

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:

MySQL Workbench

Principal x unconnected x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sakila

sprint4

Tables

- companies
- credit_cards
- products
- transactions
- users

Views

- Stored Procedures
- Functions

sys

- transactions
- world

Administration Schemas

Information

No object selected

Object Info Session

Query Completed

SQL File 4*

```
45 warehouse_id VARCHAR(10) -- hay algunos errores con los guiones ( WH-2 y WH--2)
46 );
47
48 CREATE TABLE IF NOT EXISTS users (
49   id VARCHAR(15) PRIMARY KEY,
50   name VARCHAR(100),
51   surname VARCHAR(100),
52   phone VARCHAR(15),
53   email VARCHAR(100),
54   birth_date VARCHAR(20), -- formato "Mes DD, YYYY"
55   country VARCHAR(25),
56   city VARCHAR(25),
57   postal_code VARCHAR(10), -- viene en un formato extraño con letras
58   address VARCHAR(255)
59 );
60 /*en esta parte declaro todas las FK*/
61 ALTER TABLE transactions
62 ADD constraint Fk_transactions_users foreign key (user_id) REFERENCES users (id),
63 ADD constraint Fk_transactions_companies foreign key (business_id) REFERENCES companies (company_id),
64 ADD constraint Fk_transactions_credit_cards foreign key (card_id) REFERENCES credit_cards (id);
65 /*ADD constraint Fk_transaction_products foreign key (product_ids) REFERENCES products (id); omitire esta ya cada transaccion pu
66
67
68
```

SQLAdditions

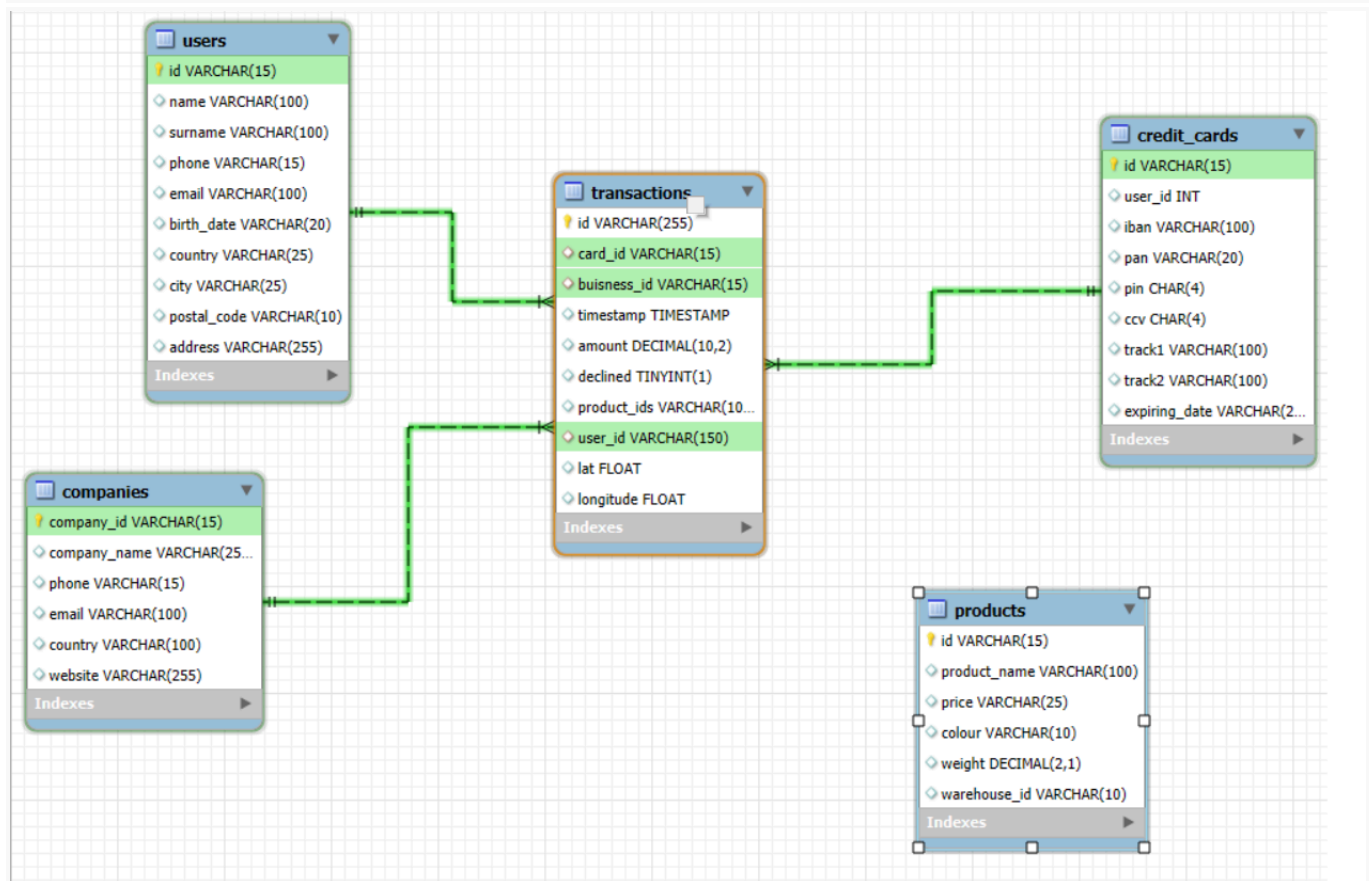
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Context Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
2	11:55:34	ALTER TABLE transactions CHANGE user_id varchar(150)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.125 sec
3	11:56:02	ALTER TABLE transactions ADD constraint Fk_transactions_users foreign key (user_id) REFERENCE...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.187 sec



En esta parte introduzco los datos de los archivos csv a las tablas creadas

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'sakila' database schema with tables like 'companies', 'credit_cards', 'products', 'transactions', and 'users'. The main editor window shows a SQL script in 'SQL File 5*' that includes comments in Spanish and SQL commands to create a table and load data from a CSV file. The 'Output' pane at the bottom shows the execution results, indicating that 587 rows were affected.

```
1 • SHOW VARIABLES LIKE "secure_file_priv"; -- este codigo es para verificar cual es el directorio seguro
2 • LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/transactions.csv' -- los slash van asi "/" no asi "\" porque? no s
3 INTO TABLE transactions
4 FIELDS TERMINATED BY ','
5 LINES TERMINATED BY '\n'
6 IGNORE 1 ROWS;
7
8
9
```

Output:

#	Time	Action	Message	Duration / Fetch
2	13:19:26	LOAD DATA INFIL...	Error Code: 1290. The MySQL server is running with the --secure-file-priv option so it cannot execute this statement	0.000 sec
3	13:20:25	LOAD DATA INFIL...	587 row(s) affected Records: 587 Deleted: 0 Skipped: 0 Warnings: 0	0.093 sec

- Ejercicio 1

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

The screenshot shows the MySQL Workbench interface with a new SQL query in 'SQL File 6*'. The query uses a subquery to find users with more than 30 transactions. The 'Result Grid' pane shows the results of the query, listing user IDs, names, and the number of transactions.

```
1 • SELECT t.user_id, u.name, count(t.user_id) AS num_trans
2 FROM transactions t
3 JOIN users u ON t.user_id=u.id
4 GROUP BY t.user_id, u.name
5 HAVING num_trans >= 30
6
```

Result Grid:

user_id	name	num_trans
267	Ocean	52
272	Hedwig	76
275	Kenyon	48
92	Lynn	39

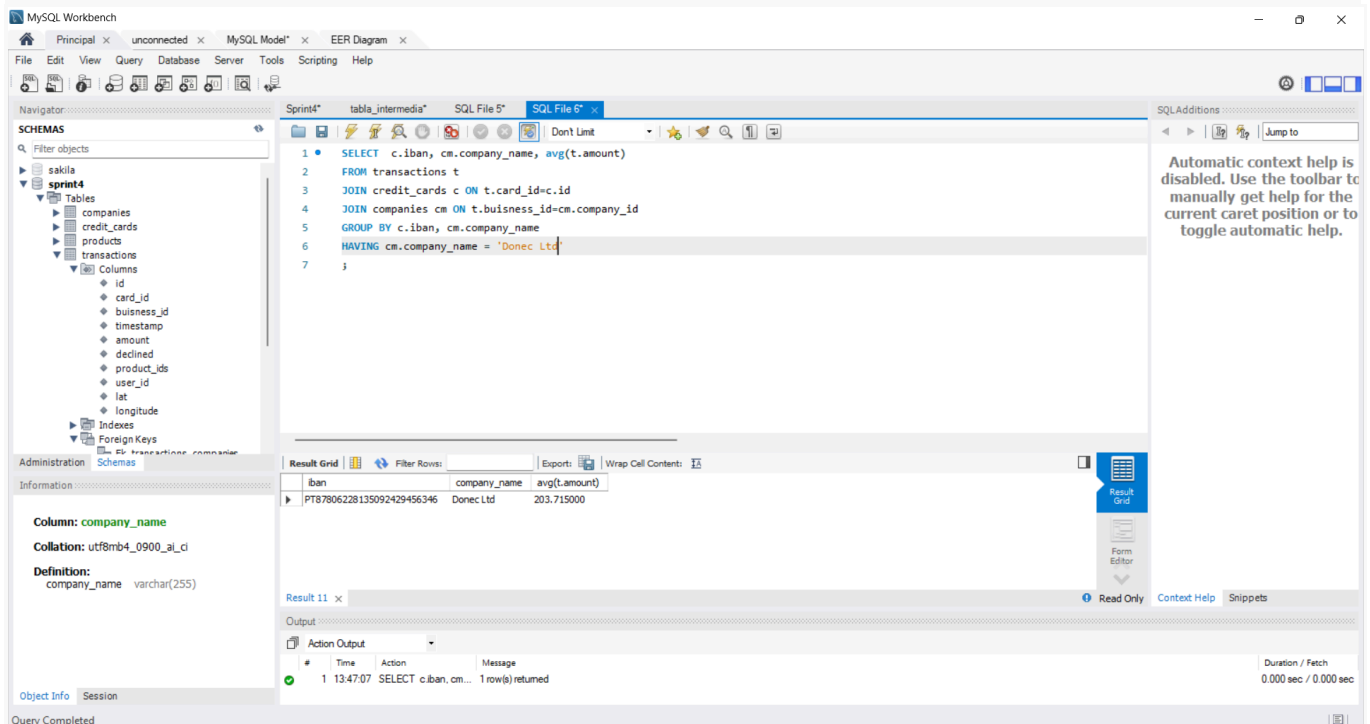
Result 4 x

Output:

#	Time	Action	Message	Duration / Fetch
1	13:30:24	SELECT t.user_id, ...	4 row(s) returned	0.000 sec / 0.000 sec

- 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.



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT c.iban, cm.company_name, avg(t.amount)
2 FROM transactions t
3 JOIN credit_cards c ON t.card_id=c.id
4 JOIN companies cm ON t.business_id=cm.company_id
5 GROUP BY c.iban, cm.company_name
6 HAVING cm.company_name = 'Donec Ltd'
7 ;
```

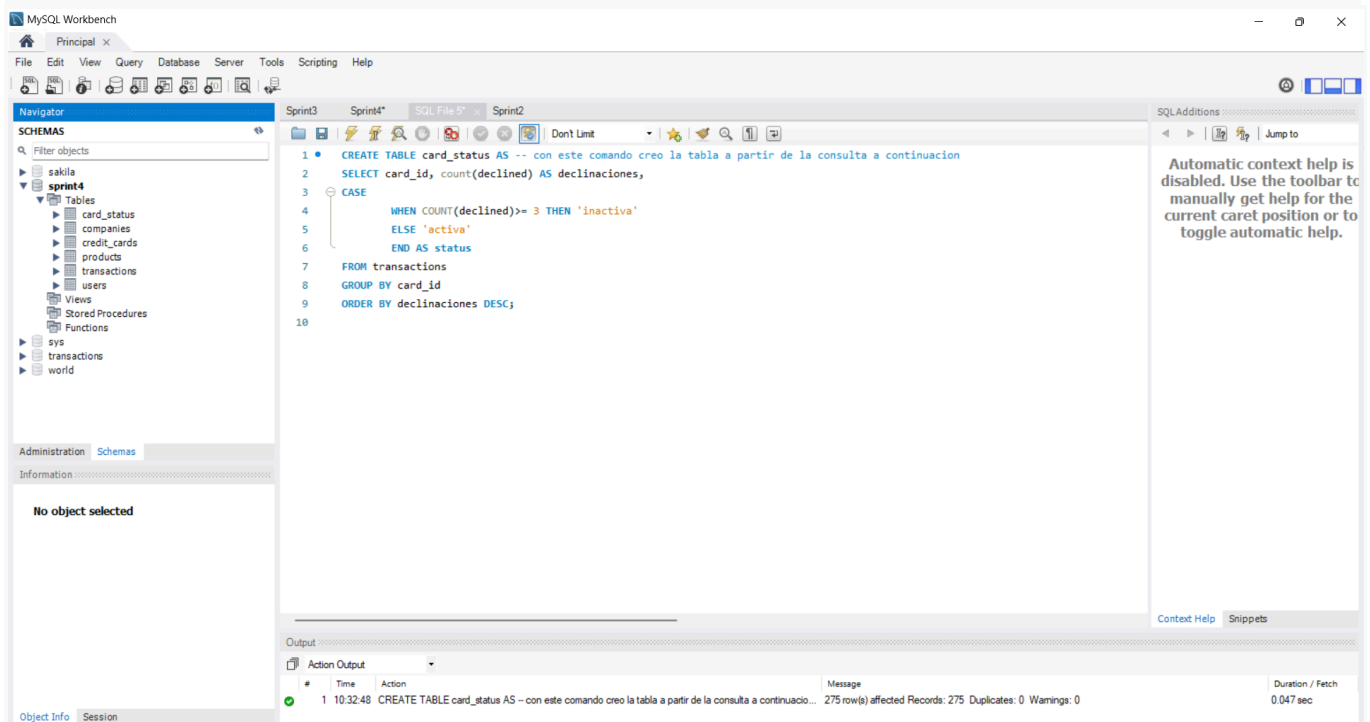
The result grid shows the following data:

iban	company_name	avg(t.amount)
PT87806228135092429456346	Donec Ltd	203.715000

The output pane shows the execution details:

```
1 13:47:07 SELECT c.iban, cm... 1 row(s) returned
Duration / Fetch: 0.000 sec / 0.000 sec
```

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:



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 CREATE TABLE card_status AS -- con este comando creo la tabla a partir de la consulta a continuacion
2 SELECT card_id, count(declined) AS declinaciones,
3 CASE
4 WHEN COUNT(declined)>= 3 THEN 'inactiva'
5 ELSE 'activa'
6 END AS status
7 FROM transactions
8 GROUP BY card_id
9 ORDER BY declinaciones DESC;
10
```

The output pane shows the execution details:

```
1 10:32:48 CREATE TABLE card_status AS -- con este comando creo la tabla a partir de la consulta a continuacio...
275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0
Duration / Fetch: 0.047 sec
```

Ejercicio 1

¿Cuántas tarjetas están activas?

The screenshot shows the MySQL Workbench interface. In the left sidebar, the 'Schemas' panel is open, showing a database named 'sakila' with a schema named 'sprint4'. Under 'sprint4', there is a table named 'card_status'. The 'Table: card_status' information is displayed, showing columns: 'card_id' (varchar(15)), 'declinaciones' (bigint), and 'status' (varchar(8)).

The main editor window shows a SQL script in the 'Sprint4' tab. The script is as follows:

```
85  /*Nivel 2 previo*/
86
87  CREATE TABLE card_status AS -- con este comando creo la tabla a partir de la consulta a continuacion
88  SELECT card_id, count(declined) AS declinaciones,
89  CASE
90  WHEN COUNT(declined)>= 3 THEN 'inactiva'
91  ELSE 'activa'
92  END AS status
93  FROM transactions
94  GROUP BY card_id
95  ORDER BY declinaciones DESC;
96
97  /*Nivel 2 Ejercicio 1*/
98
99  SELECT COUNT(status) as active_cards
100  FROM card_status;
```

The 'Result Grid' shows the execution of the query, displaying a single row with the value '266' for 'active_cards'.

The 'Output' panel at the bottom shows the execution details: '1 10:39:01 SELECT COUNT(status) as active_cards FROM card_status WHERE status = 'activa' 1 row(s) returned 0.000 sec / 0.000 sec'.

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:

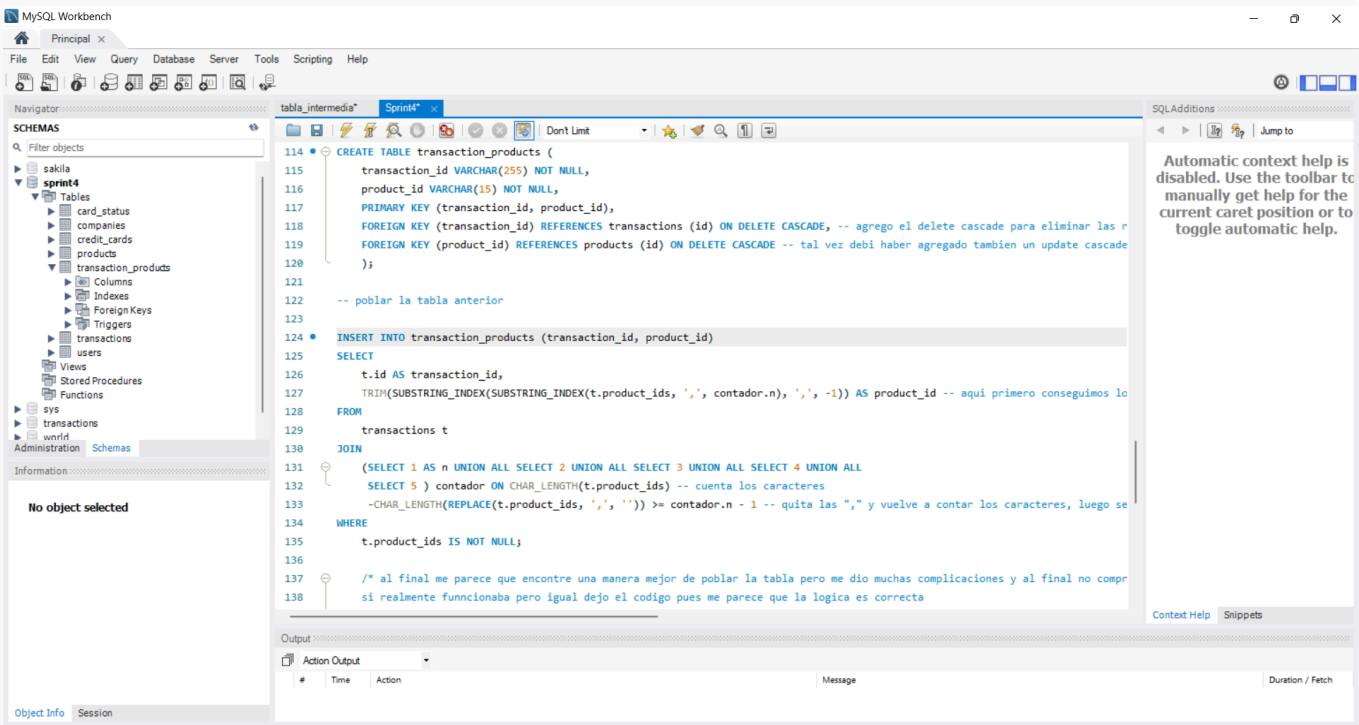
The screenshot shows the MySQL Workbench interface. In the left sidebar, the 'Schemas' panel is open, showing a database named 'sakila' with a schema named 'sprint4'. Under 'sprint4', there is a table named 'transaction_products'.

The main editor window shows a SQL script in the 'Sprint4' tab. The script is as follows:

```
1  CREATE TABLE transaction_products (
2  transaction_id VARCHAR(255) NOT NULL,
3  product_id VARCHAR(15) NOT NULL,
4  PRIMARY KEY (transaction_id, product_id),
5  FOREIGN KEY (transaction_id) REFERENCES transactions (id) ON DELETE CASCADE, -- agrego el delete cascade para eliminar las rel.
6  FOREIGN KEY (product_id) REFERENCES products (id) ON DELETE CASCADE
7  );
8
9
```

The 'Output' panel at the bottom shows the execution details: '1 10:43:48 CREATE TABLE transaction_products (transaction_id VARCHAR(255) NOT NULL, product_id V... 0 row(s) affected 0.047 sec'.

el siguiente código es el que use para poblar la tabla recién creada



Ejercicio 1

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

