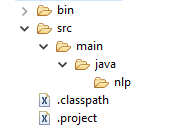
Development Environment:

1. Base: Eclipse Java EE IDE for Web Developers. Version: Mars.1 Release (4.5.1)
2. Spring STS 3.7.1: update the Eclipse base version with all available components in: <http://dist.springsource.com/release/TOOLS/update/e4.5/>
3. JDK used: Oracle 1.8.0\_05

Development:

1. Create Spring Boot application:
   1. Create a Java project in Eclipse
   2. Create folder structure:



* 1. Create Maven pom.xml file [content see source code]

The [Spring Boot Maven plugin](https://github.com/spring-projects/spring-boot/tree/master/spring-boot-tools/spring-boot-maven-plugin) provides many convenient features:

* It collects all the jars on the classpath and builds a single, runnable "über-jar", which makes it more convenient to execute and transport your service.
* It searches for the public static void main() method to flag as a runnable class.
* It provides a built-in dependency resolver that sets the version number to match [Spring Boot dependencies](https://github.com/spring-projects/spring-boot/blob/master/spring-boot-dependencies/pom.xml). You can override any version you wish, but it will default to Boot’s chosen set of versions.
  1. Add Java classes:

src/main/java/nlp/Greeting.java

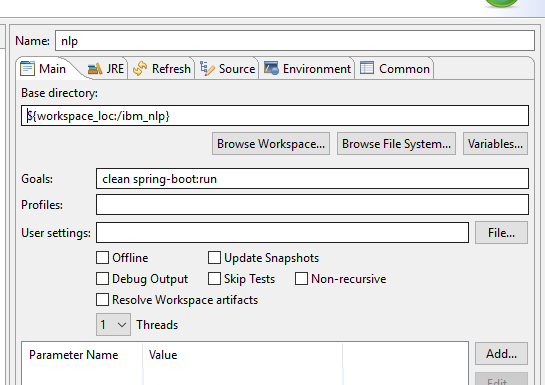
src/main/java/nlp/GreetingController.java src/main/java/nlp/Application.java

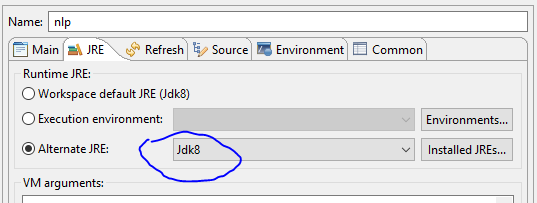
* 1. Convert to Maven project:

Right Click->Configure->Convert to Maven Project

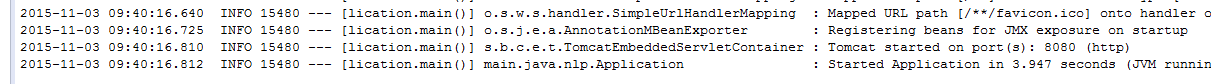
* 1. Configure Build:

Right click->Run As->Run Configuration to configure:

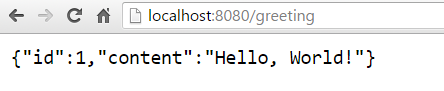




* 1. Run the configuration:



The application will be up and running:



* 1. If the started instance cannot be closed automatically, do something like the following:

C:\Users\jizhen>netstat -ano | find "8080"

TCP 0.0.0.0:8080 0.0.0.0:0 LISTENING 21476

TCP [::]:8080 [::]:0 LISTENING 21476

TCP [::1]:8080 [::1]:65236 TIME\_WAIT 0

C:\Users\jizhen>taskkill /F /PID 21476

SUCCESS: The process with PID 21476 has been terminated.

* **Appendix 1:**

To build a war file that is both executable and deployable into an external container you need to mark the embedded container dependencies as “provided”, e.g:

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

*<!-- ... -->*

<packaging>war</packaging>

*<!-- ... -->*

<dependencies>

<dependency>

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

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

</dependency>

<dependency>

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

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

<scope>provided</scope>

</dependency>

*<!-- ... -->*

</dependencies>

</project>