



Programación Java

► Configuración MySQL Local

// Práctica de Configuración MySQL-Spring Boot

Objetivos

- Configurar una aplicación Spring Boot con conexión a una Base de Datos MySQL H2 en memoria.
- Crear una base de datos MySQL en forma local.
- Crear un nuevo usuario en una base de datos local.
- Configurar una aplicación Spring Boot con conexión a una Base de Datos MySQL local física.

¡Buena suerte!





Ejercicio 1

1. Crear una aplicación Spring Boot "**SpoiledTomatoesAPI**".
2. Agregar las dependencias necesarias para levantar una base de datos MySQL H2 en memoria, en el archivo **pom.xml**.

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-data-jpa</artifactId>
</dependency>

<dependency>
  <groupId>com.h2database</groupId>
  <artifactId>h2</artifactId>
  <scope>runtime</scope>
</dependency>
```

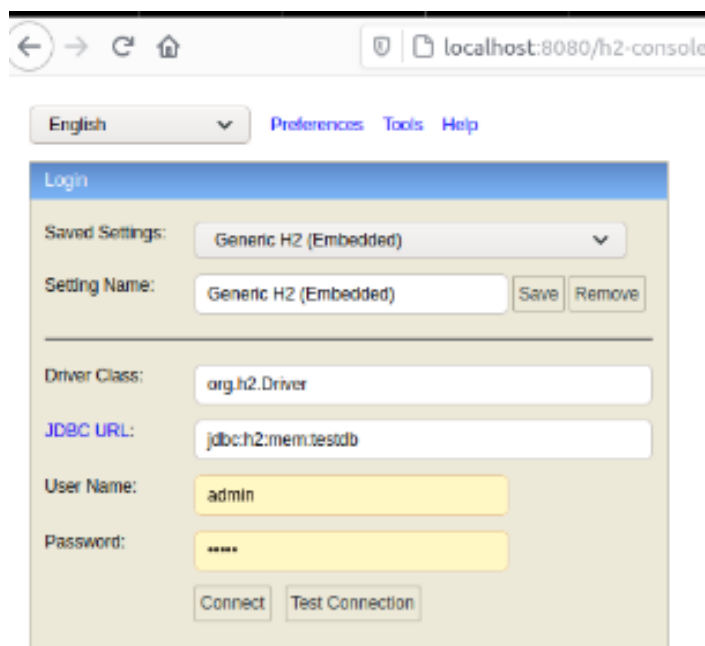
3. Agregar los parámetros necesarios en el archivo de configuración **application.properties**.

```
spring.jpa.hibernate.ddl-auto=update
spring.datasource.driverClassName=org.h2.Driver
spring.datasource.url=jdbc:h2:mem:testdb
spring.datasource.username=admin
spring.datasource.password=admin
spring.jpa.database-platform=org.hibernate.dialect.H2Dialect
spring.h2.console.enabled=true
```



4. Levantar la aplicación y comprobar que está funcionando el servicio h2
<http://localhost:8080/h2-console>.

5. Conectarse a la consola de h2 empleando los parámetros establecidos en el archivo application.properties.



6. Crear una clase MiniSerie dentro del paquete "**model**", que tenga los atributos Long id, String name, Double rating e Integer awards.

7. Para la clase del Ej. 6, generar sus métodos Getters y Setters y anotarla con **@Entity**.

8. En la clase del Ej. 6, anotar el atributo id con **@Id** y **@GeneratedValue**.

9. Crear una interfaz **IMiniserieRepository** en el paquete "repository" y anotarla con **@Repository**.

10. Hacer que la interfaz del EJ. 9 extienda a **JpaRepository<Miniserie, Long>**.

11. Levantar la aplicación y comprobar desde la consola de H2 que se haya creado una nueva tabla con el nombre "Miniserie" y los campos de la clase Miniserie creada.



Ejercicio 2

1. Cree un nuevo usuario de MySQL con la siguiente configuración
 - o nombre de usuario: **st_admin**
 - o servidor: **localhost**
 - o contraseña: **st_admin**
 - o base de datos para el usuario: **movies_db**
 - o garantizarle todos los privilegios.
2. Una vez creado el usuario, entre como usuario **st_admin** y compruebe que tiene todos los privilegios de la base de datos **movies_db**.
3. Volver al proyecto "**SpoiledTomatoesAPI**" y agregar las dependencias necesarias para levantar una base de datos MySQL física en forma local, en el archivo **pom.xml**.

```
<dependency>
  <groupId>org.springframework.data</groupId>
  <artifactId>spring-data-jdbc</artifactId>
</dependency>
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>8.0.18</version>
</dependency>
```

3. Reemplazar los parámetros necesarios en el archivo de configuración **application.properties**.

```
spring.jpa.hibernate.ddl-auto=update
spring.datasource.driverClassName=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://localhost:3306/movies_db?useSSL=false&serverTimezone=UTC
spring.datasource.username=st_admin
spring.datasource.password=st_admin
spring.jpa.database-platform=org.hibernate.dialect.MySQL5InnoDBDialect
```

4. Levantar la aplicación y comprobar desde MySQL Workbench que se haya creado una nueva tabla con el nombre "Miniserie" y los campos de la clase Miniserie creada.