Install Java latest version

Download Maven

Setup Environment variable:

JAVA\_HOME -> C:\Program Files\Java\jdk-18.0.1.1

MAVEN\_HOME -> C:\Users\cgandhi\Projects\aptos\desktop\apache-maven-3.8.5

Path ->

%JAVA\_HOME%\bin

%MAVEN\_HOME%\bin

Close and start cmd again, run -> mvn --version

Install IntelliJ Idea Community version

Set environment variable

Install MySQL community server and MySQL Workbench

Go to <https://start.spring.io/> and generate spring boot project

Run project from cmd

Go to porm.xml folder location

Run -> mvn spring-boot:run

To add dependency in porm.xml

Go to <https://start.spring.io/>

Click on add dependencies button, search for dependency to add and click on explore

Copy dependency from porm.xml file

[spring boot hotswap with Intellij IDE](https://stackoverflow.com/questions/23155244/spring-boot-hotswap-with-intellij-ide)

Adding devtools to your project

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

</dependency>

Enabling automatic build

Open the Settings --> Build-Execution-Deployment --> Compiler and enable

Build Project Automatically

Update the value of compiler.automake.allow.when.app.running

Press ctrl+shift+A and search for the registry. In the registry, enable

compiler.automake.allow.when.app.running

Open the settings -> Advanced Settings and enable

Allow auto-make to start even if developed application is currently running

Enable h2 database

Add below dependencies in application.properties

<dependency>

<groupId>com.h2database</groupId>

<artifactId>h2</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

Add below properties in application.properties

spring.h2.console.enabled=true

spring.datasource.url=jdbc:h2:mem:dcbapp

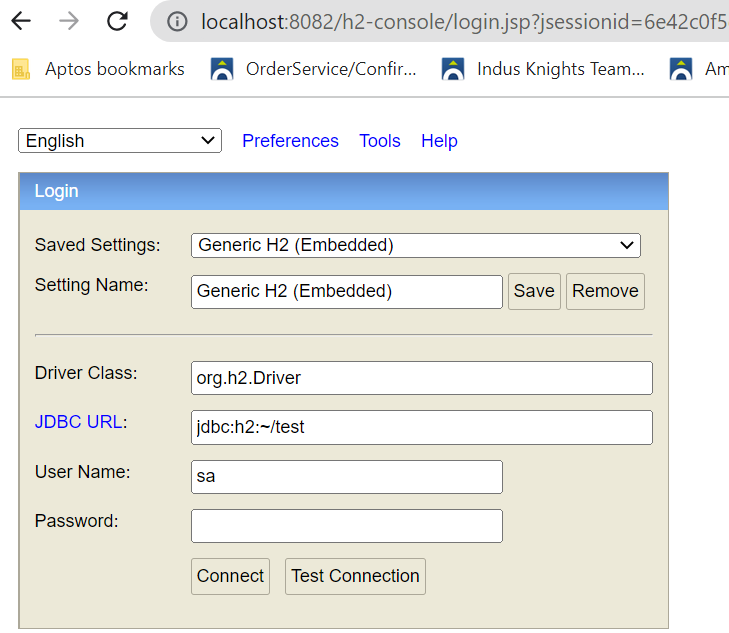
spring.datasource.driverClassName=org.h2.Driver

spring.datasource.username=sa

spring.datasource.password=password

spring.jpa.database-platform=org.hibernate.dialect.H2Dialect

Run application and browse localhost:port/h2-console



Add controller endpoints to Add, Get, Update, Delete

We will create entity, service, and repository

Entity will be annoted with @Entity

Service will be annoted with @Service

Repository will be annoted with @Repository and extends JpaRepository class

Add below in porm.xml

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

JpaRepository has methods to do basic save, get methods

We can add new methods in JpaRepository like fetchByDepartmentName, findByDepartmentNameIgnore and even write sql query, need to follow jpql spring boot documentation)

We can add validation to entity like @NotBlank,….

Add below in porm.xml

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-validation</artifactId>

</dependency>

We can remove properties getter and setter, Consturctors, tostring methods from entity using Lombok plugin

Add below in porm.xml

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<optional>true</optional>

</dependency>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

<configuration>

<excludes>

<exclude>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

</exclude>

</excludes>

</configuration>

</plugin>

Install Plugins

File -> Settings -> Plugins -> Search Lombok and install

Now annote entity with @Data, @NoArgsConstructor, @AllArgsConstructor, @builder

Spring uses slf4j for logging

In class

private final Logger LOGGER =

LoggerFactory.getLogger(DepartmentController.class);

In Method

LOGGER.info("Inside fetchDepartmentList of DepartmentController");

Controller handle exception

Create class and extends ResponseEntityExceptionHandler

Annoted class with @ControllerAdvice, @ResponseStatus