Spring Boot H2 Database

What is the in-memory database

In-memory databases rely on system memory as opposed to disk space for storage of data. Because memory access is faster than disk access. We use the in-memory database when we do not need to persist the data. The in-memory database is an embedded database. The in-memory databases are volatile, by default, and all stored data loss when we restart the application.

The widely used in-memory databases are **H2, HSQLDB**(HyperSQL Database)**,**and**Apache Derby.**It creates the configuration automatically.

Persistence vs. In-memory Database

The persistent database persists the data in physical memory. The data will be available even if the database server is bounced. Some popular persistence databases are [**Oracle**](https://www.javatpoint.com/oracle-tutorial)**,**[**MySQL**](https://www.javatpoint.com/mysql-tutorial)**,**[**Postgres**](https://www.javatpoint.com/postgresql-tutorial)**,** etc.

In the case of the **in-memory database,** data store in the **system memory**. It loses the data when the program is closed. It is helpful for **POC**s (Proof of Concepts), not for a production application. The widely used in-memory database is **H2.**

## What is the H2 Database

**H2** is an **embedded, open-source,**and**in-memory** database. It is a relational database management system written in [Java](https://www.javatpoint.com/java-tutorial). It is a **client/server** application. It is generally used in **unit testing**. It stores data in memory, not persisting the data on disk.

**Advantages**

* Zero configuration
* It is easy to use.
* It is lightweight and fast.
* It provides simple Configuration to switch between a real database and in-memory database.
* It supports standard SQL and JDBC API.
* It provides a web console to maintain in the database.

## Configure H2 Database

If we want to use H2 database in an application we need to add the following dependency in pom.xml file:

1. **<dependency>**
2. **<groupId>**com.h2database**</groupId>**
3. **<artifactId>**h2**</artifactId>**
4. **<scope>**runtime**</scope>**
5. **</dependency>**

After adding the dependency, we need to configure **data source URL, driver class name, username,** and **password** of H2 database. Spring Boot provide an easy way to configure these properties in **application.properties** file.

1. spring.datasource.url=jdbc:h2:mem:testdb
2. spring.datasource.driverClassName=org.h2.Driver
3. spring.datasource.username=sa
4. spring.datasource.**password**=
5. spring.jpa.**database**-platform=org.hibernate.dialect.H2Dialect

In the **spring.datasource.url** property, **mem** is the name of an in-memory database and **testdb** is the name of schema that H2 provides, by default. We can also define our own schema and database. The default username is **sa** and the blank password denotes an **empty** password. If we want to change the username and password, we can override these values.

## Persist the data in H2 Database

If we want to persist the data in the H2 database, we should store data in a file. To achieve the same, we need to change the datasource [URL](https://www.javatpoint.com/url-full-form) property.

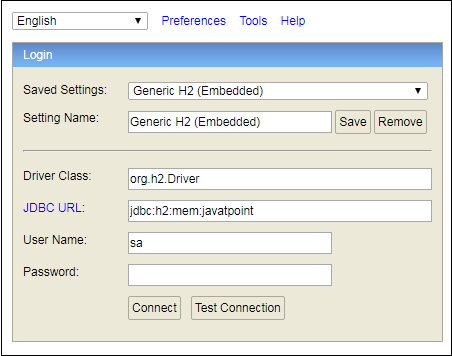
1. #persist the data
2. spring.datasource.url=jdbc:h2:file:/data/sampledata
3. spring.datasource.url=jdbc:h2:C:/data/sampledata

## H2 Console

By default, the console view of the H2 database is disabled. Before accessing the H2 database, we must enable it by using the following property.

1. #enabling the H2 console
2. spring.h2.console.enabled=**true**

Once we have enabled the H2 console, now we can access the H2 console in the browser by invoking the URL **http://localhost:8080/h2-console**. The following figure shows the console view of the H2 database.



For Example

[Spring Boot H2 Database - javatpoint](https://www.javatpoint.com/spring-boot-h2-database)