# Lab S4: Spring Boot

## Objectives

In this lab you will use the Spring Initializr to create a demo Hello World Application

## Lab Setup

Because we are going to using the Maven build tool, ensure that it is installed and configured before doing the lab. You can confirm this by opening a command window and typing “mvn -version”.

Text

Description automatically generatedYou should see the following or something similar – the actual version and pathname will probably be different

## Creating the App with Spring Initializr

### Step 1: Use the Sprint Start Page

1. Go to the web page <https://start.spring.io>
2. Graphical user interface, application

   Description automatically generatedYou should see something like this:
3. The defaults “Java” and “Maven” should be selected as well as the highest numbered version that does *not* have (SNAPSHOT) or (M\*) following.
4. Ensure that you have selected the “Jar” packaging and the highest Java version that is less than yours (i.e. if you are running Java 12, then select Java 11.

### Step 2: Adding the Dependencies

1. Graphical user interface, text, application, email

   Description automatically generatedSelect the dependencies box and then choose the Spring Web starter from the dropdown list

### Step 3: Generate the App

1. Click on “Generate” and download the resulting zip file.

Graphical user interface, text, application

Description automatically generated

## Import the Project in Eclipse

In this step, you will take the generated Maven project and import it into Eclipse to build the application.

Remember, Spring Boot just did all the configuration to create a Web App, but we still have to actually write the App code. This will be done in future sections on building Web Services and Web Apps.

### Step 1: Unzip the project

1. Move the zip file to a working directory (whatever you want) and unzip the archive
2. Graphical user interface, application

   Description automatically generatedThe resulting Maven project should look like this:

### Step 2: Import as a Maven project

1. Graphical user interface, text, application

   Description automatically generatedIn Eclipse, chose “Import” from the main File menu and select Maven Project and then “Import Existing Maven Project:.
2. In the File selector, chose the directory of your downloaded project and then select finish.
3. It will take some time to download all the Maven dependencies and build the project, but you can track the progress in the lower right hand corner:

Table

Description automatically generated

1. Graphical user interface, application, Word

   Description automatically generatedOnce Maven is finished, the project should look like this:
2. You can check in the Maven dependencies to see that a number of Spring Boot jars are now part of the project.

Text

Description automatically generated

## Running the App

Remember that there is no actual code for the App to run, we still have to do some work ourselves.

1. Open the DemoApplication.java file.

Graphical user interface, text, application

Description automatically generated

1. Just as in previous labs, run this as a Java application, *do not run it on a server since Spring Boot has bundled a server as part of the Application.*
2. You should see in the console the startup message

Text

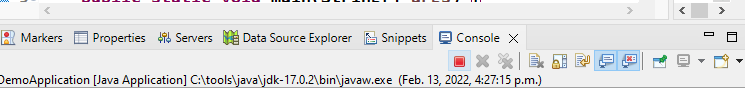
Description automatically generated

1. Text, letter

   Description automatically generatedYou should also be able to see (over to the right) messages about starting up the server and serving up the application on port 8080.
2. Going to the webpage, we can see that the application is running but saying that there is no actual web content, including an error page.

Graphical user interface, text, application, email

Description automatically generated

1.  Shut down the application by pressing the red “terminate” button on the console toolbar