



# 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 **ETSIISI-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.

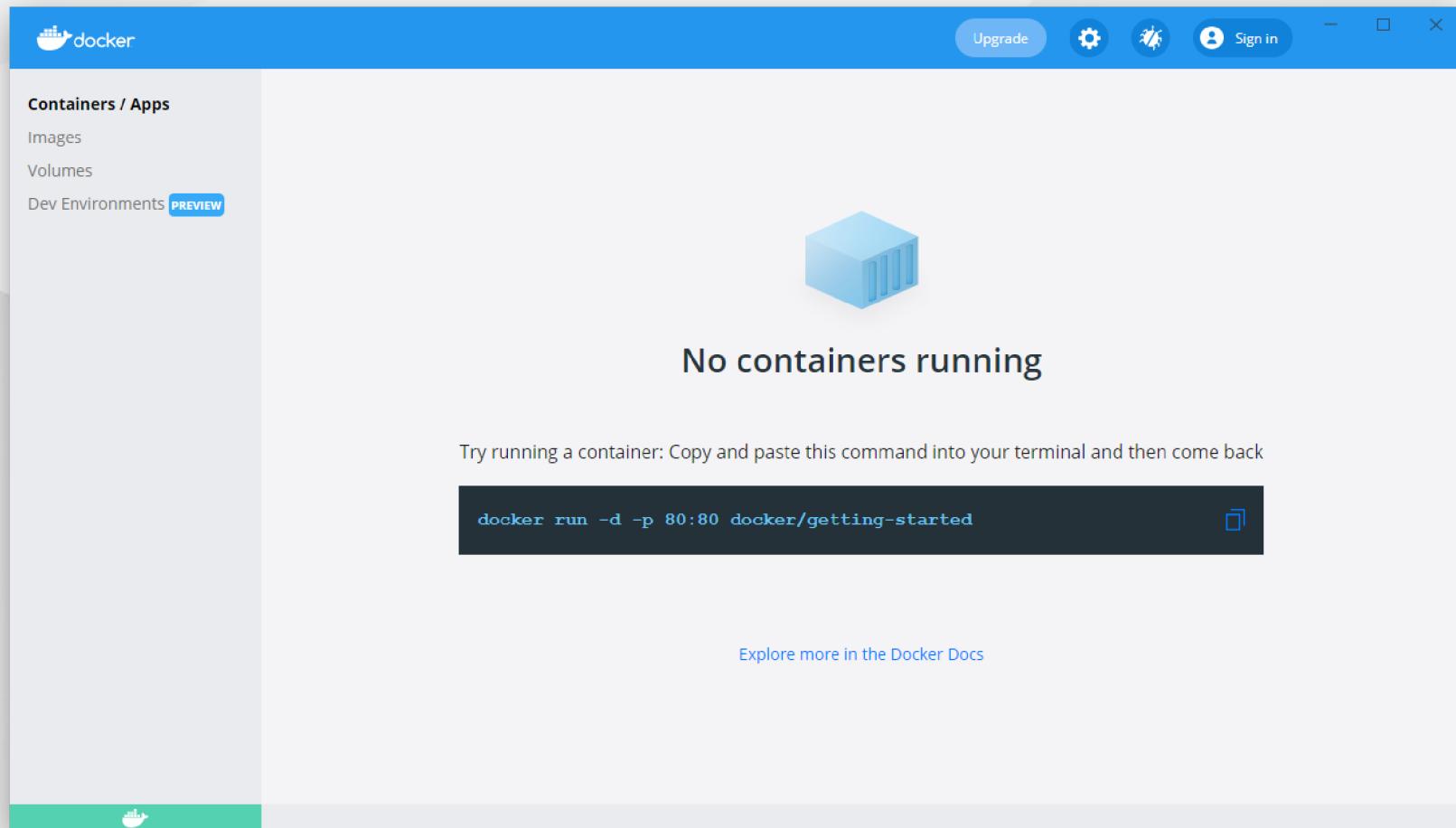
## Arrancando el servidor de MySQL

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 ejecútala.

# 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:

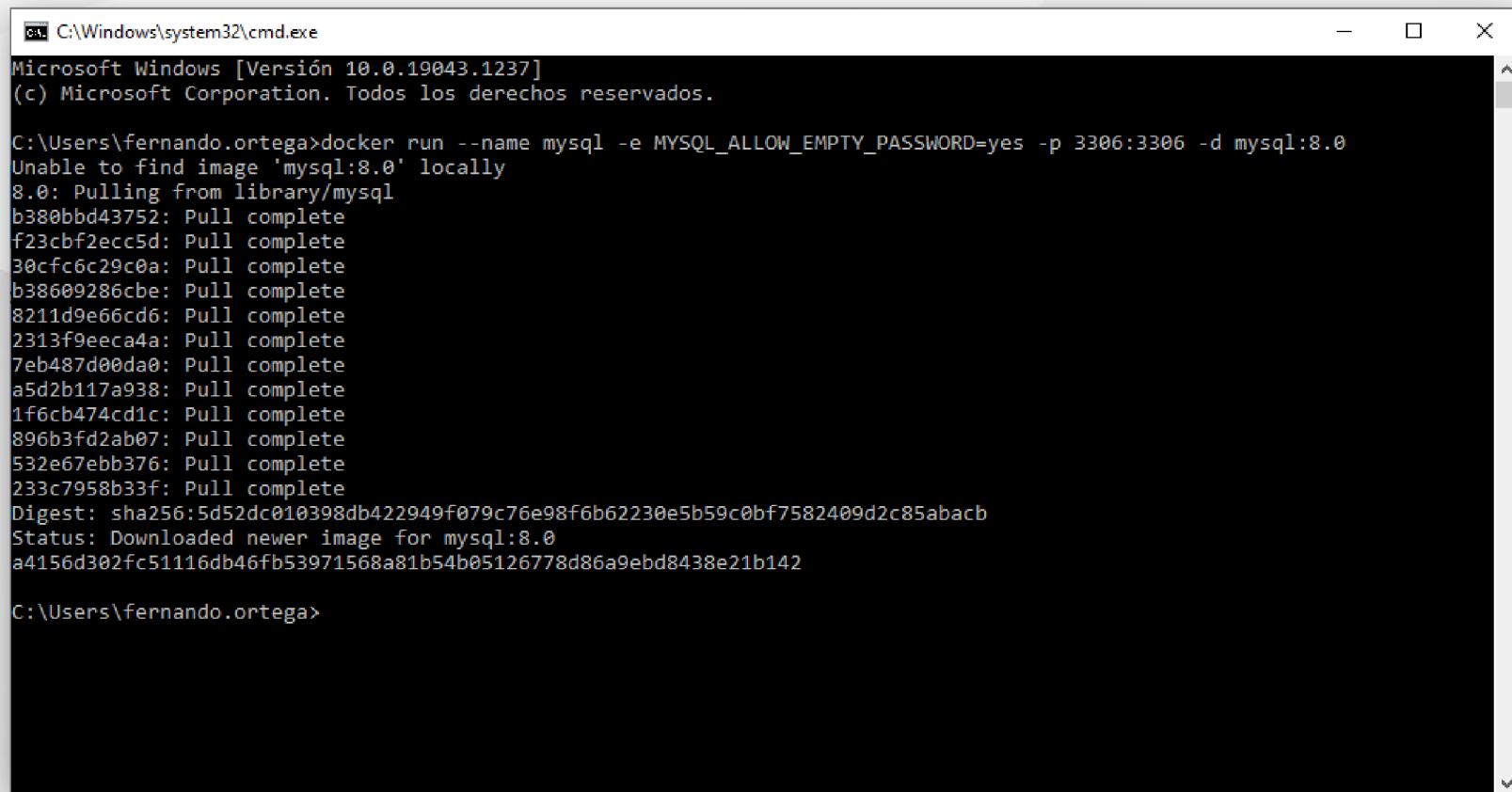
```
docker run --name mysql -e MYSQL_ALLOW_EMPTY_PASSWORD=yes -p 3306:3306 -d mysql:8.0
```

# 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:



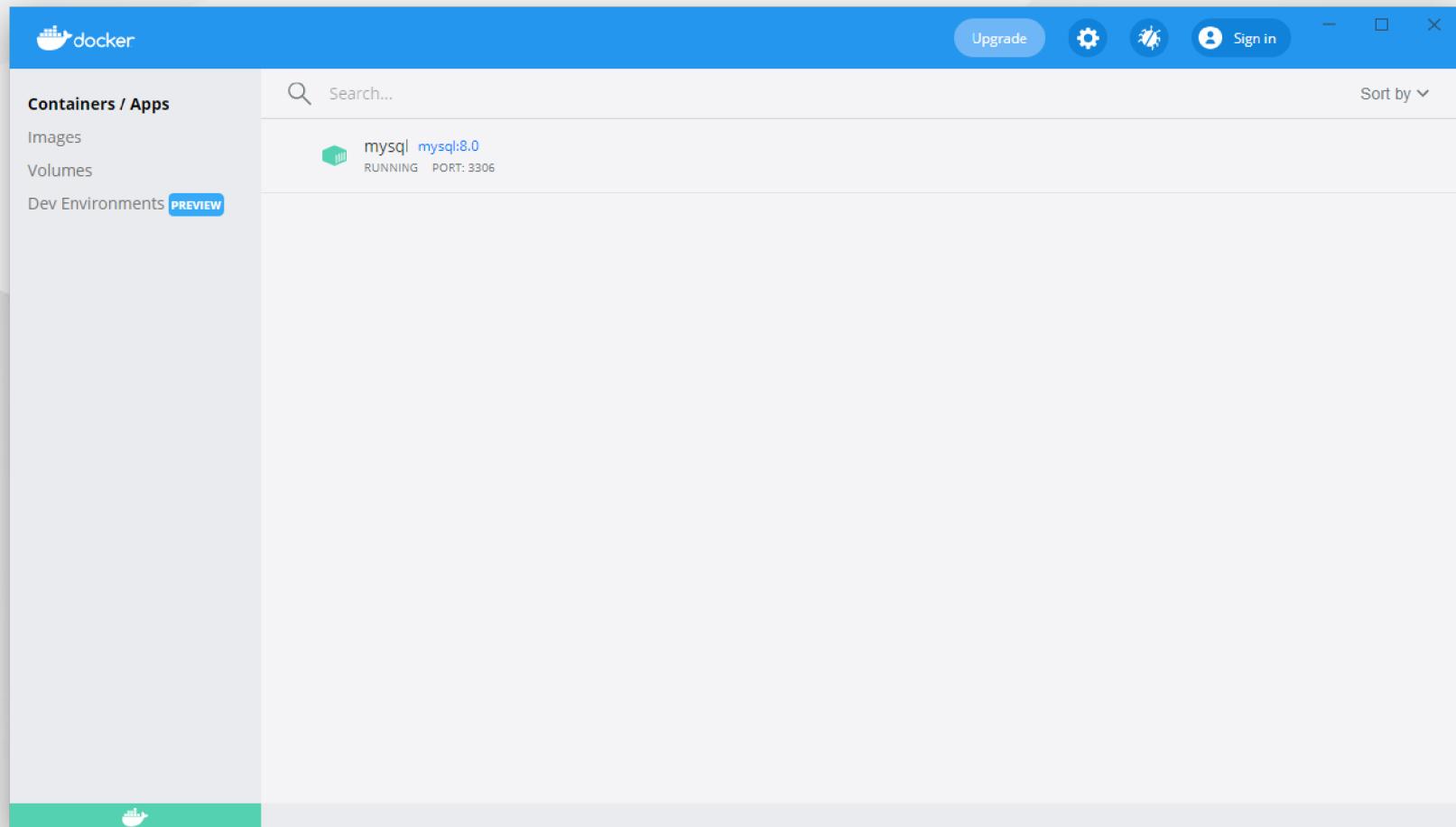
```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.19043.1237]
(c) Microsoft Corporation. Todos los derechos reservados.

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
2313f9eeeca4a: 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>
```

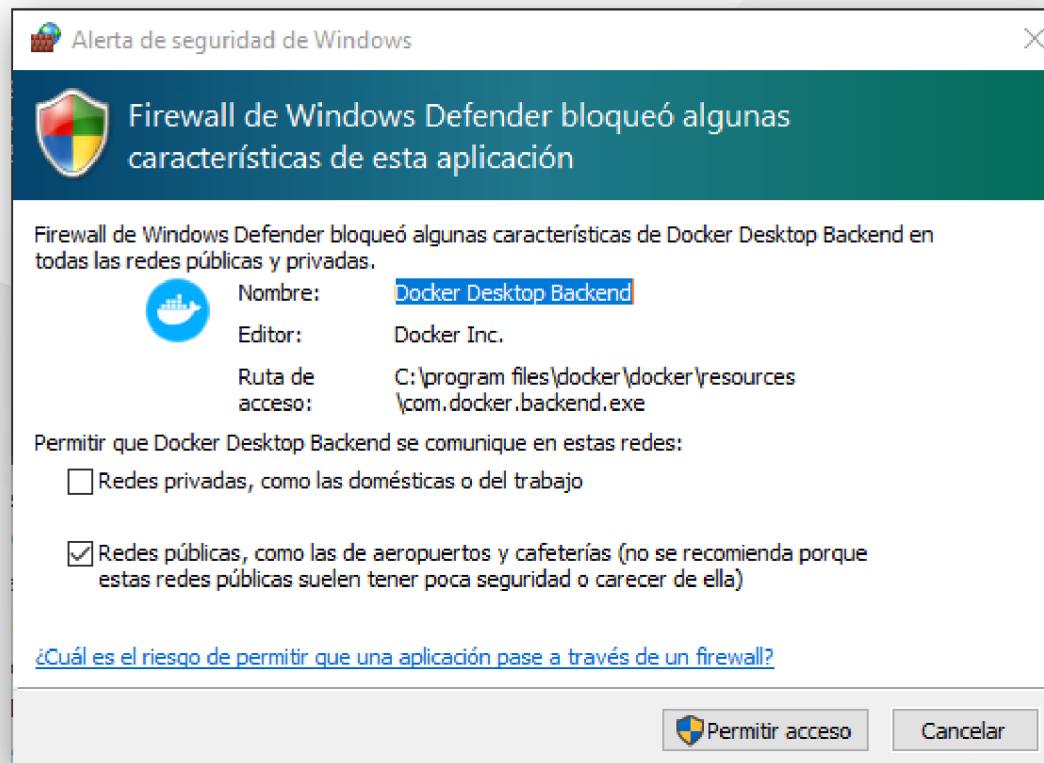
# Arrancando el servidor de MySQL

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.

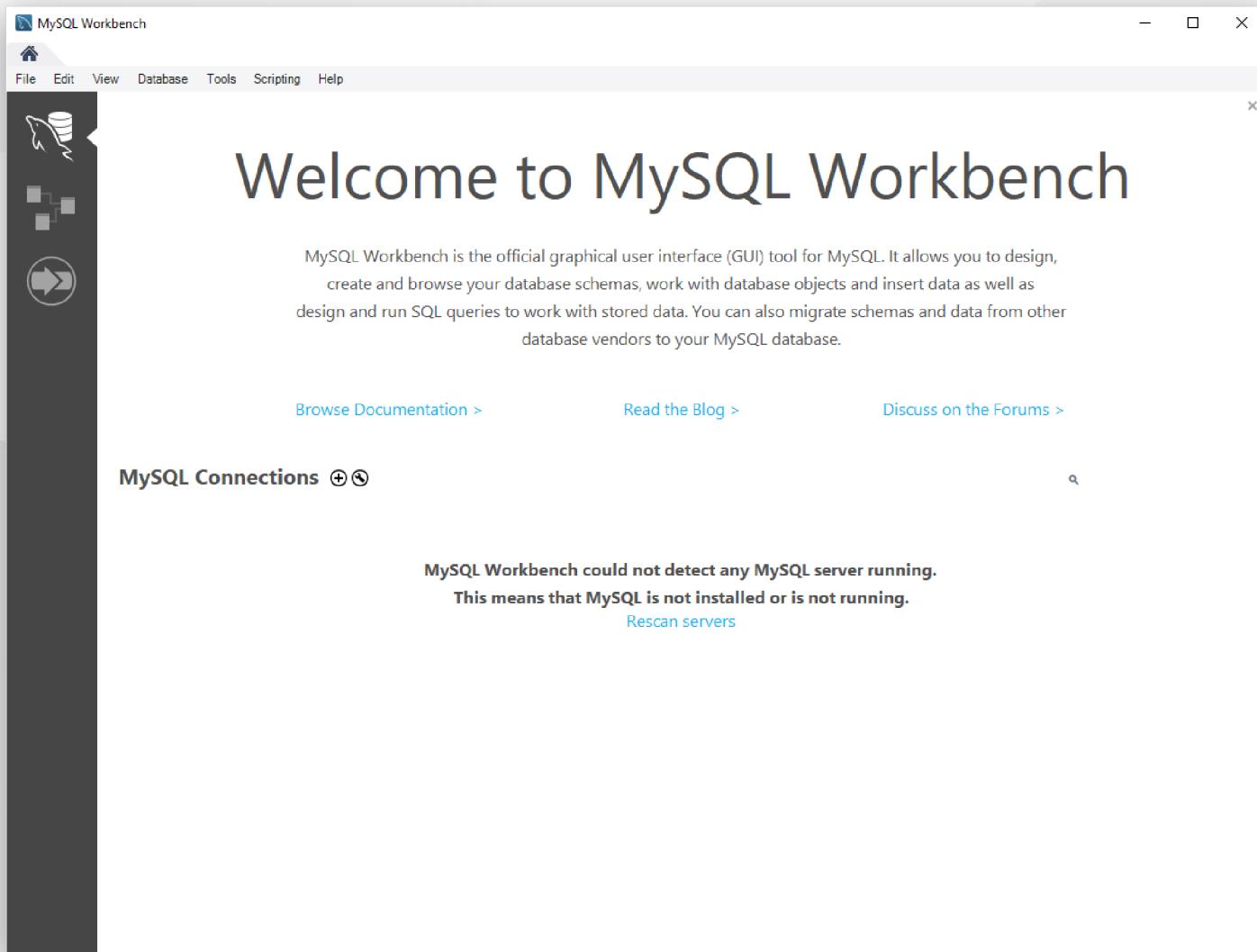


## Ejecutando MySQL Workbench

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

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

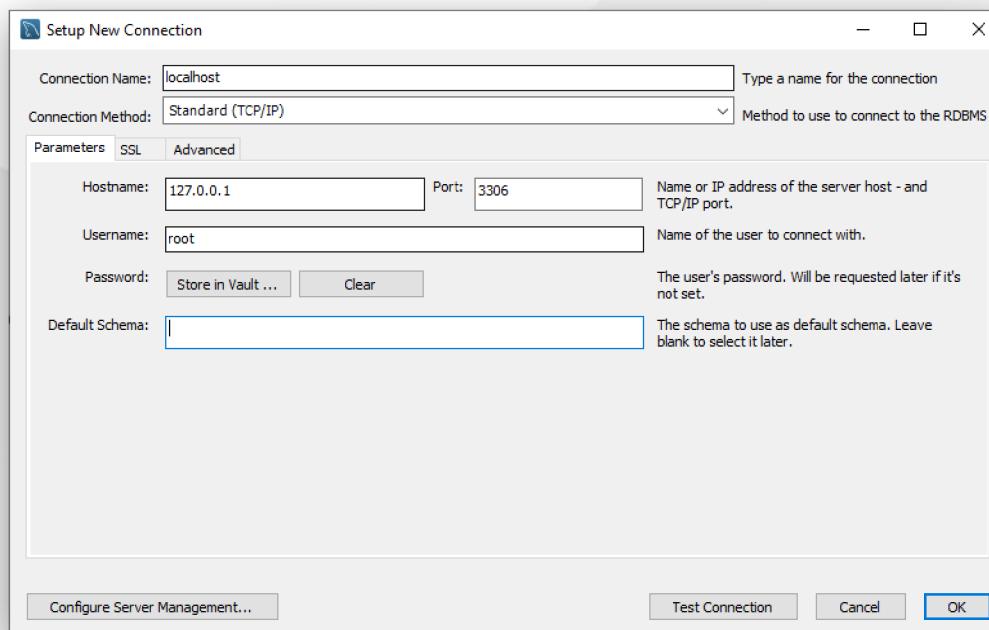
# Ejecutando MySQL Workbench



# 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.



## 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.

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the connection to "localhost" and various management, instance, and performance metrics.
- Query Editor (Query 1):** Displays the SQL code for creating a schema and three tables:

```
1 • CREATE SCHEMA formula1
2   DEFAULT CHARACTER SET utf8
3   COLLATE utf8_spanish2_ci;
4
5 • USE formula1;
6
7 • ◇ CREATE TABLE status (
8   statusId INT NOT NULL,
9   status VARCHAR(250) DEFAULT NULL,
10  PRIMARY KEY (statusId)
11 );
12
13 • ◇ CREATE TABLE circuits (
14   circuitId INT NOT NULL,
15   circuitRef VARCHAR(250) DEFAULT NULL,
16   name VARCHAR(250) DEFAULT NULL,
17   location VARCHAR(250) DEFAULT NULL,
18   country VARCHAR(250) DEFAULT NULL,
19   PRIMARY KEY (circuitId)
20 );
21
22 • ◇ CREATE TABLE constructors (
23   constructorId INT NOT NULL,
24   name VARCHAR(250) DEFAULT NULL,
```
- SQLAdditions:** A note stating "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."
- Output:** An empty pane for displaying the results of the executed queries.

# 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 left sidebar displays the 'SCHEMAS' tree, with 'formula1' selected, showing tables like circuits, constructors, drivers, pitstops, qualifying, races, results, status, views, stored procedures, and functions. The main area shows a 'Query 1' tab with the query `SELECT * FROM formula1.drivers;`. Below it is a 'Result Grid' displaying 16 rows of driver data:

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	Sébastien	Bourdais	1979-02-28	French
8	7	RAI	Kimi	Räikkönen	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

The bottom pane shows the 'Output' window with log entries for the executed queries. The last entry is:

```
62 10:06:18 SELECT * FROM formula1.drivers LIMIT 0, 1000 842 row(s) returned 0.000 sec / 0.000 sec
```

# Licencia

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

Puede encontrar su código en el siguiente enlace:  
<https://github.com/bbddetsisi/material-docente>