



Spring

H2





Database in-memory H2

- I database in-memory sono delle sorgenti dati che vengono spesso utilizzate per la fase di testing.
- Le tabelle di un DB in-memory vengono cancellate quando l'applicazione si arresta(comportamento di default).
- Spring Boot fornisce un supporto completo ad H2.
- H2 è configurabile tramite proprietà da applicare nell'application.properties.

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



Application properties



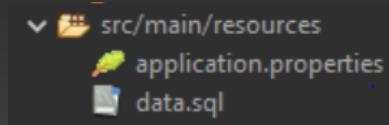
```
spring.datasource.url=jdbc:h2:mem:dbprova
spring.datasource.dbcp2.driver-class-name=org.h2.Driver
spring.datasource.username=root
spring.datasource.password=root
spring.jpa.show-sql=true
spring.jpa.database-
platform=org.hibernate.dialect.H2Dialect
spring.h2.console.enabled=true
#sarà il path utilizzato per la console del database
spring.h2.console.path=/consoleDB
#tramite quest'istruzione comunichiamo al db di eseguire
le istruzioni
#presenti nel data.sql dopo la creazione del db
spring.jpa.defer-datasource-initialization=true
```





Inizializzazione del database

- Creare un file «.sql» nella cartella «src/main/resources».



```
1 INSERT INTO RUOLO (nome) VALUES ('admin');
2 INSERT INTO RUOLO (nome) VALUES ('user');
3
4 INSERT INTO UTENTE (email,eta,nome) VALUES('matiasasis@gmail.com', 23, 'matias');
5 INSERT INTO UTENTE (email,eta,nome) VALUES('saraverdi@gmail.com', 27, 'sara');
6
7 INSERT INTO UTENTE_RUOLI(UTENTI_ID, RUOLI_ID) VALUES(1, 1);
8 INSERT INTO UTENTE_RUOLI(UTENTI_ID, RUOLI_ID) VALUES(2, 2);
```





Database in-memory H2

- I database in-memory sono delle sorgenti dati che vengono spesso utilizzate per la fase di testing.
- Le tabelle di un DB in-memory vengono cancellate quando l'applicazione si arresta(comportamento di default).
- Spring Boot fornisce un supporto completo ad H2.
- H2 è configurabile tramite proprietà da applicare nell'application.properties.

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





Database in-memory H2

- I database in-memory sono delle sorgenti dati che vengono spesso utilizzate per la fase di testing.
- Le tabelle di un DB in-memory vengono cancellate quando l'applicazione si arresta(comportamento di default).
- Spring Boot fornisce un supporto completo ad H2.
- H2 è configurabile tramite proprietà da applicare nell'application.properties.

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



Console H2



- La console per la gestione del DB è raggiungibile di default all'URL:
`http://localhost:8080/h2-console`
- Ma tramite questa configurazione:
 - E' possibile raggiungerla all'URL: `http://localhost:8080/consoleDB`

```
spring.h2.console.path=/consoleDB
```

The screenshot shows the H2 Console login interface. The URL in the browser is `localhost:8080/consoleDB/test.do?sessionid=10a67d1bd67574c22d660ff0981cefb8`. The page has a "Login" header. It contains fields for "Saved Settings" (set to "Generic H2 (Embedded)"), "Setting Name" (set to "Generic H2 (Embedded)"), and buttons for "Save" and "Remove". Below these are fields for "Driver Class" (set to "org.h2.Driver"), "JDBC URL" (set to "jdbc:h2:mem:dbprova"), "User Name" (set to "test"), and "Password" (empty). At the bottom are "Connect" and "Test Connection" buttons. A green status bar at the bottom says "Test successful".

spring
boot

Console H2



Auto commit Max rows: 1000 | Auto complete Off | Auto select On | ?

Run Run Selected Auto complete Clear SQL statement:

jdbc:h2:mem:dbprova

- RUOLO
- UTENTE
- UTENTE_RUOLI
- INFORMATION_SCHEMA
- Users

H2 2.1.214 (2022-06-13)

Important Commands

	Displays this Help Page
	Shows the Command History
	Ctrl+Enter Executes the current SQL statement
	Shift+Enter Executes the SQL statement defined by the text selection
	Ctrl+Space Auto complete
	Disconnects from the database

Sample SQL Script

Delete the table if exists	DROP TABLE IF EXISTS TEST;
Create a new table with ID and NAME columns	CREATE TABLE TEST(ID INT PRIMARY KEY, NAME VARCHAR(255));
Add a new row	INSERT INTO TEST VALUES(1, 'Hello');
Add another row	INSERT INTO TEST VALUES(2, 'World');
Query the table	SELECT * FROM TEST ORDER BY ID;
Change data in a row	UPDATE TEST SET NAME='Hi' WHERE ID=1;
Remove a row	DELETE FROM TEST WHERE ID=2;
Help	HELP ...

Run Run Selected Auto complete Clear SQL statement:

SELECT * FROM UTENTE

SELECT * FROM UTENTE;

ID	EMAIL	ETA	NOME
1	matiasasis@gmail.com	23	matias
2	saraverdi@gmail.com	27	sara

(2 rows, 1 ms)

Edit

