



Tema 4: Importación y Exportación desde Workbench

MYSQL
STUDIUM



Introducción

En un taller del tema anterior, se ha trabajado con la importación y exportación desde la línea de comandos. Ahora vamos a realizar las mismas tareas y con los mismos fines, pero desde nuestro querido MySQL Workbench.

Importaremos desde un fichero CSV que es bastante habitual, y también trabajaremos con la Exportación e Importación de ficheros típicamente de SQL. Comencemos con CSV.

Primeros pasos

Primero, crear el **fichero Excel** con los datos de los registros que queremos insertar. El orden de las columnas no tiene por qué coincidir con el orden de los campos en la tabla destino. Tampoco el nombre de las columnas, pero al menos, que se parezcan y que tengan un nombre significativo.

Partimos de una base de datos con la siguiente tabla llamada "coches":

| Column Name | Datatype | PK | NN | UQ | B | UN | ZF | AI | G | Default/Expression |
|----------------------|-------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------|
| numeroMatriculaCoche | VARCHAR(10) | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| marcaCoche | VARCHAR(45) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| modeloCoche | VARCHAR(45) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| colorCoche | VARCHAR(45) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| fechaCompra | DATE | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| dniPersonaFK | VARCHAR(9) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |

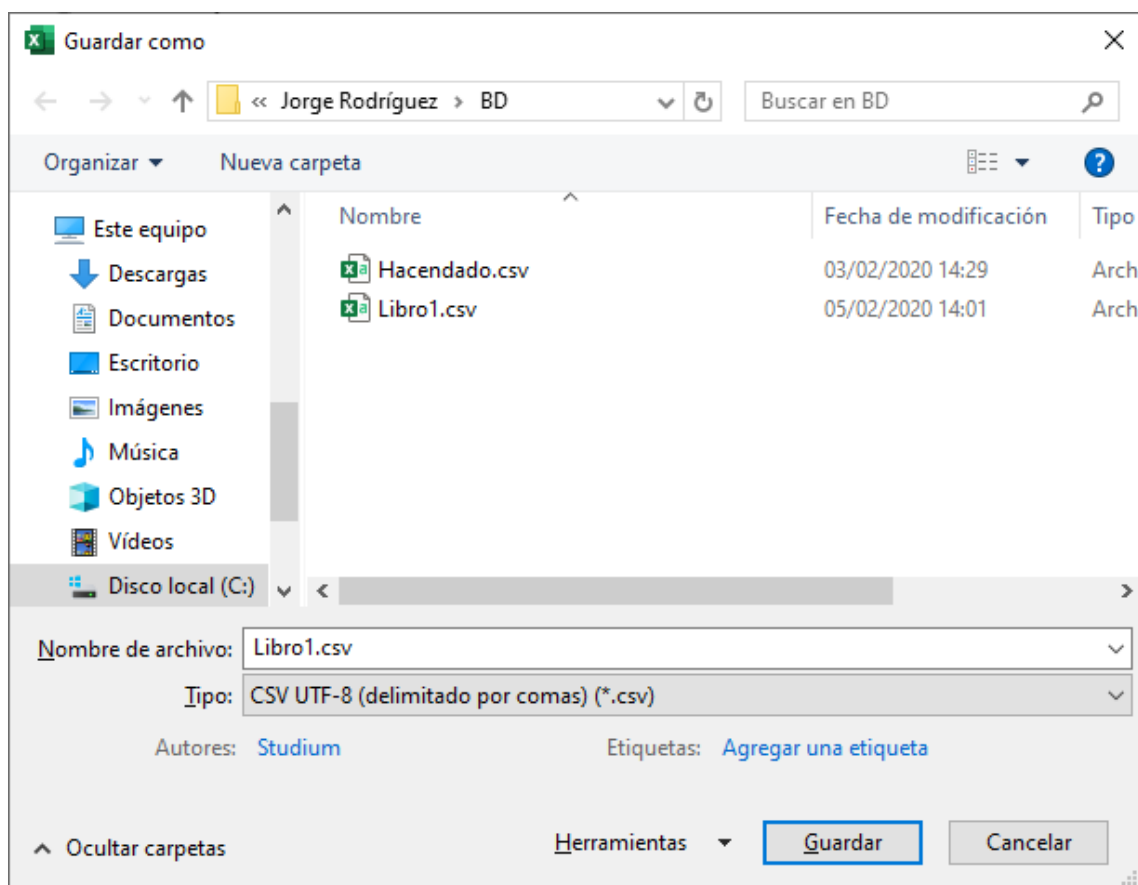
Ahora, en Excel, insertaremos varios registros con las columnas correspondientes a los campos de esta tabla:

| | A | B | C | D | E | F |
|---|-----------|--------|--------|------------|-----------|------------|
| 1 | matricula | marca | modelo | color | dni | fecha |
| 2 | 1232-BGG | Toyota | Rav4 | Gris perla | 12344785F | 2020-01-02 |
| 3 | 4585-HLM | Nissan | Micra | Blanco | 12457845G | 2020-02-04 |

Cuidado con los campos tipo **fecha**, que deben ir en formato americano (YYYY-MM-DD).

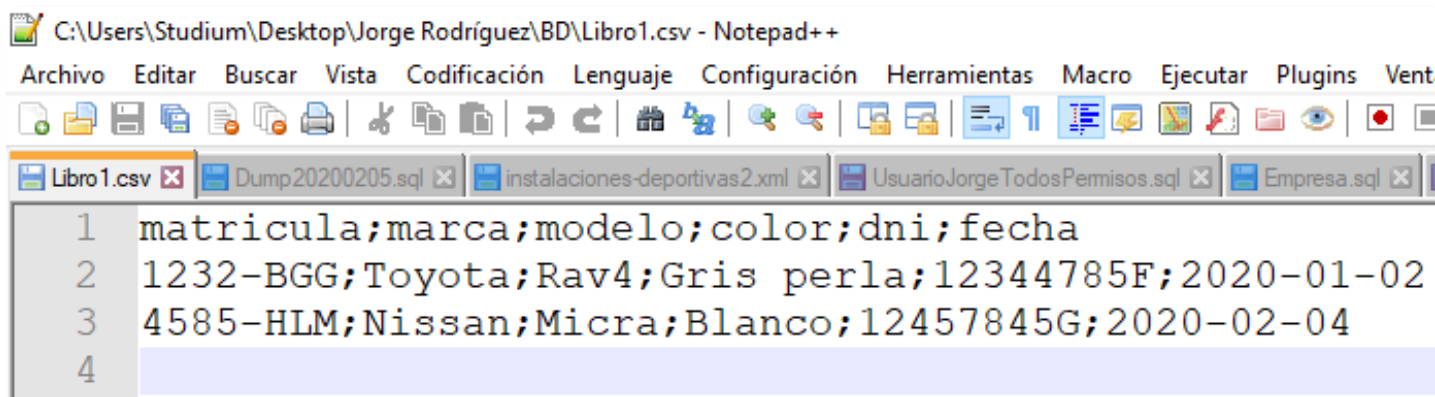


Ahora, podemos guardar como Excel normal, pero para nuestro objetivo, debemos usar otro formato de guardado:



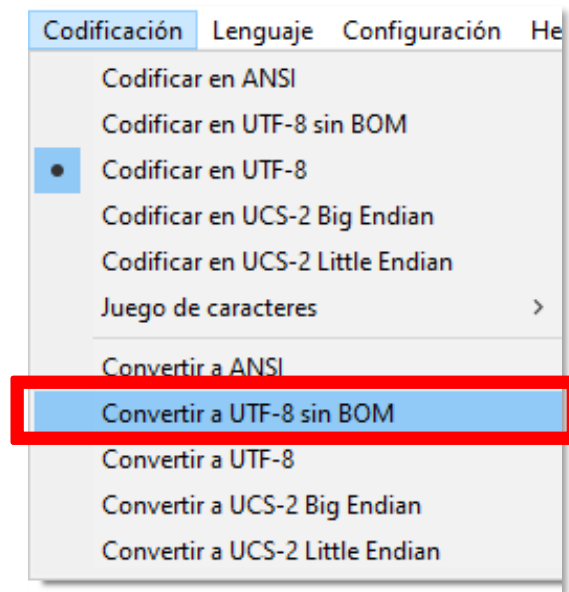
Cuidado que hay varios tipos de CSV. Nosotros necesitamos precisamente ese, el "CSV UTF-8 (delimitando por comas) (*.csv)". Y salimos de Excel.

Ahora debemos abrir el fichero CSV con un **editor de texto** que permita trabajar con **codificación de caracteres**. En mi caso usaré el software gratuito NotePad++:





En la opción de “Codificación”, debemos elegir “Convertir a UTF-8 sin BOM”:



Cuidado: Hay que elegir **Convertir**, no Codificar

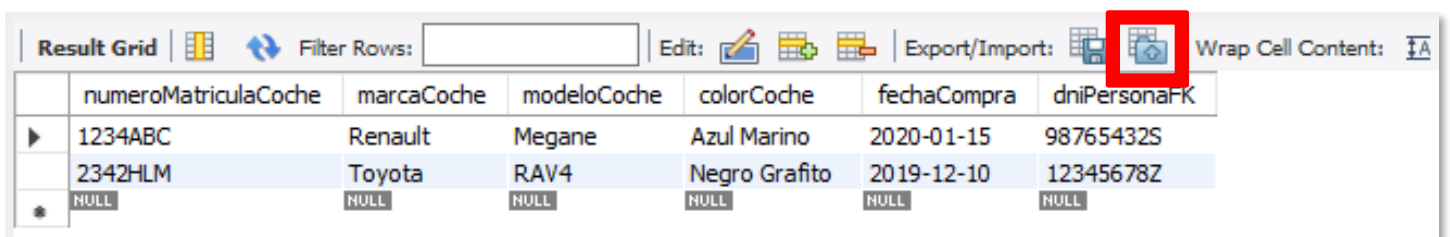
Tras lo cual, debemos guardar el fichero y cerrar NotePad++.

Importación de CSV

Entraremos en nuestro **MySQL Workbench**, desde donde haremos la importación. Tenemos dos posibilidades:

Desde la ventana de resultados de una consulta

Mostraremos el contenido de la tabla destino (`SELECT * FROM tabla_correspondiente`) de los datos que provienen de la hoja Excel:

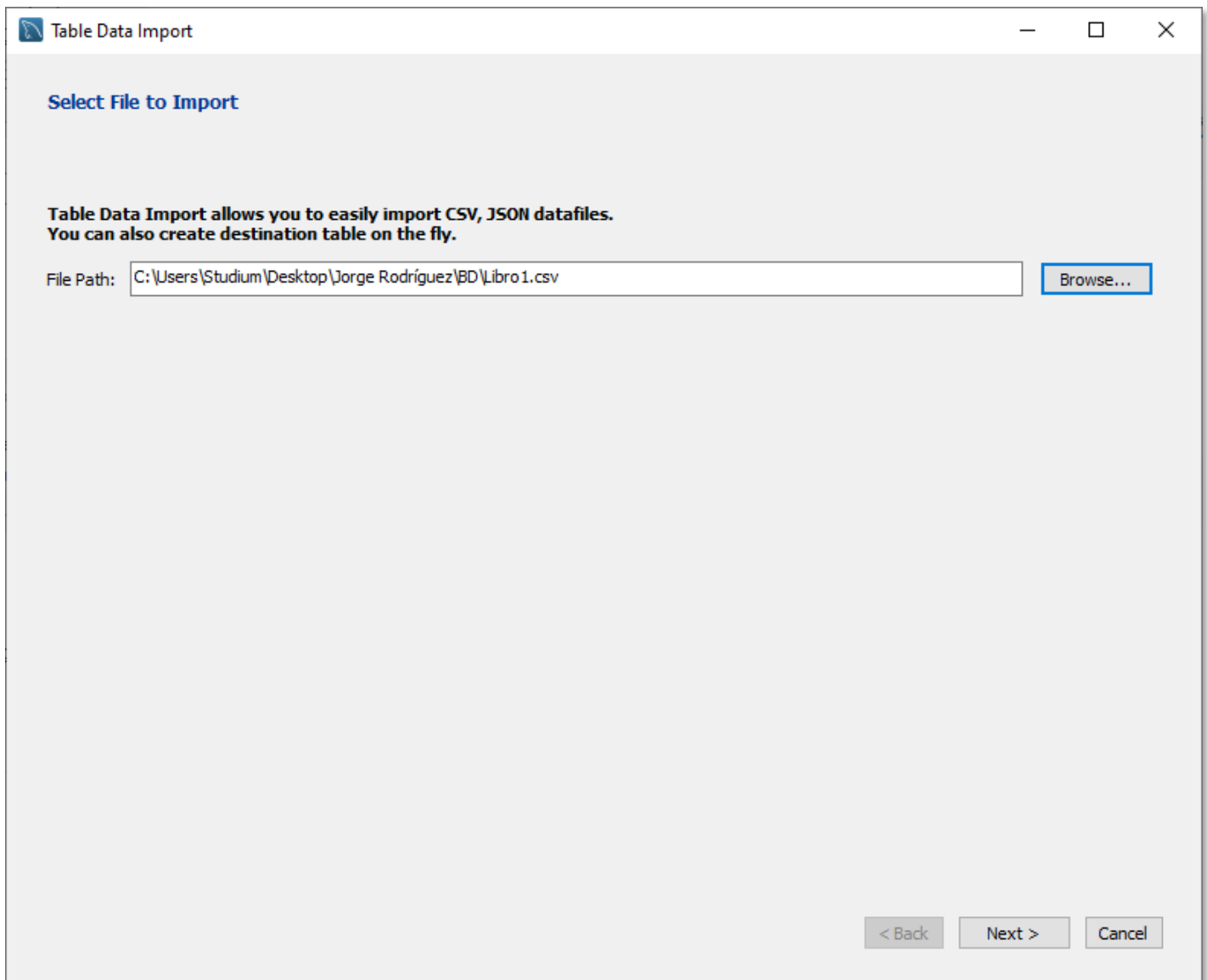


| | numeroMatriculaCoche | marcaCoche | modeloCoche | colorCoche | fechaCompra | dniPersonaFK |
|---|----------------------|------------|-------------|---------------|-------------|--------------|
| ▶ | 1234ABC | Renault | Megane | Azul Marino | 2020-01-15 | 98765432S |
| | 2342HLM | Toyota | RAV4 | Negro Grafito | 2019-12-10 | 12345678Z |
| * | NULL | NULL | NULL | NULL | NULL | NULL |

Pulsaremos el botón de “Importar” representado por una cuadrícula, detrás de una carpeta azul con una flecha en su interior, como se indica en la figura anterior.



Elegiremos el fichero csv transformado recién guardado con “Browse...”:



Tras pulsar “Next”, debemos elegir si queremos importar los datos en una tabla YA existente (primera opción) o crear sobre la marcha una tabla nueva para que recepcione los datos que estamos importando. Esta opción es menos recomendable pues la estructura será inferida según los tipos de los datos, pero puede haber problemas con campos clave, claves foráneas, los propios tipos elegido... Aunque todo se puede arreglar luego con un buen ALTER TABLE.



Table Data Import

Select Destination

Select destination table and additional options.

☒ Use existing table: ejercicio9.coches

☐ Create new table: ejercicio9 . Libro1

☐ Truncate table before import

< Back Next > Cancel

Si marcamos la opción de “Truncate table before import”, se borrarán todos los datos que ya existan en la tabla destino. Pero no es nuestro caso.

Así que, tras elegir la primera opción, pulsamos en “Next”.



Table Data Import

Configure Import Settings

Detected file format: csv

Encoding:

utf-8

Columns:

| <input checked="" type="checkbox"/> Source Column | Dest Column |
|---|--------------------------------|
| <input checked="" type="checkbox"/> matricula | <div>numeroMatriculaCoch</div> |
| <input checked="" type="checkbox"/> marca | <div>marcaCoche</div> |
| <input checked="" type="checkbox"/> modelo | <div>modeloCoche</div> |
| <input checked="" type="checkbox"/> color | <div>colorCoche</div> |
| <input checked="" type="checkbox"/> dni | <div>fechaCompra</div> |
| <input checked="" type="checkbox"/> fecha | <div>dniPersonaFK</div> |

| matricula | marca | modelo | color | dni | fecha |
|-----------|--------|--------|------------|-----------|------------|
| 1232-BGG | Toyota | Rav4 | Gris perla | 12344785F | 2020-01-02 |
| 4585-HLM | Nissan | Micra | Blanco | 12457845G | 2020-02-04 |

< Back

Next >

Cancel

En esta pantalla se configuran varios aspectos. El primero es la codificación, que si hemos hecho todos los pasos anteriores con Excel y con NotePad++, no tendremos que cambiar a una codificación diferentes de "utf-8" tal como aparece por defecto.

Lo siguiente que tenemos que establecer es la concordancia entre las columnas del fichero CSV (izquierda) con los campos de la tabla en la columna derecha.



En mi caso, hay un descuadre entre las columnas "dni" y "fecha" y los campos "fechaCompra" y "dniPersonaFK". Simplemente, haciendo uso de los desplegables podemos emparejarlos adecuadamente antes de pulsar en "Next":

Table Data Import

Configure Import Settings

Detected file format: csv

Encoding:

utf-8

Columns:

| <input checked="" type="checkbox"/> | Source Column | Dest Column |
|-------------------------------------|---------------|--------------------------------|
| <input checked="" type="checkbox"/> | matricula | <div>numeroMatriculaCoch</div> |
| <input checked="" type="checkbox"/> | marca | <div>marcaCoche</div> |
| <input checked="" type="checkbox"/> | modelo | <div>modeloCoche</div> |
| <input checked="" type="checkbox"/> | color | <div>colorCoche</div> |
| <input checked="" type="checkbox"/> | dni | <div>dniPersonaFK</div> |
| <input checked="" type="checkbox"/> | fecha | <div>fechaCompra</div> |

| matricula | marca | modelo | color | dni | fecha |
|-----------|--------|--------|------------|-----------|------------|
| 1232-BGG | Toyota | Rav4 | Gris perla | 12344785F | 2020-01-02 |
| 4585-HLM | Nissan | Micra | Blanco | 12457845G | 2020-02-04 |

< Back

Next >

Cancel



En la siguiente pantalla simplemente pulsamos "Next" para iniciar proceso:

Table Data Import

Import Data

The following tasks will now be performed. Please monitor the execution.

☐ Prepare Import

☐ Import data file

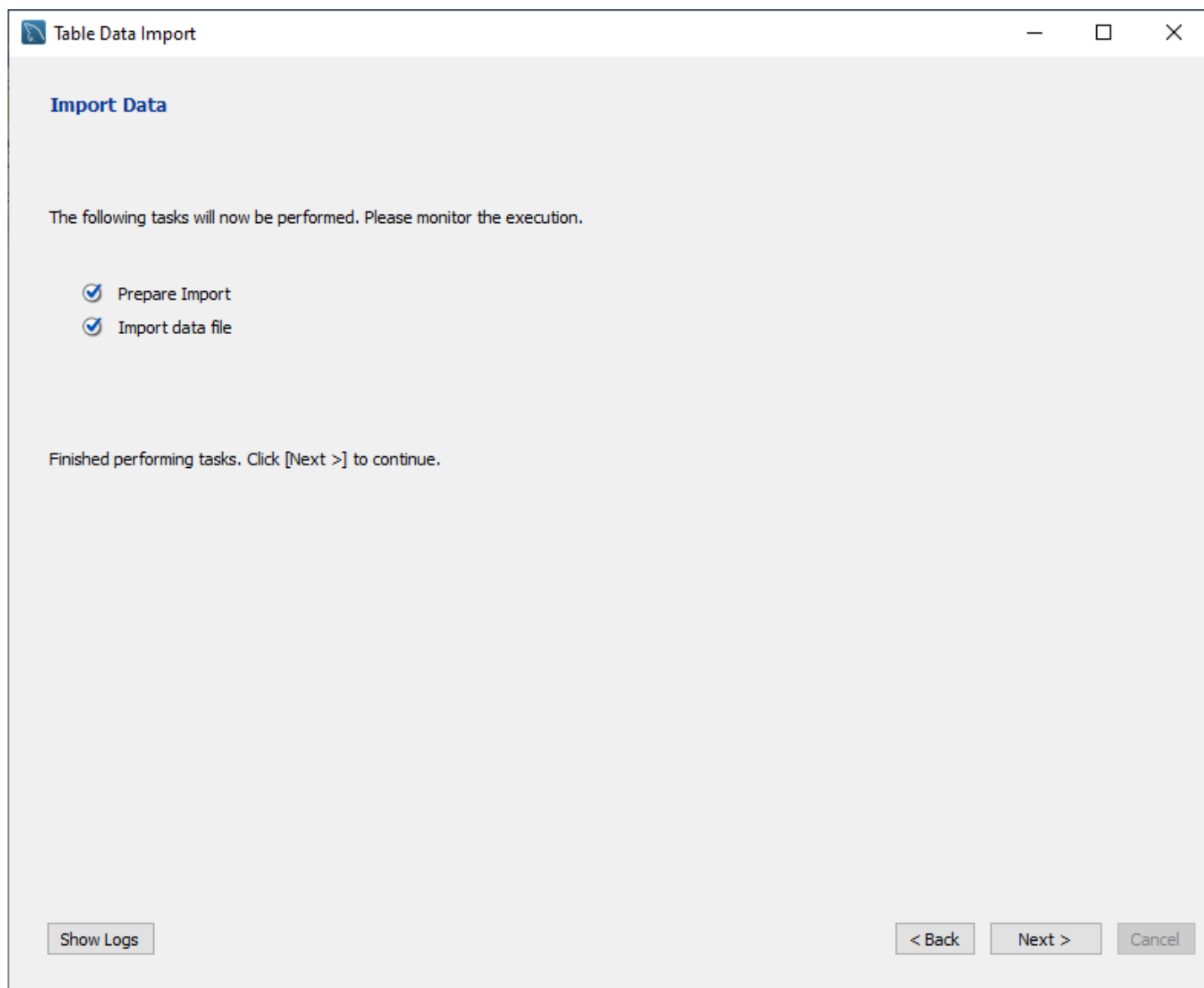
Click [Next >] to execute.

Show Logs

< Back Next > Cancel

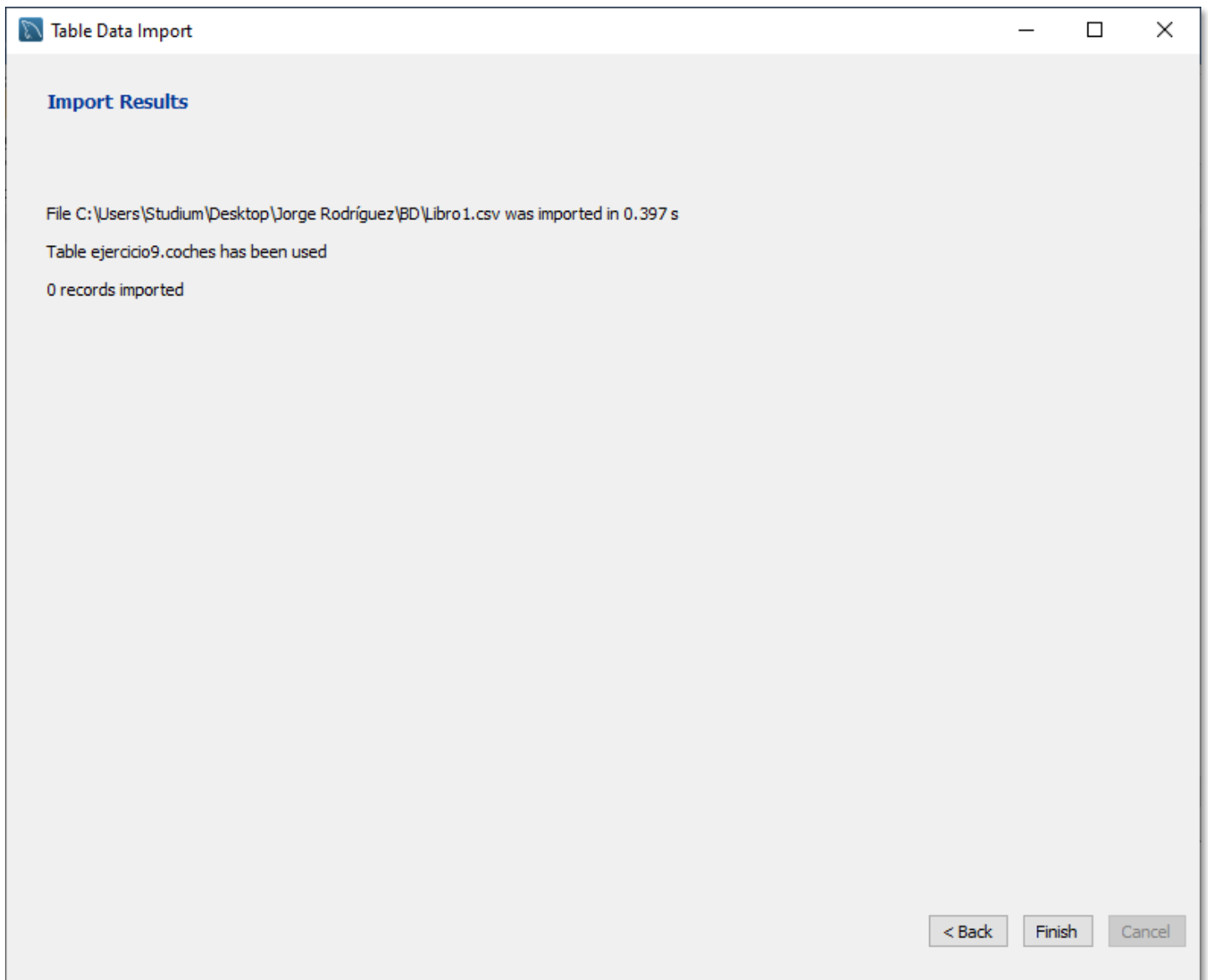


Si todo ha ido bien:





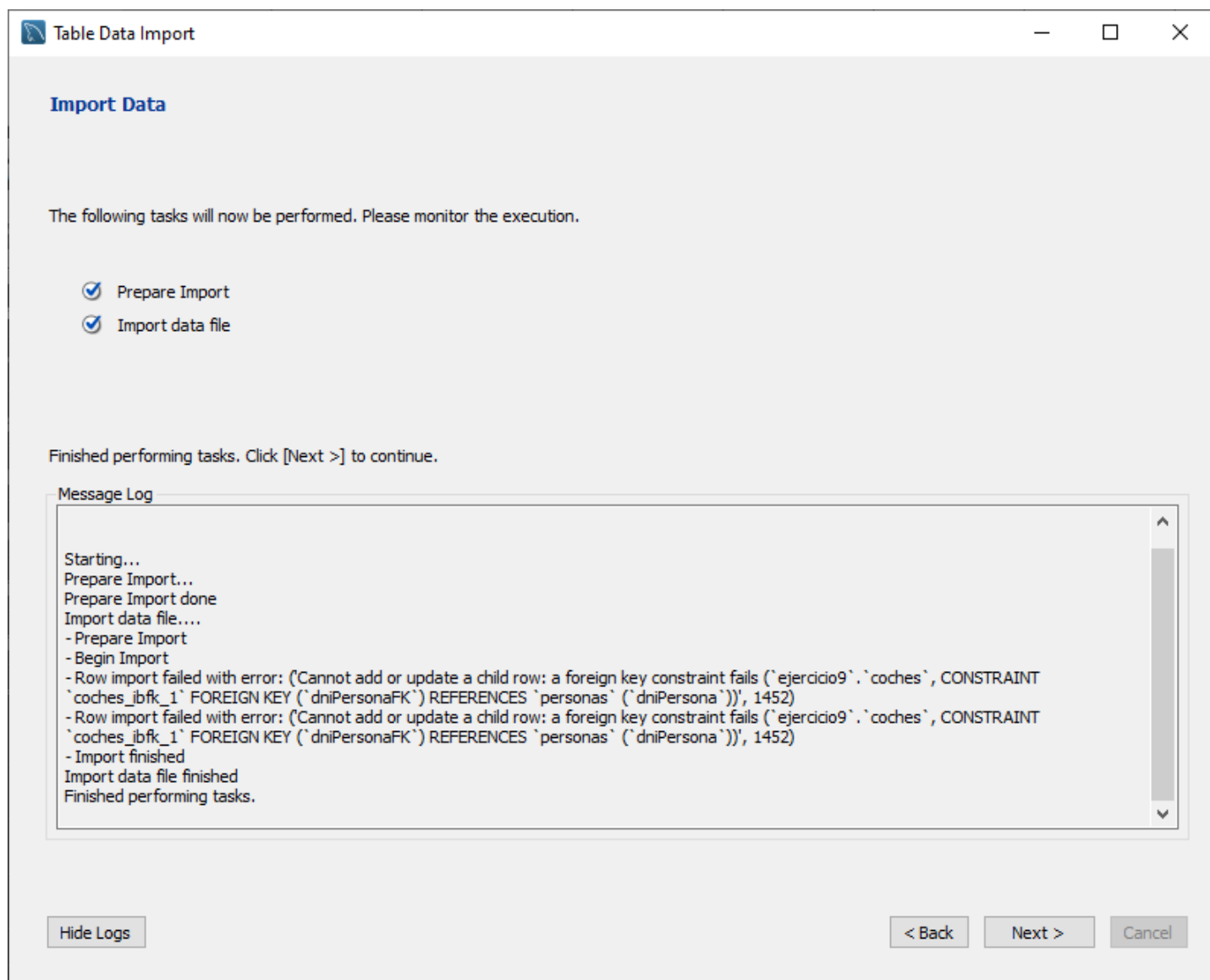
Volver a pulsar "Next":



Y se nos informa de cómo ha ido la importación.



Si hay algún error (Observad cómo se indica que se han importado 0 registros), debemos volver atrás pulsando “Back”, pulsar en “Show logs” y volver a intentar la importación:



Aquí vemos que, por ejemplo, el campo *dniPersonaFK* debe tener un valor referenciado en la otra tabla, en **personas**. Debemos rehacer el fichero CSV metiendo datos válidos.

Y volvemos al principio para realizar correctamente la importación:

| | numeroMatriculaCoche | marcaCoche | modeloCoche | colorCoche | fechaCompra | dniPersonaFK |
|---|----------------------|------------|-------------|---------------|-------------|--------------|
| ▶ | 1232-BGG | Toyota | Rav4 | Gris perla | 2020-01-02 | 12121212D |
| | 1234ABC | Renault | Megane | Azul Marino | 2020-01-15 | 98765432S |
| | 2342HLM | Toyota | RAV4 | Negro Grafito | 2019-12-10 | 12345678Z |
| | 4585-HLM | Nissan | Micra | Blanco | 2020-02-04 | 12554578D |
| ★ | NULL | NULL | NULL | NULL | NULL | NULL |

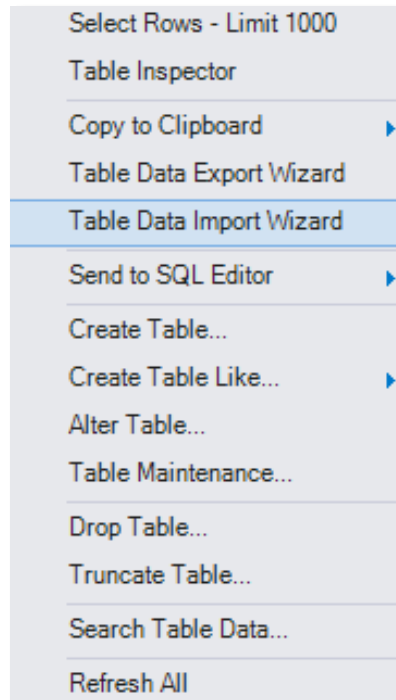


Todo el proceso, otra vez.

Desde asistente (Wizard)

Ahora veamos cómo realizar esta misma importación desde un fichero CSV, pero haciendo uso de un **Asistente** o **Wizard**.

Con el botón secundario del ratón sobre la tabla en la que queremos importar los datos sacaremos el menú contextual para elegir la opción “Table Data Import Wizard”:

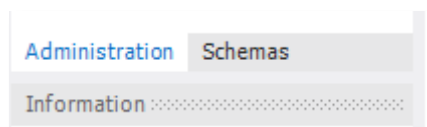


Se nos pregunta por el nombre del fichero y la ruta de este. A partir de aquí, el proceso es idéntico al descrito con anterioridad desde la página 4.

Exportando desde MySQL Workbench

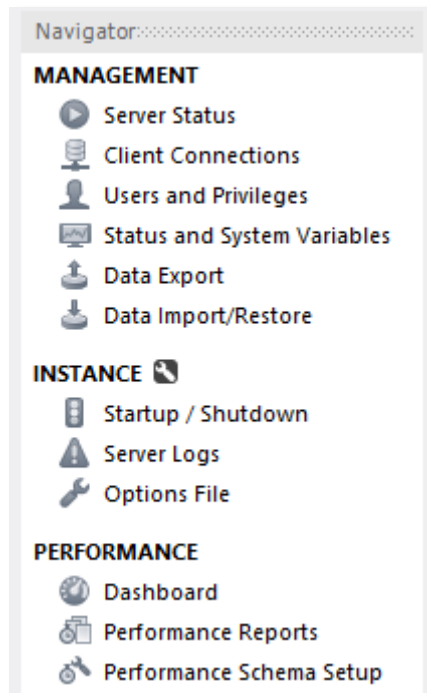
Para exportar desde Workbench es bien sencillo.

Debemos en primer lugar acceder a las opciones de “Administration” en la pestaña correspondiente en la parte izquierda central:

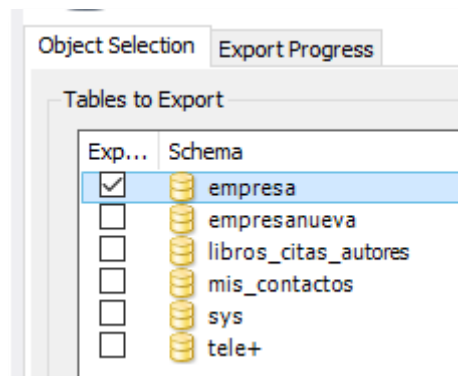




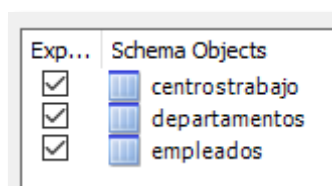
El menú superior mostrará ahora las opciones de administración. Elegimos, obviamente, la opción de “Data Export”:



En “Tables to Export” elegimos las bases de datos a exportar, que pueden ser más de una.



Debemos asegurarnos de que se marque el recuadro de check y que la base de datos o esquema también se quede marcado en celeste para que, en la parte derecha, aparezcan las tablas de dicha base de datos, como podemos ver en la siguiente imagen:

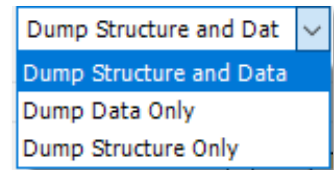


Ahora, en la parte inferior debemos elegir entre varias opciones de exportación.



Justo debajo de las tablas, hay un desplegable en el que elegir si queremos exportar:

- Estructura y Datos
- Sólo datos, o
- Sólo Estructura



Lo siguiente a elegir es el tipo de exportación, entre “Export to Dump Project Folder” que hace la copia en una carpeta, un fichero por cada tabla, o bien “Export to Self-Contained File” que exporta todo en un único fichero. En ambos casos debemos elegir ubicación, y en esta segunda opción, también el nombre del fichero.

Se recomienda por comodidad esta segunda opción.

Por último, también recomendado, se puede marcar la opción para incluir la sentencia de creación de la base de datos en “Include Create Schema”.

Una vez configurada toda la exportación pulsamos el botón de “Start Export” y comienza la exportación según los parámetros establecidos.

En cualquier caso, si editamos los ficheros generados, nos encontraremos con sentencias SQL que permiten crear bases de datos, crear tablas, insertar datos en tablas, etc.

Importando de fichero SQL

Igual de sencillo que exportar es importar.

Como bien hemos dicho al final del apartado anterior, los ficheros generados son ficheros de texto en cuyo interior podemos encontrar sentencias SQL.

Para importar dicha información podemos hacer un proceso similar al de Exportación acudiendo al menú de “Administration” y luego a la opción “Data Import/Restore” con prácticamente las mismas opciones que para la Exportación.

Pero, por otro lado, podríamos hacer la importación ejecutando directamente en una pestaña de MySQL Workbench todas y cada una de las instrucciones que podemos encontrar en los ficheros de texto generados en la Exportación.



02/08/2023