

« Spring Boot Web JSP: Bootstrap »

30/04/2018

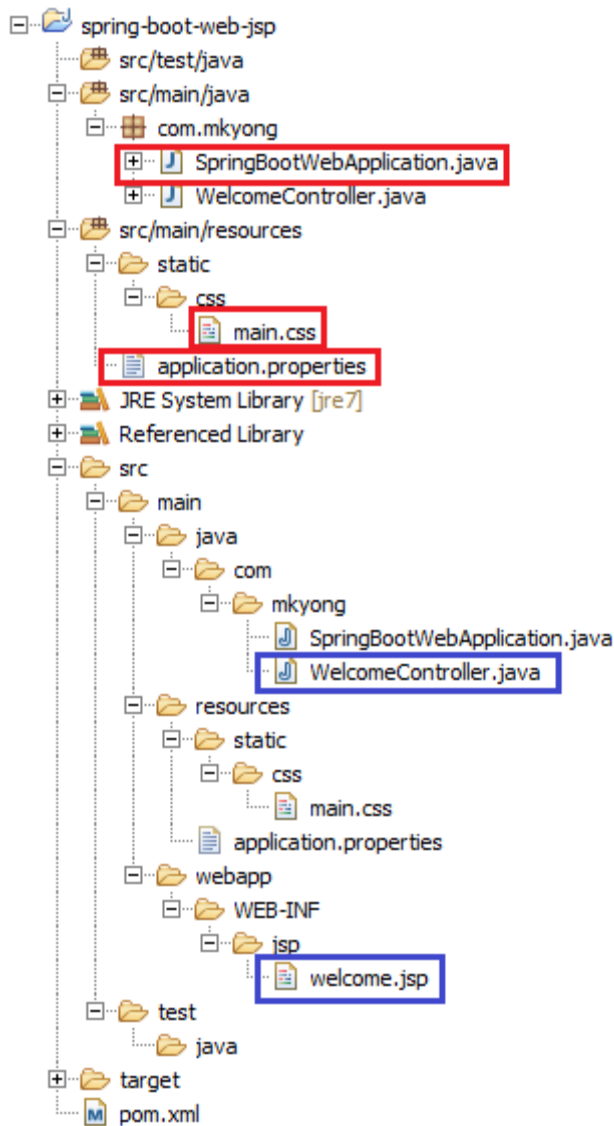
By : REGRAG mouhcine

Table des matières

1- Arborescence du projet	3
3- Exécution de l'exemple	4
4- Travail à faire	5
5- Fichiers sources	9

1- Arborescence du projet :

La structure de base d'un projet spring boot est comme suit :



On trouve les fichiers suivants :

- Pour la configuration : application.properties, SpringBootWebApplication.java.
- Pour le contrôleur : WelcomeContoller.java.
- Pour les vues : welcome.jsp

Il faut respecter cette organisation, en créant le package :

com.mkyong

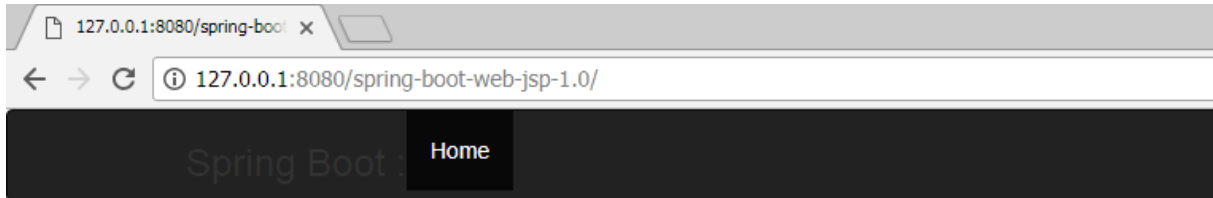
La bibliothèque utilisateur nommé ici Referenced Library doit contenir ce qui suit :

-  jquery-1.11.1.jar - C:\Users\hp\.m2\repository\org\jquery\jquery\1.11.1
-  bootstrap-3.3.7.jar - C:\Users\hp\.m2\repository\org\bootstrap\bootstrap\3.3.7
-  ecj-4.6.1.jar - C:\Users\hp\.m2\repository\org\ecj\ecj\4.6.1
-  tomcat-embed-jasper-8.5.6.jar - C:\Users\hp\.m2\repository\org\apache\tomcat\embed\tomcat-embed-jasper\8.5.6
-  spring-webmvc-4.3.4.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\spring-webmvc\4.3.4.RELEASE
-  spring-web-4.3.4.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\spring-web\4.3.4.RELEASE
-  jackson-core-2.8.4.jar - C:\Users\hp\.m2\repository\com\fasterxml\jackson\core\jackson-core\2.8.4
-  jackson-annotations-2.8.4.jar - C:\Users\hp\.m2\repository\com\fasterxml\jackson\core\jackson-annotations\2.8.4
-  jackson-databind-2.8.4.jar - C:\Users\hp\.m2\repository\com\fasterxml\jackson\core\jackson-databind\2.8.4
-  classmate-1.3.3.jar - C:\Users\hp\.m2\repository\com\fasterxml\classmate\1.3.3
-  jboss-logging-3.3.0.Final.jar - C:\Users\hp\.m2\repository\org\jboss\logging\jboss-logging\3.3.0.Final
-  hibernate-validator-5.2.4.Final.jar - C:\Users\hp\.m2\repository\org\hibernate\hibernate-validator\5.2.4.Final
-  tomcat-embed-websocket-8.5.6.jar - C:\Users\hp\.m2\repository\org\apache\tomcat\embed\tomcat-embed-websocket\8.5.6
-  tomcat-embed-el-8.5.6.jar - C:\Users\hp\.m2\repository\org\apache\tomcat\embed\tomcat-embed-el\8.5.6
-  tomcat-embed-core-8.5.6.jar - C:\Users\hp\.m2\repository\org\apache\tomcat\embed\tomcat-embed-core\8.5.6
-  spring-boot-starter-tomcat-1.4.2.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\boot\spring-boot-starter-tomcat\1.4.2.RELEASE
-  snakeyaml-1.17.jar - C:\Users\hp\.m2\repository\org\yaml\snakeyaml\1.17
-  log4j-over-slf4j-1.7.21.jar - C:\Users\hp\.m2\repository\org\log4j\log4j-over-slf4j\1.7.21
-  jul-to-slf4j-1.7.21.jar - C:\Users\hp\.m2\repository\org\slf4j\jul-to-slf4j\1.7.21
-  jcl-over-slf4j-1.7.21.jar - C:\Users\hp\.m2\repository\org\slf4j\jcl-over-slf4j\1.7.21
-  slf4j-api-1.7.21.jar - C:\Users\hp\.m2\repository\org\slf4j\slf4j-api\1.7.21
-  logback-core-1.1.7.jar - C:\Users\hp\.m2\repository\ch\qos\logback\logback-core\1.1.7
-  logback-classic-1.1.7.jar - C:\Users\hp\.m2\repository\ch\qos\logback\logback-classic\1.1.7
-  spring-boot-starter-logging-1.4.2.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\boot\spring-boot-starter-logging\1.4.2.RELEASE
-  spring-boot-autoconfigure-1.4.2.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\boot\spring-boot-autoconfigure\1.4.2.RELEASE
-  spring-expression-4.3.4.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\spring-expression\4.3.4.RELEASE
-  spring-beans-4.3.4.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\spring-beans\4.3.4.RELEASE
-  spring-aop-4.3.4.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\spring-aop\4.3.4.RELEASE
-  spring-context-4.3.4.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\spring-context\4.3.4.RELEASE
-  spring-core-4.3.4.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\spring-core\4.3.4.RELEASE
-  spring-boot-1.4.2.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\boot\spring-boot\1.4.2.RELEASE
-  spring-boot-starter-1.4.2.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\boot\spring-boot-starter\1.4.2.RELEASE
-  spring-boot-starter-web-1.4.2.RELEASE.jar - C:\Users\hp\.m2\repository\org\springframework\boot\spring-boot-starter-web\1.4.2.RELEASE
-  jstl-1.2.jar - C:\Users\hp\.m2\repository\javax\servlet\jstl\1.2
-  validation-api-1.1.0.Final.jar - C:\Users\hp\.m2\repository\javax\validation\validation-api\1.1.0.Final

2- Exécution de l'exemple :

L'exemple consiste à réaliser un premier demo avec spring boot, qui permet d'afficher un message dans une Page web :

La page welcome.jsp c'est par défaut la page d'accueil du projet :



Spring Boot Web JSP Example

Message: Hello world

3- Travail à faire :

a- Allez à la page welcome.jsp et ajouter le code :

```
<li><a href="about">About</a></li>
```

Comme suit :

```
<nav class="navbar navbar-inverse">
  <div class="container">
    <div class="navbar-header">
      <h2>Spring Boot</h2>
    </div>
    <div id="navbar" class="collapse navbar-collapse">
      <ul class="nav navbar-nav">
        <li class="active"><a href="#">Home</a></li>
        <li><a href="about">About</a></li>
      </ul>
    </div>
  </div>
</div>
```

b- Allez au fichier du contrôleur `WelcomeController.java` et ajouter le code suivant :

```
@RequestMapping("/about")
public String about(Map<String, Object> model, @RequestParam(value="msg",
required=false, defaultValue="World") String msg) {
    model.put("msg", this.msg);
    return "about";
}

@RequestMapping("/hello")
public String welcome(Map<String, Object> model,
@RequestParam(value="message", required=false, defaultValue="World") String
message) {
    model.put("message", this.message);
    return "welcome";
}
```

En déclarant la variable msg comme suit :

```
private String msg="This is about";
```

c- Créer une page about.jsp :

Avec le code suivant :

```
<!DOCTYPE html>
<%@ taglib prefix="spring" uri="http://www.springframework.org/tags"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<html lang="en">
<head>

<link rel="stylesheet" type="text/css"
href="webjars/bootstrap/3.3.7/css/bootstrap.min.css" />

<!--
<spring:url value="/css/main.css" var="springCss" />
<link href="${springCss}" rel="stylesheet" />
```

```

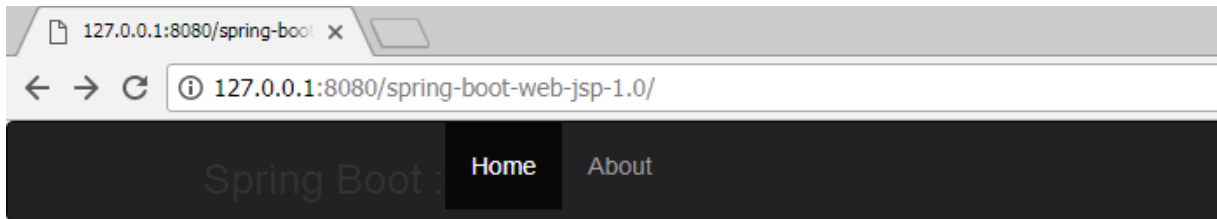
-->
<c:url value="/css/main.css" var="jstlCss" />
<link href="{jstlCss}" rel="stylesheet" />
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>About</title>
</head>
<body>
<nav class="navbar navbar-inverse">
  <div class="container">
    <div class="navbar-header">
      <h3>Spring Boot :</h3>
    </div>
    <div id="navbar" class="collapse navbar-collapse">
      <ul class="nav navbar-nav">
        <li><a href="1.0/hello">Home</a></li>
        <li class="active"><a href="about">About</a></li>
      </ul>
    </div>
  </div>
</nav>

<div class="container">
  <div class="starter-template">
    <h1>Spring Boot Web JSP Example</h1>
    <h2>Message: {msg}</h2>
  </div>
</div>
</body>
</html>

```

d- Résultat des modifications :

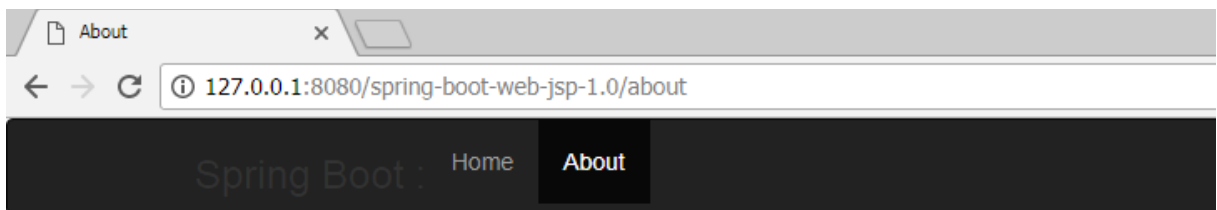
La page About s'ajoute au menu comme suit :



Spring Boot Web JSP Example

Message: Hello world

En appuyant sur le lien About on obtient la page suivante :



Spring Boot Web JSP Example

Message: This is about

3- Fichiers sources :

1- SpringBootApplication.java :

```
package com.mkyong;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.boot.builder.SpringApplicationBuilder;
import org.springframework.boot.web.support.SpringBootServletInitializer;

@SpringBootApplication
public class SpringBootApplication extends SpringBootServletInitializer {

    @Override
    protected SpringApplicationBuilder configure(SpringApplicationBuilder
application) {
        return application.sources(SpringBootApplication.class);
    }

    public static void main(String[] args) throws Exception {
        SpringApplication.run(SpringBootApplication.class, args);
    }
}
```

2- WelcomeController.java :

```
package com.mkyong;

import java.util.Map;

import org.springframework.beans.factory.annotation.Value;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;

@Controller
public class WelcomeController {

    // inject via application.properties
    @Value("${welcome.message:test}")
    private String message = "Hello World";

    @RequestMapping("/")
    public String welcome(Map<String, Object> model) {
        model.put("message", this.message);
        return "welcome";
    }
}
```

3- application.properties :

```
spring.mvc.view.prefix: /WEB-INF/jsp/  
spring.mvc.view.suffix: .jsp  
welcome.message: Hello world
```

4- welcome.jsp :

```
<!DOCTYPE html>
<%@ taglib prefix="spring" uri="http://www.springframework.org/tags"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<html lang="en">

<head>
<link rel="stylesheet" type="text/css"
      href="webjars/bootstrap/3.3.7/css/bootstrap.min.css" />

<c:url value="/css/main.css" var="jstlCss" />
<link href="{jstlCss}" rel="stylesheet" />

<script type="text/javascript"
        src="webjars/bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>

<body>

    <nav class="navbar navbar-inverse">
        <div class="container">
            <div class="navbar-header">
                <h3>Spring Boot :</h3>
            </div>
            <div id="navbar" class="collapse navbar-collapse">
                <ul class="nav navbar-nav">
                    <li class="active"><a href="#">Home</a></li>
                </ul>
            </div>
        </div>
    </nav>

    <div class="container">

        <div class="starter-template">
            <h1>Spring Boot Web JSP Example</h1>
            <h2>Message: {message}</h2>
        </div>

    </div>
    <!-- /.container -->

</body>

</html>
```

5- main.css :

```
h1{  
    color:#0000FF;  
}  
  
h2{  
    color:#FF0000;  
}
```

6- pom.xml :

```
<?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">
  <modelVersion>4.0.0</modelVersion>

  <artifactId>spring-boot-web-jsp</artifactId>
  <packaging>war</packaging>
  <name>Spring Boot Web JSP Example</name>
  <description>Spring Boot Web JSP Example</description>
  <url>https://www.mkyong.com</url>
  <version>1.0</version>

  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>1.4.2.RELEASE</version>
  </parent>

  <properties>
    <java.version>1.7</java.version>
  </properties>

  <dependencies>

    <!-- Web -->
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>

    <!-- Web with Tomcat + Embed -->
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-tomcat</artifactId>
      <scope>provided</scope>
    </dependency>

    <!-- JSTL -->
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>jstl</artifactId>
    </dependency>

    <!-- Need this to compile JSP -->
    <dependency>
      <groupId>org.apache.tomcat.embed</groupId>
      <artifactId>tomcat-embed-jasper</artifactId>
      <scope>provided</scope>
    </dependency>

    <!-- Need this to compile JSP -->
    <dependency>
      <groupId>org.eclipse.jdt.core.compiler</groupId>
      <artifactId>ecj</artifactId>
      <version>4.6.1</version>
  </dependencies>
</project>
```

```
        <scope>provided</scope>
    </dependency>

    <!-- Optional, for bootstrap -->
    <dependency>
        <groupId>org.webjars</groupId>
        <artifactId>bootstrap</artifactId>
        <version>3.3.7</version>
    </dependency>

</dependencies>
<build>
    <plugins>
        <!-- Package as an executable jar/war -->
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
    </plugins>
</build>
</project>
```