



TRABAJANDO CON MySQL

Configuración del entorno



Esta obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional](#).

Icono diseñado por Flaticon

Arrancando el servidor de MySQL

Lo primero que debes hacer es arrancar el servidor de MySQL .

Existen múltiples formas de hacerlo y nosotros nos hemos decantado por arrancarlo mediante Docker :

<https://www.docker.com/>

Docker es un servicio de virtualización que permite desplegar de forma aplicaciones (o servicios) dentro de contenedores software.

Arrancando el servidor de MySQL

Antes de arrancar MySQL debes ejecutar **Docker** . Sigue estos pasos:

1. Abre la carpeta **ETSISI-X64** de tu escritorio.
2. Abre la carpeta **Parar-Arrancar Servicios**.
3. Ejecuta **Arrancar Docker.bat**.

Con esto preparamos nuestro equipo del CIC para que sea capaz de lanzar el servicio de **Docker** .

Si instalas **Docker** en tu ordenador, estos pasos no serán necesarios, ya que por defecto **Docker** se inicia al arrancar el sistema.

Arrancando el servidor de MySQL

Ahora vamos a ejecutar el programa `Docker Desktop` que dispone tanto del servicio de `Docker` para gestionar los contenedores y como de una interfaz gráfica que nos facilita la gestión de estos.

Busca la aplicación `Docker Desktop` en el menú de inicio y ejecutala.

Arrancando el servidor de MySQL

Cuando la aplicación arranque, deberíamos ver su interfaz:



Arrancando el servidor de MySQL

Con `Docker` funcionando, vamos a lanzar el servidor de MySQL :

1. Abre un terminal
 - i. Abre el menú de inicio
 - ii. Escribe "*Ejecutar*" y abre el programa
 - iii. Escribe "*cmd*" y lanza el comando para que se abra un terminal
2. En el terminal, escribe el siguiente comando:

```
docker run --name mysql -e MYSQL_ALLOW_EMPTY_PASSWORD=yes -p 3306:3306 -d mysql:8.0
```

Arrancando el servidor de MySQL

- `docker run` indica a `Docker` que ejecute un comando
- `--name mysql` pone nombre al contenedor que vamos a ejecutar
- `-e MYSQL_ALLOW_EMPTY_PASSWORD=yes` permite acceder a la base de datos sin contraseña. **Esto nunca debe hacerse en entornos de producción.**
- `-p 3306:3306` expone el puerto `3306` del contenedor para que sea accesible desde el exterior.
- `-d` arranca el contenedor como un demonio (servicio).
- `mysql:8.0` indica que se use el servidor `MySQL` en su versión `8.0.x`.

Arrancando el servidor de MySQL

Cuando ejecutemos el comando deberíamos ver lo siguiente:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.19043.1237]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\fernando.ortega>docker run --name mysql -e MYSQL_ALLOW_EMPTY_PASSWORD=yes -p 3306:3306 -d mysql:8.0
Unable to find image 'mysql:8.0' locally
8.0: Pulling from library/mysql
b380bbd43752: Pull complete
f23cbf2ecc5d: Pull complete
30cfc6c29c0a: Pull complete
b38609286cbe: Pull complete
8211d9e66cd6: Pull complete
2313f9eeca4a: Pull complete
7eb487d00da0: Pull complete
a5d2b117a938: Pull complete
1f6cb474cd1c: Pull complete
896b3fd2ab07: Pull complete
532e67ebb376: Pull complete
233c7958b33f: Pull complete
Digest: sha256:5d52dc010398db422949f079c76e98f6b62230e5b59c0bf7582409d2c85abacb
Status: Downloaded newer image for mysql:8.0
a4156d302fc51116db46fb53971568a81b54b05126778d86a9ebd8438e21b142

C:\Users\fernando.ortega>
```


Arrancando el servidor de MySQL

Y **Docker Desktop** debería estar en el siguiente estado:



Arrancando el servidor de MySQL

Cuando se lanza el servidor de `MySQL` puede aparecer una advertencia del *firewall* para permitir conexión desde el exterior. Al no tener permisos de administrador, diremos que **Cancelar** y solo podremos trabajar en local.



Ejecutando MySQL Workbench

Con el servidor de `MySQL` ejecutándose, vamos a abrir el cliente de `MySQL Workbench` :

1. Abre la carpeta **ETSI-SI-X64** de tu escritorio.
2. Ejecuta la aplicación **MySQLWorkbench**.

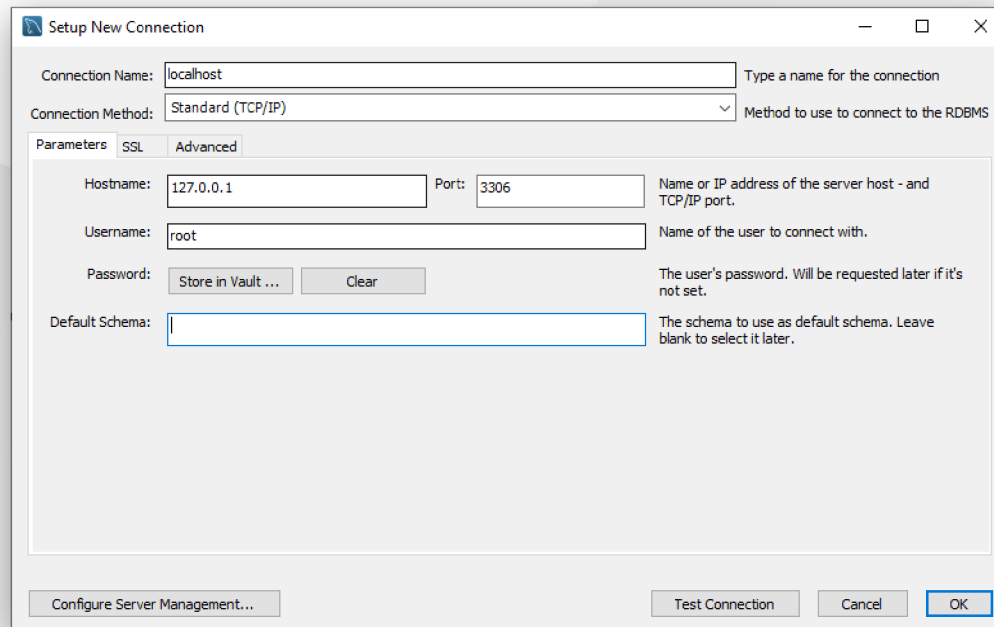
Ejecutando MySQL Workbench



Ejecutando MySQL Workbench

Vamos a crear una nueva conexión para que MySQL Workbench se conecte a nuestro servidor de MySQL .

1. Pulsa el icono + que hay junto a *MySQL Connections*.
2. Rellena los campos de acuerdo con la siguiente imagen y pulsa **Ok**.



The screenshot shows the 'Setup New Connection' dialog box in MySQL Workbench. The dialog has a title bar with a minus, maximize, and close button. It contains the following fields and options:

- Connection Name:** A text field containing 'localhost'. To its right is the text 'Type a name for the connection'.
- Connection Method:** A dropdown menu showing 'Standard (TCP/IP)'. To its right is the text 'Method to use to connect to the RDBMS'.
- Parameters tab:** Three tabs are visible: 'Parameters' (selected), 'SSL', and 'Advanced'.
- Hostname:** A text field containing '127.0.0.1'. To its right is a 'Port' field containing '3306'. To the right of the port field is the text 'Name or IP address of the server host - and TCP/IP port.'
- Username:** A text field containing 'root'. To its right is the text 'Name of the user to connect with.'
- Password:** A text field is empty. To its right are two buttons: 'Store in Vault ...' and 'Clear'. To the right of these buttons is the text 'The user's password. Will be requested later if it's not set.'
- Default Schema:** A text field is empty. To its right is the text 'The schema to use as default schema. Leave blank to select it later.'

At the bottom of the dialog, there are four buttons: 'Configure Server Management...', 'Test Connection', 'Cancel', and 'OK' (which is highlighted with a blue border).

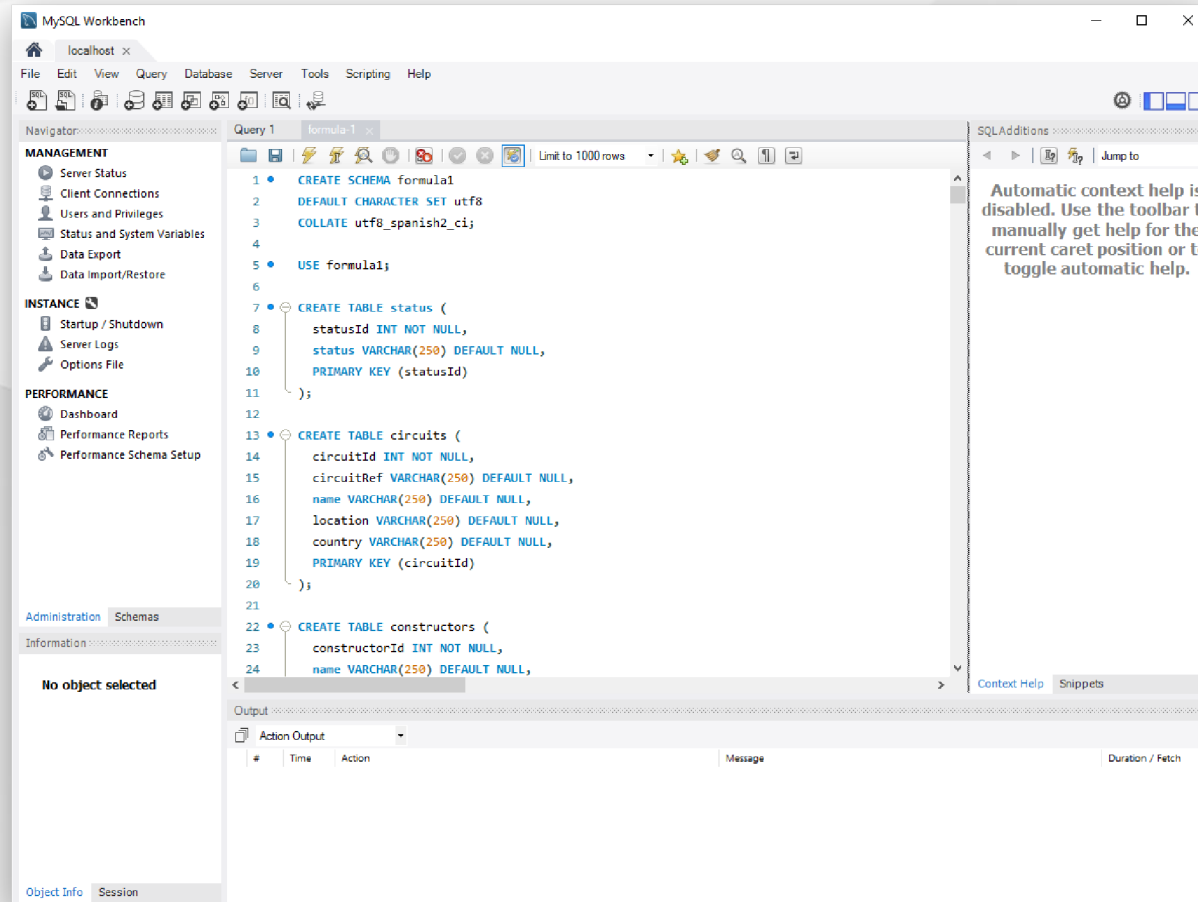
Cargando la base de datos de Formula 1

Para finalizar, vamos a cargar la base de datos de Formula 1 que usaremos en los ejercicios prácticos.

1. Pulsa **File**.
2. Pulsa **Open SQL Script**.
3. Abre el fichero **formula1.sql** que encontrarás en Moodle.

Cargando la base de datos de Formula 1

Ahora pulsa el icono del rayo.



Cargando la base de datos de Formula 1

La base de datos se habrá cargado y podremos trabajar con ella.

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'formula1' database selected. The 'Query' editor in the center contains the SQL query: `SELECT * FROM formula1.drivers;`. The 'Result Grid' below the query displays a table with 16 rows of driver data. The 'Output' pane at the bottom shows the execution log, including the query execution and the number of rows returned (842 rows).

driverId	number	code	forename	surname	birthday	nationality
1	44	HAM	Lewis	Hamilton	1985-01-07	British
2	0	HEI	Nick	Heidfeld	1977-05-10	German
3	6	ROS	Nico	Rosberg	1985-06-27	German
4	14	ALO	Fernando	Alonso	1981-07-29	Spanish
5	0	KOV	Heikki	Kovalainen	1981-10-19	Finnish
6	0	NAK	Kazuki	Nakajima	1985-01-11	Japanese
7	0	BOU	Sebastian	Bourdais	1979-02-28	French
8	7	RAI	Kimi	Raikkonen	1979-10-17	Finnish
9	0	KUB	Robert	Kubica	1984-12-07	Polish
10	0	GLO	Timo	Glock	1982-03-18	German
11	0	SAT	Takuma	Sato	1977-01-28	Japanese
12	0	PIQ	Nelson	Piquet Jr.	1985-07-25	Brazilian
13	19	MAS	Felipe	Massa	1981-04-25	Brazilian
14	0	COU	David	Coulthard	1971-03-27	British
15	0	TRU	Jarno	Trulli	1974-07-13	Italian
16	99	SUT	Adrian	Sutil	1983-01-11	German

Output:

#	Time	Action	Message	Duration / Fetch
57	10:05:46		684 row(s) affected Records: 684 Duplicates: 0 Warnings: 0	0.047 sec
58	10:05:46		690 row(s) affected Records: 690 Duplicates: 0 Warnings: 0	0.031 sec
59	10:05:46		692 row(s) affected Records: 692 Duplicates: 0 Warnings: 0	0.047 sec
60	10:05:46	INSERT INTO 'results' ('resultId', 'raceId', 'driverId', 'constructorId', 'num...	110 row(s) affected Records: 110 Duplicates: 0 Warnings: 0	0.016 sec
61	10:06:07	SELECT * FROM formula1.pitstops LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
62	10:06:18	SELECT * FROM formula1.drivers LIMIT 0, 1000	842 row(s) returned	0.000 sec / 0.000 sec