
60 OPTIONALLY ENCLOSED BY ''''  
61 LINES TERMINATED BY '\\r\\n'  
62 IGNORE 1 ROWS  
63 (id, name, phone, email, country, website);
```

Output

Action Output

#	Time	Action	Message
1	17:12:01	LOAD DATA INFILE C:\\ProgramData\\MySQL\\MyS...	100 row(s) affected Records: 100 Deleted: 0 Skipped: 0 Warnings: 0

Creo la tabla product indicando como PK el campo id y todos los campos Not Null.

```
65 CREATE TABLE product (  
66     id INT PRIMARY KEY,  
67     name VARCHAR(255) NOT NULL,  
68     price VARCHAR(20) NOT NULL,  
69     colour VARCHAR(7) NOT NULL,  
70     weight DECIMAL(5, 2) NOT NULL,  
71     warehouse_id VARCHAR(10) NOT NULL  
72 );
```

Output

Action Output

#	Time	Action	Message
1	17:35:28	CREATE TABLE product (id INT PRIMARY KEY, ...	0 row(s) affected

Tarea S4.01. Creación de bases de datos

Cargo los datos:

```
74 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\products.csv'
75 INTO TABLE product
76 FIELDS TERMINATED BY ','
77 OPTIONALLY ENCLOSED BY '"'
78 LINES TERMINATED BY '\n'
79 IGNORE 1 ROWS
80 (id, name, price, colour, weight, warehouse_id);
81
82
83
```

Output

#	Time	Action	Message
1	17:35:57	LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL...	100 row(s) affected Records: 100 Deleted: 0 Skipped: 0 Warnings: 0

Deberíamos hacer un cambio en el campo de precio ya que no puede salir el símbolo del \$ porque es Varchar y necesitamos que sea un dato numérico para poder realizar operaciones en un futuro.

Pero tenemos activada la restricción de seguridad que es para evitar modificaciones o eliminaciones por error sin una clausula WHERE.

Por ello deberíamos desactivar esa restricción de seguridad con SET_SQL_SAFE_UPDATES = 0, luego realizar los cambios que hemos nombrado anteriormente y luego volver a activar las restricciones de seguridad con SET_SQL_SAFE_UPDATES = 1. A continuación se realizaría otra modificación en la columna precio indicando que es decimal con 10 posiciones para los enteros y 2 para los decimales. Pero nada de esto vamos a hacer porque no nos hace falta ahora mismo este cambio y pondríamos en riesgo la integridad de la tabla.

Ahora creo la tabla credit_card indicando como PK la columna id, todos los campos NOT NULL.

```
84 • CREATE TABLE credit_card (
85     id VARCHAR(20) PRIMARY KEY,
86     user_id INT NOT NULL,
87     iban VARCHAR(34) NOT NULL,
88     pan VARCHAR(20) NOT NULL,
89     pin INT NOT NULL,
90     cvv INT NOT NULL,
91     track1 VARCHAR(255) NOT NULL,
92     track2 VARCHAR(255) NOT NULL,
93     expiring_date DATE NOT NULL
94 );
95
```

Output

#	Time	Action	Message
1	19:18:43	CREATE TABLE credit_card (id VARCHAR(20) PRIMARY KEY, user_id INT NOT NU...	0 row(s) affected

Tarea S4.01. Creación de bases de datos

Ahora cargo los datos haciendo cambios en el campo `expiring_date` para ponerlo en fomato DATE.

```
106
107 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\credit_cards.csv'
108 INTO TABLE credit_card
109 FIELDS TERMINATED BY ','
110 OPTIONALLY ENCLOSED BY '"'
111 LINES TERMINATED BY '\\n'
112 IGNORE 1 ROWS
113 (id, user_id, iban, pan, pin, cvv, track1, track2, @expiring_date)
114 SET expiring_date = STR_TO_DATE(@expiring_date, '%m/%d/%y');
115
```

Output

Action Output

#	Time	Action	Message
✓ 1	17:45:18	LOAD DATA INFILE 'C:\\ProgramData\\MyS...	275 row(s) affected Records: 275 Deleted: 0 Skipped: 0 Warnings: 0

Por último creo la tabla `transaction` indicando como Pk el campo `id`, todos los campos NOT NULL y las FK .

Esta es la tabla de hechos del modelo, contiene FK's de otras tablas y más métricas que el resto

```
118
119 • CREATE TABLE transaction (
120     id VARCHAR(50) PRIMARY KEY,
121     card_id VARCHAR(20) NOT NULL,
122     business_id VARCHAR(20) NOT NULL,
123     timestamp DATETIME NOT NULL,
124     amount DECIMAL(10, 2) NOT NULL,
125     declined BOOLEAN NOT NULL,
126     product_ids VARCHAR(255) NOT NULL,
127     user_id INT NOT NULL,
128     lat DECIMAL(20, 10) NOT NULL,
129     longitude DECIMAL(20, 10) NOT NULL,
130     FOREIGN KEY (card_id) REFERENCES credit_card(id),
131     FOREIGN KEY (business_id) REFERENCES company(id),
132     FOREIGN KEY (user_id) REFERENCES user(id)
133 );
134
```

Output

Action Output

#	Time	Action	Message
✓ 1	18:02:58	CREATE TABLE transaction (id VARCHAR(50) PRIMARY KEY, ca...	0 row(s) affected

Tarea S4.01. Creación de bases de datos

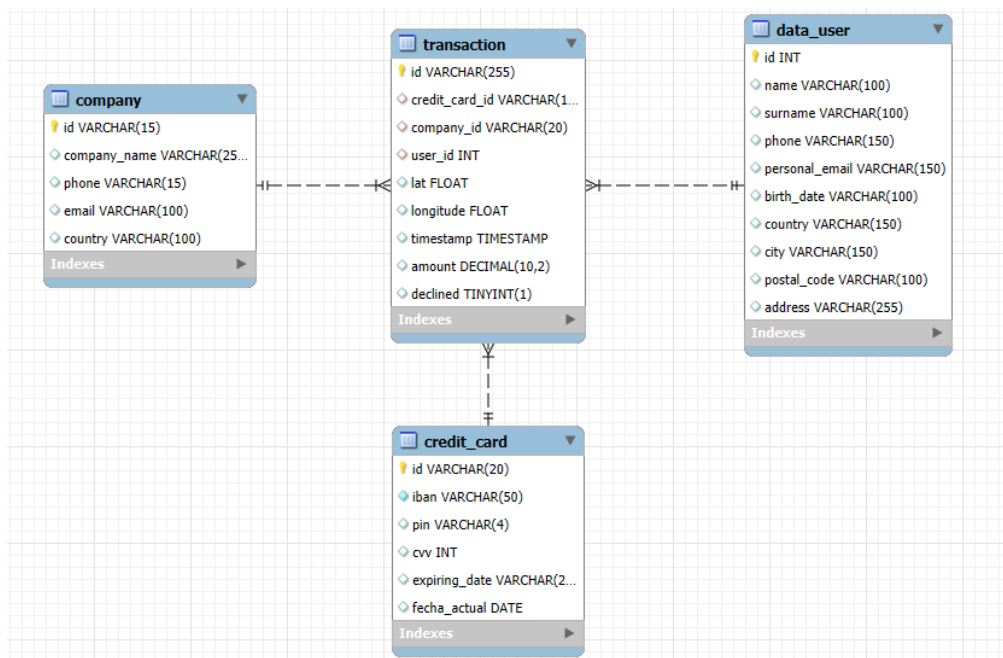
Cargo los datos en la tabla transaction

```
135
136 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\transactions.csv'
137 INTO TABLE transaction
138 FIELDS TERMINATED BY ';'
139 OPTIONALLY ENCLOSED BY '"'
140 LINES TERMINATED BY '\\r\\n'
141 IGNORE 1 ROWS
142 (id, card_id, business_id, timestamp, amount, declined, product_ids, user_id, lat, longitude);
143
```

Output

#	Time	Action	Message
1	18:03:43	LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\U...	587 row(s) affected Records: 587 Deleted: 0 Skipped: 0 Warnings: 0

Ahora que ya tengo creada la Base de Datos a genero el Diagrama ER.



Es un diagrama de estrella compuesto por 4 tablas, la tabla Transaction, que es la tabla de hechos con la Pk id y tres claves foráneas, cómo tal es la tabla que contiene más métrica. La credit_card_id, company_id e user_id y 3 tablas de dimensiones, data_user con Pk id, credit_card, con su Pk id y la company con la Pk id. La cardinalidad de la base de datos es la de modelo de estrella con una relación de n:1 ya que una tarjeta, una compañía o un usuario pueden aparecer en muchas transacciones pero una transacción solo puede tener una tarjeta, un usuario o puede ser realizada por una compañía.

Tarea S4.01. Creación de bases de datos

• Ejercicio 1

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

```
16 • SELECT u.id, u.name, u.surname, u.email
17 FROM user u
18 WHERE u.id IN (
19     SELECT t.user_id
20     FROM transaction t
21     GROUP BY t.user_id
22     HAVING COUNT(t.id) > 30
23 )
```

Result Grid			
Filter Rows: <input type="text"/>			
Edit:			
Export/Import:			
Wrap Cell Content:			
id	name	surname	email
92	Lynn	Riddle	vitae.aliquet@outlook.edu
267	Ocean	Nelson	aenean@yahoo.com
272	Hedwig	Gilbert	sem.eget@idoud.edu
275	Kenyon	Hartman	convallis.ante.lectus@yahoo.com
NULL	NULL	NULL	NULL

• Ejercicio 2

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

```
154
155 -- Ejercicio2
156 • SELECT c.iban, AVG(t.amount) AS average_amount
157 FROM transaction t
158 INNER JOIN credit_card c ON t.card_id = c.id
159 INNER JOIN company comp ON t.business_id = comp.id
160 WHERE comp.name = 'Donec Ltd'
161 GROUP BY c.iban;
162
```

Result Grid	
Filter Rows: <input type="text"/>	
Export:	
Wrap Cell Content:	
iban	average_amount
PT87806228135092429456346	203.715000

Tarea S4.01. Creación de bases de datos

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:

Creo la tabla credit_card_status que solo tendrá un id (que es el id de la tarjeta) y un estado (status). Pongo como PK el id de la tarjeta.

```
---
153                                     -- NIVEL 2 --
154                                     -- Ejercicio 1 --
155 • CREATE TABLE credit_card_status (
156     id VARCHAR(20) PRIMARY KEY,
157     status VARCHAR(30) NOT NULL
158 );
159 |
```



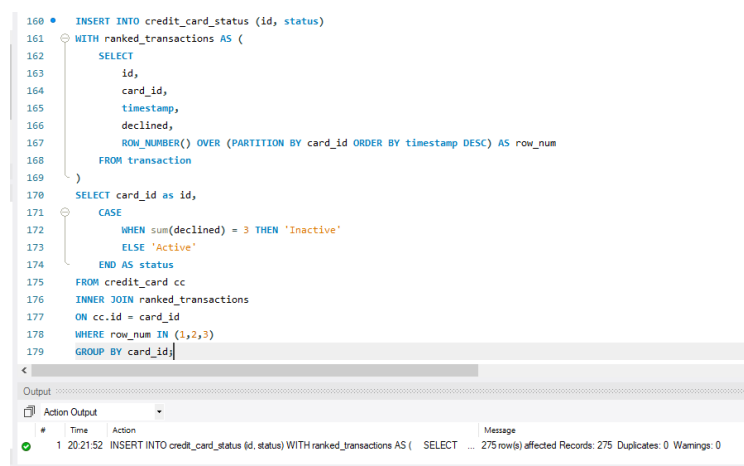
La clave para mí ha sido el row_number(), que me dice el número de fila, y he particionado por card_id y orden descendente, de tal manera que para cada tarjeta me va a dar N filas de transacciones, siendo la fila con row number 1 la más reciente para cada tarjeta.

Hago un join con las credit_card y el conjunto temporal (uso de WITH) que he creado ("ranked_transactions").

Agrupar por card id, filtrando solo las transaction con row number 1, 2 o 3

Sumo los declined y si es igual 3 es que las tres últimas fueron declinadas.

```
160 • INSERT INTO credit_card_status (id, status)
161     WITH ranked_transactions AS (
162     SELECT
163         id,
164         card_id,
165         timestamp,
166         declined,
167         ROW_NUMBER() OVER (PARTITION BY card_id ORDER BY timestamp DESC) AS row_num
168     FROM transaction
169     )
170     SELECT card_id as id,
171     CASE
172     WHEN sum(declined) = 3 THEN 'Inactive'
173     ELSE 'Active'
174     END AS status
175     FROM credit_card cc
176     INNER JOIN ranked_transactions
177     ON cc.id = card_id
178     WHERE row_num IN (1,2,3)
179     GROUP BY card_id;
```

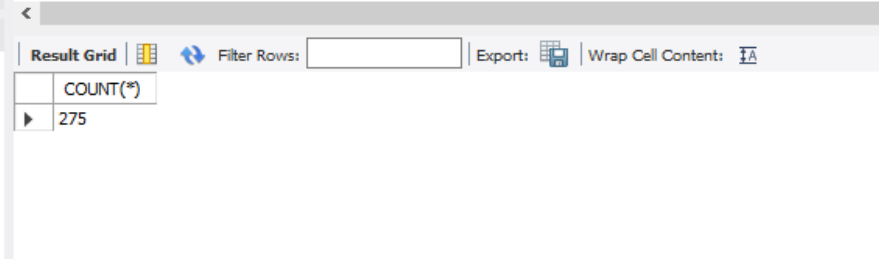


Tarea S4.01. Creación de bases de datos

- Ejercicio 1

¿Cuántas tarjetas están activas?

```
181
182 • SELECT COUNT(*) FROM credit_card_status
183 WHERE status = 'Active';
184
```



The screenshot shows a database query interface. At the top, a SQL query is entered in a text area: `SELECT COUNT(*) FROM credit_card_status WHERE status = 'Active';`. Below the query area, there is a toolbar with options like "Result Grid", "Filter Rows", "Export", and "Wrap Cell Content". The "Result Grid" is currently selected, and it displays a single row with the value 275, representing the count of active credit cards.

COUNT(*)
275

Hay 275 tarjetas activas. Hago la consulta haciendo un conteo en `credit_card_status` filtrando por la de status activas.