

Índice

Enunciado Tarea.....	2
Descargar el proyecto desde start.spring.io.....	3
Descargar el proyecto desde start.spring.io.....	3
Abrirlo en visual Studio Code.....	3
Crear entidades y repositorios.....	4
data.sql.....	4
schema.sql.....	5
Configurar la base de datos H2 en memoria o en archivo.....	5
Configura application.properties para trabajar con una base de datos H2 en memoria o en archivo.....	5
Ahora probamos a entrar con la URL: http://localhost:8080/h2-console	7

Enunciado Tarea

Objetivos:

En esta práctica, los alumnos desarrollarán una aplicación sencilla utilizando **Spring Boot** con una base de datos embebida **H2**. El objetivo es que aprendan a interactuar con bases de datos usando **Spring Data JPA**. Como desafío adicional, podrán mostrar los datos en una página web utilizando **Spring Web**.

Instrucciones:

1. Descargar el proyecto desde start.spring.io ➡:

- Añadir las siguientes dependencias:
 - **Spring Data JPA**
 - **H2 Database**
 - **Spring Web** (opcional para el desafío).
- Descargar el proyecto y abrirlo en un IDE como Visual Studio Code.

2. Configurar la base de datos H2 en memoria o en archivo:

- Configura `application.properties` para trabajar con una base de datos H2 en **memoria** o en **archivo**.
- Habilita la consola de H2 para probar tus consultas en la URL: <http://localhost:8080/h2-console> ➡.

3. Crear entidades y repositorios:

- Crea las entidades necesarias para interactuar con la base de datos. Puedes usar la estructura de **Usuarios y Roles** de una práctica anterior o bien crear tu propia estructura.

4. Interacción con la base de datos:

- Prueba la base de datos en memoria o en archivo, verificando que los datos se almacenan y consultan correctamente interactuando desde tu aplicación Java.

5. Desafío opcional:

- Los alumnos que quieran un reto adicional pueden integrar **Spring Web** para mostrar los datos en una página HTML.
- Crea un controlador web para mostrar los datos en un formato visual sencillo a través de una página web.

6. Probar la aplicación:

- Accede a la consola de H2 para ejecutar consultas manuales y verificar los datos: <http://localhost:8080/h2-console> ➡.
- Prueba las funcionalidades de la aplicación con la base de datos en memoria o en archivo.

7. Memoria y entrega:

- Los alumnos deben entregar una **memoria y el código** con los pasos realizados, configuraciones aplicadas y resultados obtenidos.
- Incluye capturas de pantalla o un vídeo.

8. Presentación en clase:

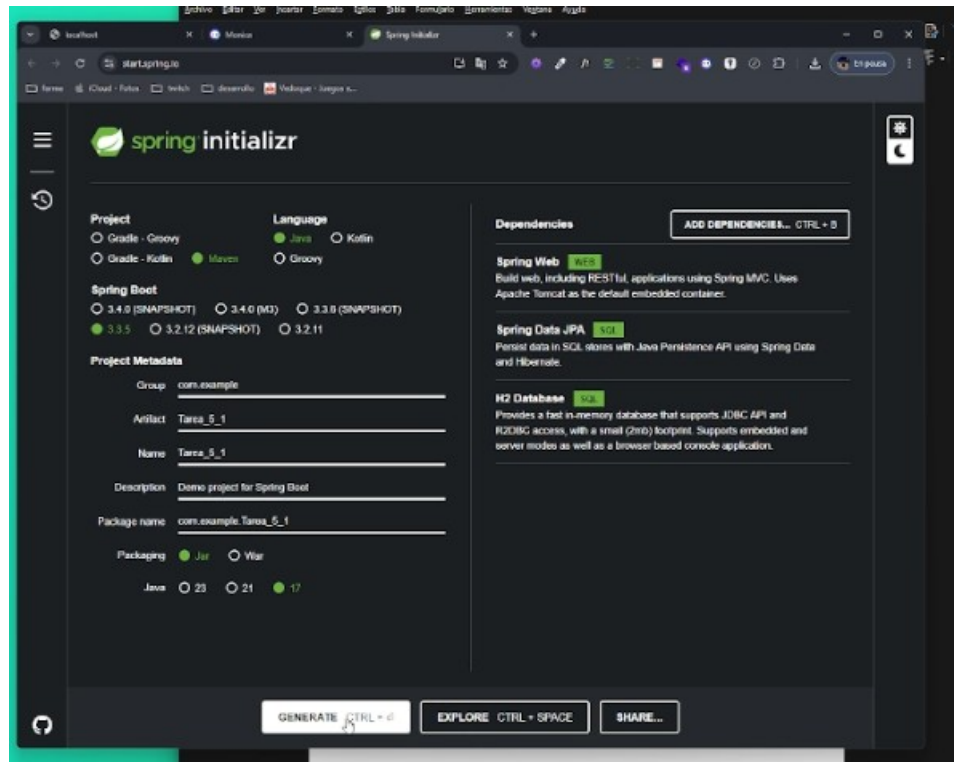
- Una vez finalizada la tarea, los alumnos deberán **demostrar el funcionamiento** de la aplicación al profesor, mostrando cómo interactúan con la base de datos y, si hicieron el desafío, cómo muestran los datos en una página web.

Nota: Los alumnos pueden utilizar la base de datos de ejemplo que contiene **Usuarios y Roles** o crear su propia base de datos como en la práctica 4.1 o cualquier otra pensada por el alumno.

Descargar el proyecto desde start.spring.io

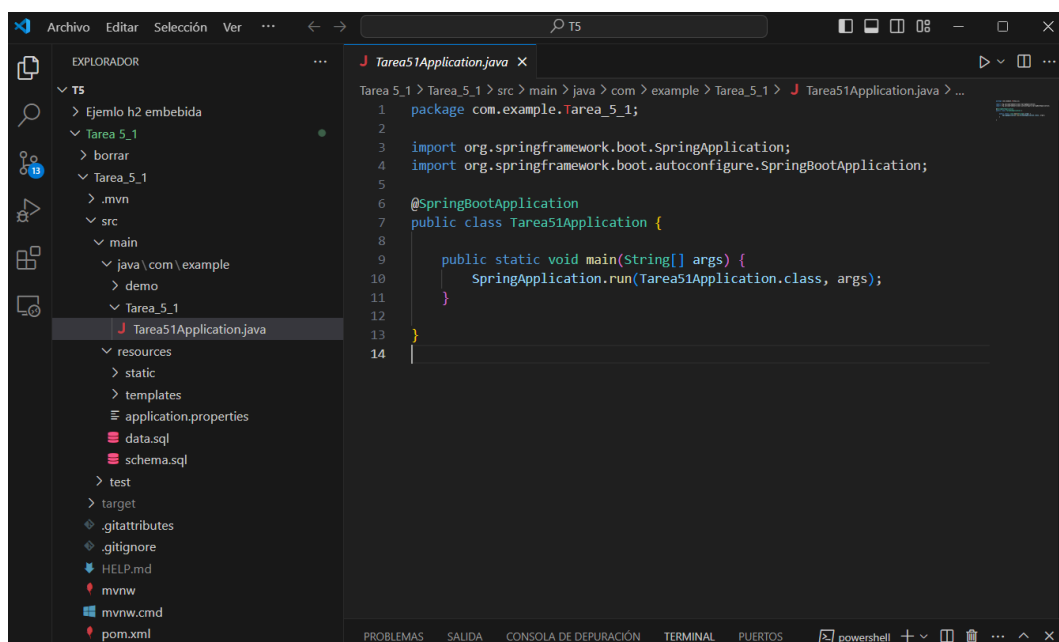
Descargar el proyecto desde start.spring.io

Elegimos las dependencias y le damos a generar. Se nos descargara un .zip

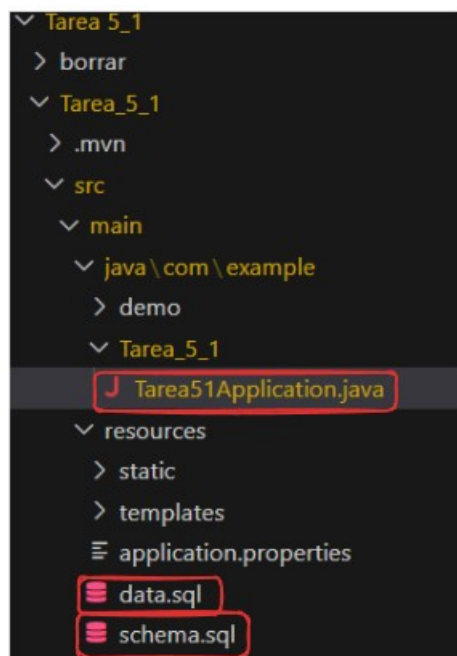


Abrirlo en visual Studio Code

Y creamos en la carpeta resources un data.sql y un schema.sql



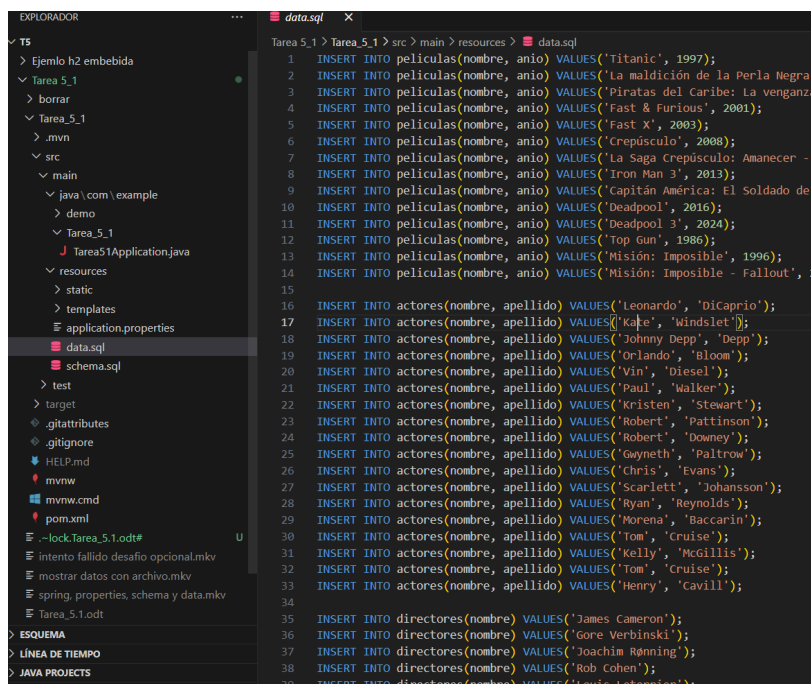
Crear entidades y repositorios



en resources creamos data.sql y schema.sql

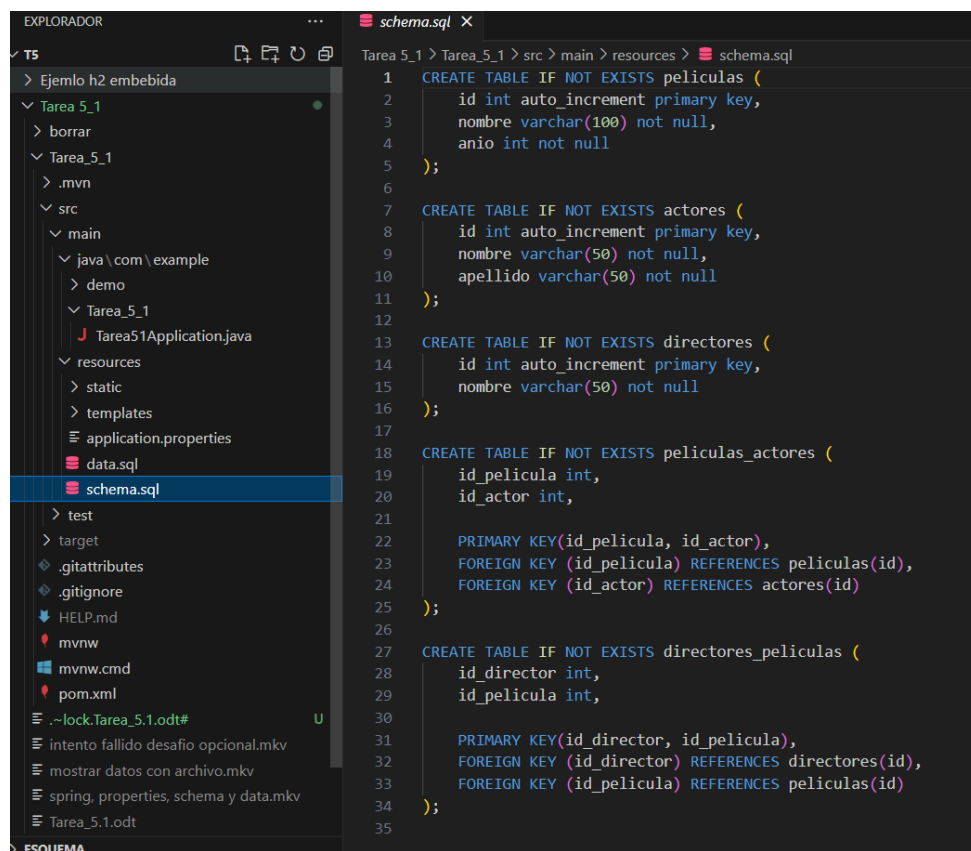
data.sql

se ponen los datos que se insertan en la base de datos. Para que funcione no podemos utilizar comillas dobles.



schema.sql

ponemos la base de datos y no funciona con el drop database if exists, ni el use database.



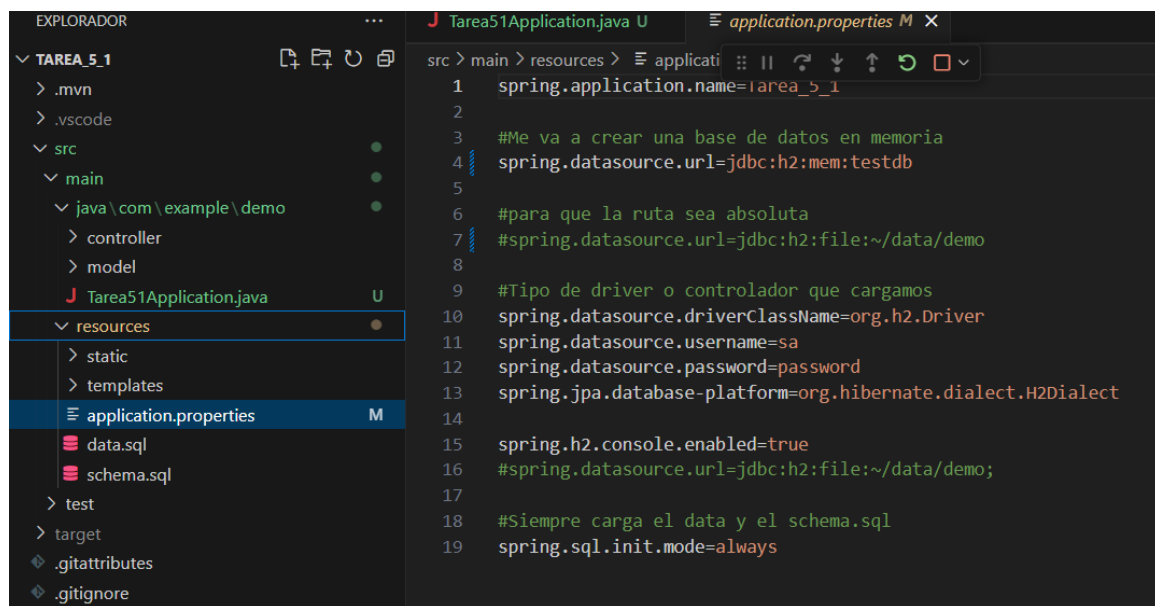
Configurar la base de datos H2 en memoria o en archivo

Configura `application.properties` para trabajar con una base de datos H2 en **memoria** o en **archivo**.

En memoria:

La ruta para tener la base de datos en **memoria ram** lo ponemos así:

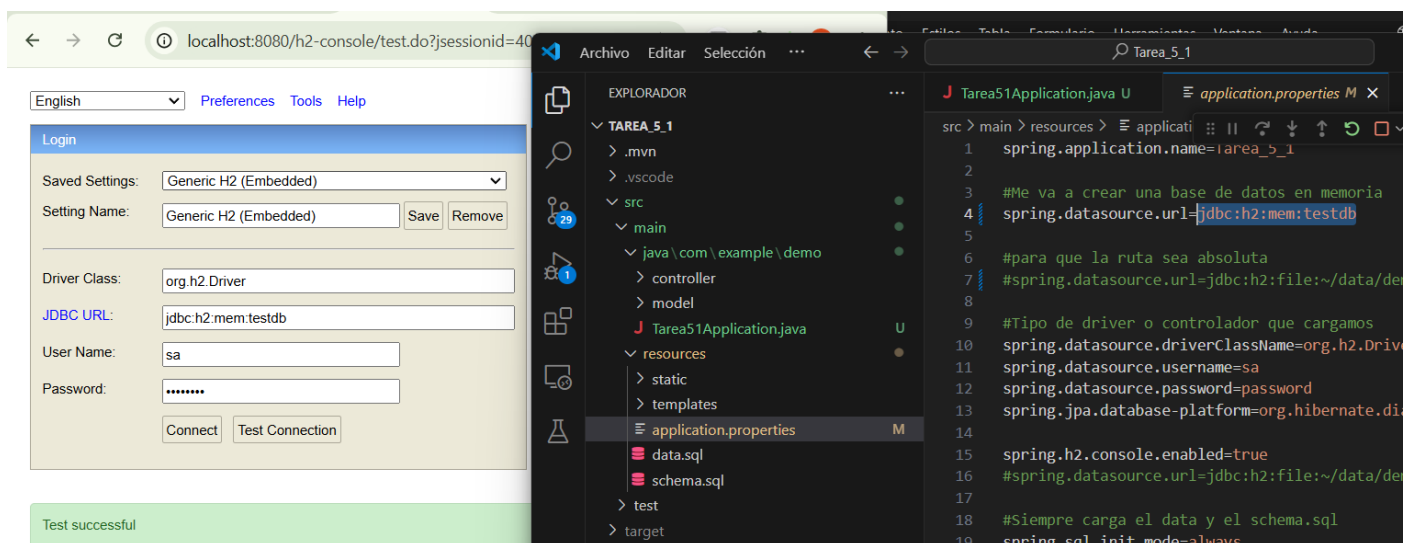
```
spring.datasource.url=jdbc:h2:mem:testdb
```



```

1  spring.application.name=area_5_1
2
3  #Me va a crear una base de datos en memoria
4  spring.datasource.url=jdbc:h2:mem:testdb
5
6  #para que la ruta sea absoluta
7  #spring.datasource.url=jdbc:h2:file:~/data/demo
8
9  #Tipo de driver o controlador que cargamos
10 spring.datasource.driverClassName=org.h2.Driver
11 spring.datasource.username=sa
12 spring.datasource.password=password
13 spring.jpa.database-platform=org.hibernate.dialect.H2Dialect
14
15 spring.h2.console.enabled=true
16 #spring.datasource.url=jdbc:h2:file:~/data/demo;
17
18 #Siempre carga el data y el schema.sql
19 spring.sql.init.mode=always
    
```

Ahora probamos a entrar con la URL: <http://localhost:8080/h2-console>.



En archivo:

Para que cree un **archivo** en una carpeta data en la carpeta de usuario lo hacemos así y donde pone demo ponemos el nombre del archivo

```
spring.datasource.url=jdbc:h2:file:~/data/demo
```

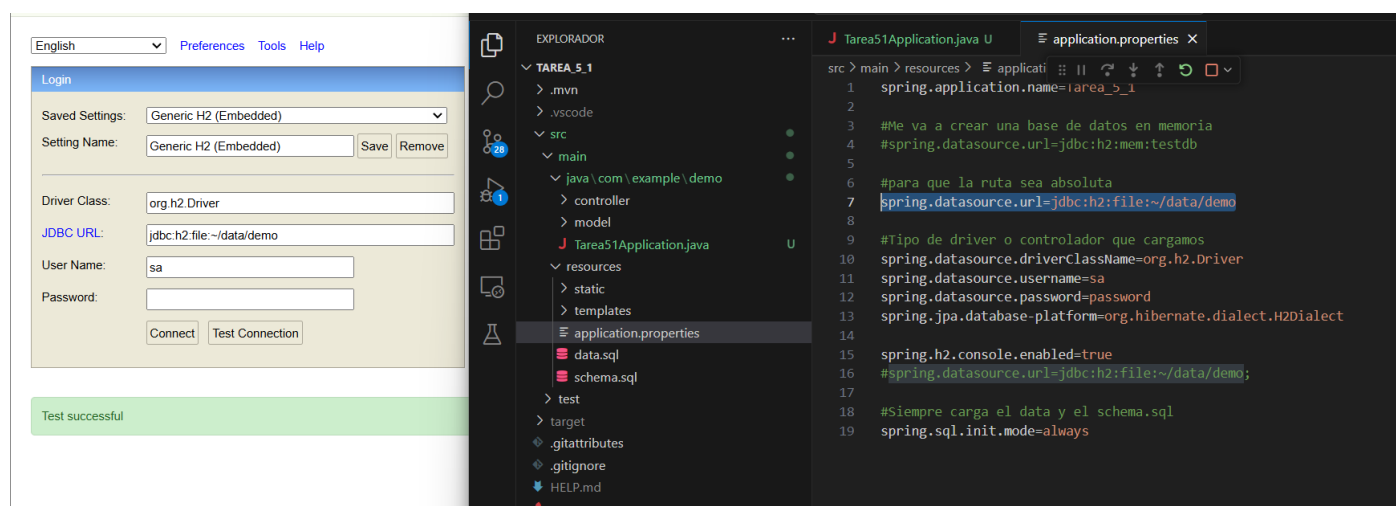
area 5.1: Aplicación Spring Boot con Base de Datos Embebida H2

```

src > main > resources > application.properties
1  spring.application.name=Tarea_5_1
2
3  #Me va a crear una base de datos en memoria
4  #spring.datasource.url=jdbc:h2:mem:testdb
5
6  #para que la ruta sea absoluta
7  spring.datasource.url=jdbc:h2:file:~/data/demo
8
9  #Tipo de driver o controlador que cargamos
10 spring.datasource.driverClassName=org.h2.Driver
11 spring.datasource.username=sa
12 spring.datasource.password=password
13 spring.jpa.database-platform=org.hibernate.dialect.H2Dialect
14
15 spring.h2.console.enabled=true
16 #spring.datasource.url=jdbc:h2:file:~/data/demo;
17
18 #Siempre carga el data y el schema.sql
19 spring.sql.init.mode=always
    
```

Ahora probamos a entrar con la URL: <http://localhost:8080/h2-console>.

Lo mismo pero ahora ponemos la ruta del fichero y comentamos la otra para hacer la prueba con el fichero



si nos vamos a C: > usuarios > Usuario > data

Comprobamos que se ha creado el archivo

