1，首先你需要在你的pom文件中加入jsp依赖支持。

<parent>

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

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

<version>1.5.3.RELEASE</version>

</parent>

<dependencies>

<dependency>

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

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

</dependency>

<!-- spring boot 内置tomcat jsp支持 -->

<dependency>

<groupId>org.apache.tomcat.embed</groupId>

<artifactId>tomcat-embed-jasper</artifactId>

</dependency>

<!-- servlet api -->

<!--<dependency>-->

<!--<groupId>javax.servlet</groupId>-->

<!--<artifactId>javax.servlet-api</artifactId>-->

<!--</dependency>-->

<!-- jstl -->

<dependency>

<groupId>jstl</groupId>

<artifactId>jstl</artifactId>

<version>${jstl.version}</version>

</dependency>

2，然后在配置文件中指定你的视图位置

yml写法

spring:

mvc:

view:

prefix: /

suffix: .jsp

proferties写法

spring.mvc.view.prefix=/

spring.mvc.view.suffix=.jsp

注意prefix这里写的就是你自己的jsp文件存放的位置，例如你在WEB-INF/jsp下面，路径就是

spring.mvc.view.prefix=/WEB-INF/jsp

3,在完成上面的步骤之后，运行项目就可以访问jsp界面，但是别高兴的太早，当你打成一个jar包的时候，你会发现你不能访问jsp了，原因是你需要

在你的pom文件把webapp的文件打包进去。

在pom文件中<build>里面加入以下代码

<build>

<plugins>

<!-- spring dev -->

<plugin>

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

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

<version>1.4.2.RELEASE</version>

<configuration>

<mainClass>test.Application</mainClass>

</configuration>

<executions>

<execution>

<goals>

<goal>repackage</goal>

</goals>

</execution>

</executions>

<dependencies>

<!-- spring热部署 -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>springloaded</artifactId>

<version>1.2.6.RELEASE</version>

</dependency>

</dependencies>

</plugin>

<!-- 忽略无web.xml警告 -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-war-plugin</artifactId>

<configuration>

<failOnMissingWebXml>false</failOnMissingWebXml>

</configuration>

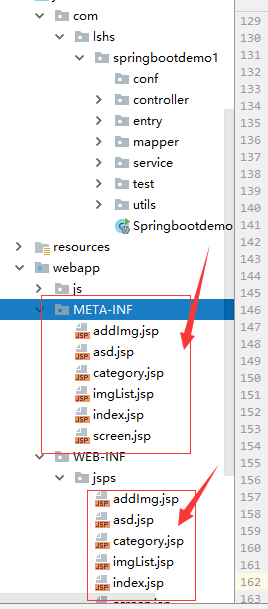
</plugin>

</plugins>

<resources>

<!-- 打包时将jsp文件拷贝到META-INF目录下 -->

没有的话就新建一个文件夹



<resource>

<!-- 指定resources插件处理哪个目录下的资源文件 -->

<directory>src/main/webapp</directory>

<!--注意此次必须要放在此目录下才能被访问到 -->

<targetPath>META-INF/resources</targetPath>

<includes>

<include>\*\*/\*\*</include>

</includes>

</resource>

<resource>

<directory>src/main/resources</directory>

<includes>

<include>\*\*/\*\*</include>

</includes>

<filtering>false</filtering>

</resource>

</resources>

</build>

在这里如果你的pom文件继承了springboot的父pom是1.5.3版本的，这里很坑一点就是

spring-boot-maven-plugin 1.5.3版本的打包控件打出来的jar包是无法访问jsp的，直接404，即使你的jsp文件打包进去了。

spring-boot-maven-plugin 1.4.2版本就是可以的，我不知道这两个版本打包控件有什么区别，有知道的大神可以讲讲。

下面的这段代码

<**resource**>

*<!-- 指定resources插件处理哪个目录下的资源文件 -->*

<**directory**>src/main/webapp</**directory**>

*<!--注意此次必须要放在此目录下才能被访问到 -->*

<**targetPath**>META-INF/resources</**targetPath**>

<**includes**>

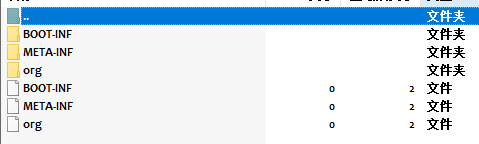
<**include**>\*\*/\*\*</**include**>

</**includes**>

</**resource**>

就是把jsp文件打进jar包的代码。打完包的同学，可以把jar包解压看看里面有没有这个文件。

用解压文件打开jar



META-INF

