A Quick Introduction on how to create Web Applications With Spring Boot

1. What is Spring Boot?

Spring Boot is an awesome tool that makes it easy to create stand-alone, production-grade Spring-based applications that are self-contained and can be deployed and run with minimal configuration. It is an opinionated package of the Spring platform, along with third-party libraries, which makes it possible to setup and build a complex enterprise project with much less effort/fuss. Spring Boot can be used to create Java applications that are packaged as a self-contained (with embedded tomcat, jetty or undertow web servlet container) .jar file, which can then be executed by running the command, *java -jar theProjectName.jar* or packaged as the traditional .war file and deployed to a Java Servlet web container or a full-blown App server.

2. **Installing Spring Boot**:

One of the quickest/easiest options for installing and getting started with Spring Boot is to download and setup the Spring Boot CLI (Command Line Interface), from https://repo.spring.io/release/org/springframework/boot/spring-boot-cli/. To set it up, simply unzip the package into a folder of your choice, (e.g. C:\spring\springboot) and then add a SPRING_HOME environment variable, pointing to the spring-boot-cli folder (e.g. c:\spring\springboot) and also add an entry for %SPRING_HOME%\bin to the Path environment variable. To test/verify the installation, simply open a Command/terminal window and execute, spring --version. This should display the version information, indicating the Spring Boot CLI is installed and ready for use.

3. **Using the Spring Boot CLI to create a Spring Web Application**:

3.1. Create a new folder for your applications e.g. c:\spring\myapps.

3.2. Cd into the new folder and execute the following command:

```
c:\spring\myapps> spring init
--dependencies=web,data-jpa,h2,mail,security,session,thymeleaf
myspringbootwebapp1
```

This command causes the spring boot cli to initialize a new spring application, using the Spring Initializr webtool (found online at https://start.spring.io), and have it ready with all the dependencies specified, as an Apache Maven project (which is the default project build type. Gradle type projects can also be created).

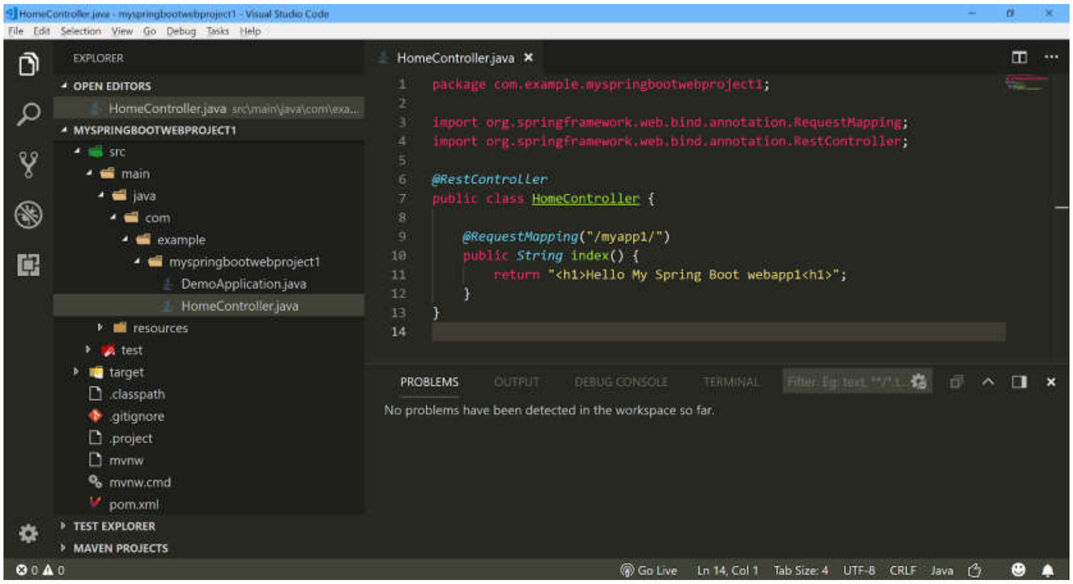
3.3. Next, cd into the 'myspringbootwebapp1' project folder and load the project source code into an IDE or Code editor of your choice e.g. VSCode or Eclipse or Netbeans or IntelliJ IDEA. To load it into VSCode, simply run -

```
c:\spring\myapps\myspringbootwebapp1> code .
```

at the command prompt.

3.4. Add a new Java source file to the package named, HomeController.java; and in it enter the following source code:

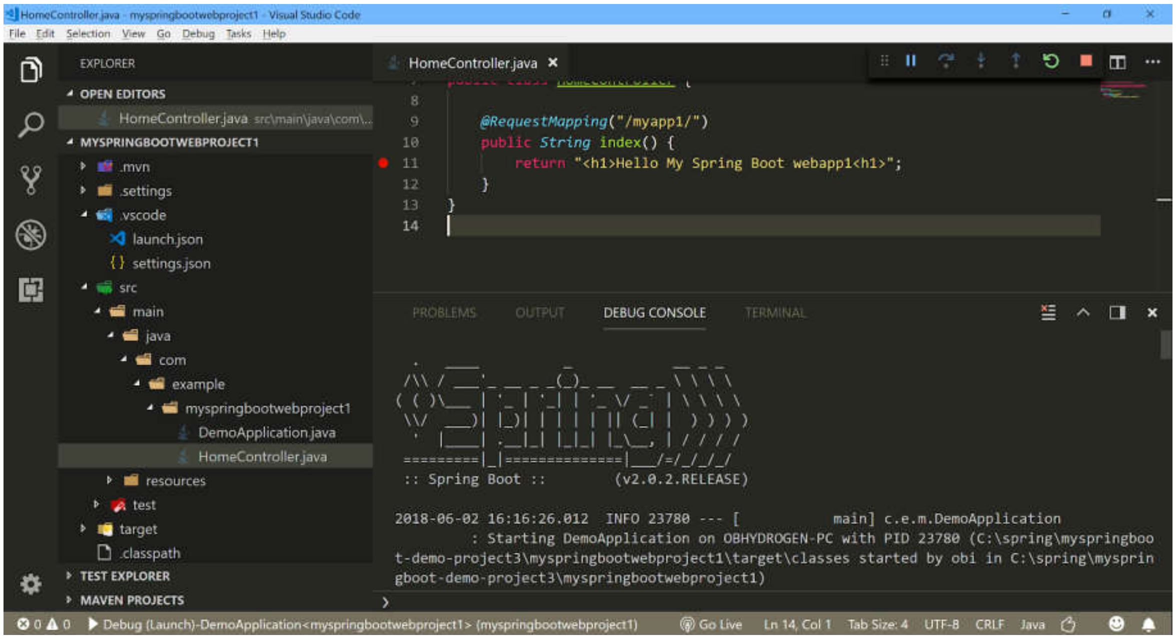
```java
package com.example.myspringbootwebproject1;

import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HomeController {

    @RequestMapping("/myapp1/")
    public String index() {
        return "Hello My Spring Boot webapp1";
    }
}
```



```java
package com.example.myspringbootwebproject1;

import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HomeController {

    @RequestMapping("/myapp1/")
    public String index() {
        return "<h1>Hello My Spring Boot webapp1<h1>";
    }
}
```

No problems have been detected in the workspace so far.

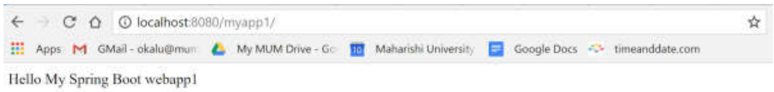
3.5. To test run the application, create a new Java launch configuration and select and run it. This will launch the spring app and start the embedded web container.



```
    @RequestMapping("/myapp1/")
    public String index() {
        return "<h1>Hello My Spring Boot webapp1<h1>";
    }
}
```

```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::        (v2.0.2.RELEASE)

2018-06-02 16:16:26.012  INFO 23780 --- [           main] c.e.m.DemoApplication
         : Starting DemoApplication on OBHYDROGEN-PC with PID 23780 (C:\spring\myspringboo
t-demo-project3\myspringbootwebproject1\target\classes started by obi in C:\spring\mysprin
gboot-demo-project3\myspringbootwebproject1)
```

3.6. To checkout the webapp, go to the http://localhost:8080/myapp1/ url.



Hello My Spring Boot webapp1

3.7. gfgfgf

3.8. gdfdfd

4. cchggkj

// -- End --//