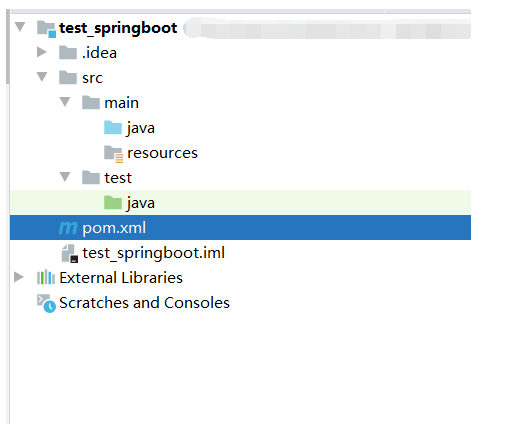
官网链接: Developing Your First Spring Boot Application

 使用Spring Boot可以极大方便开发。下面以一个网页的请求和响应为例。

一、创建一个Maven项目

 创建一个名为test\_springboot的Maven项目。



二、导入坐标

 在pom.xml中导入坐标

<parent>

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

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

<version>2.6.4</version>

</parent>

<dependencies>

<dependency>

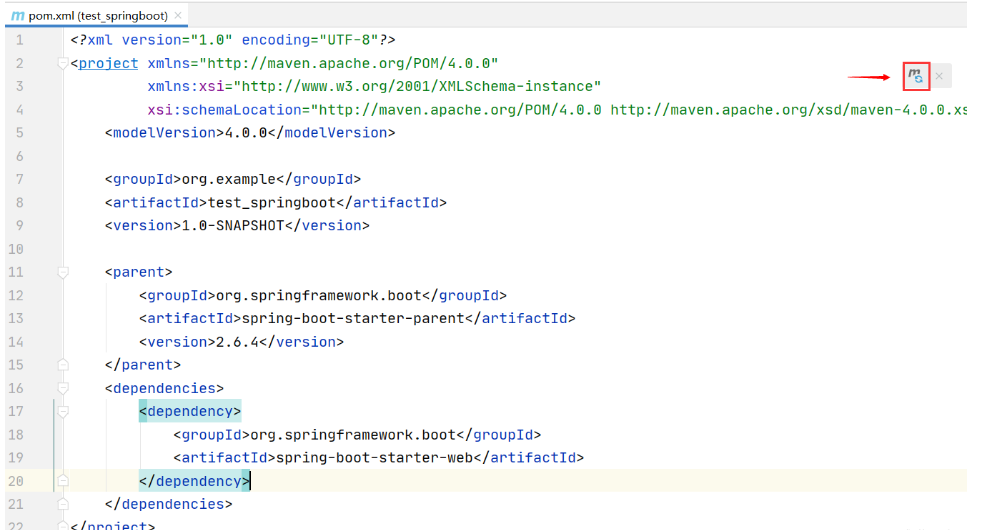
<groupId>org.springframework.boot</groupId>

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

</dependency>

</dependencies>

并且点击右上角的刷新。等待加载完jar包



三、创建启动类

 在com.my包下创建Application类，加入@RestController注解。main函数加入一行SpringApplication.run( Application.class,args )项目就是运行这个main函数启动。

package com.my;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class Application {

public static void main(String[] args) {

SpringApplication.run( Application.class,args );

}

}

四、创建请求响应类（后端）

 在com.my.bean创建一个Person类，用来请求响应交互数据。

package com.my.bean;

public class Student {

private String id;

private String name;

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

在com.my.controller包中创建HelloController类。将该类加上@RestController注解。

创建一个接受请求的方法，在该方法上加上@RequestMapping("/Hello")。收到请求的name和id就会返回该对象。

package com.my.controller;

import com.my.bean.Student;

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

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

import java.util.HashMap;

import java.util.Map;

@RestController

public class HelloController {

@RequestMapping("/Hello") //获取 http://localhost:8080/Hello 的请求信息

public Map<String ,Object> Hello(String id , String name){ //获取 /Hello?id=value1&name=value2的值

Map<String ,Object> map = new HashMap<String,Object>();

Student s = new Student();

s.setId(id);

s.setName(name);

map.put("code",1);

map.put("obj",s);

return map;

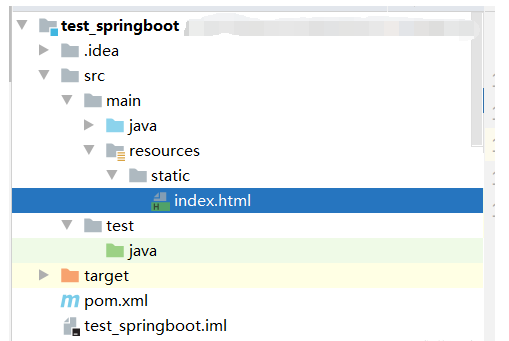
}

}

五、加入HTML（前端）

 在src下的main下的resources下创建static目录，在static目录下创建index.html。

 在index.html加入如下代码



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Title</title>

</head>

<body>

<form action="/Hello">

id:<input type="text" name="id" > <br>

name:<input type="text" name="name"> <br>

<input type="submit" >

</form>

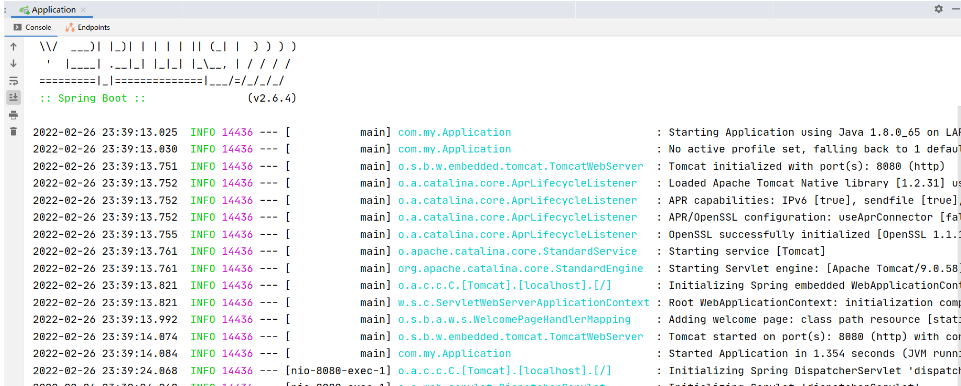
</body>

</html>

六、测试

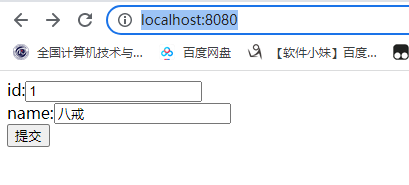
 运行第三步的Application类。

 控制台效果:

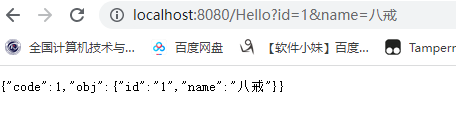


 在浏览器输入http://localhost:8080/index.html 。网页出现如下效果:

 任意输入数据。点击提交：



 点击提交后的页面效果为:



Localhost换成本地Ip同理

七、项目上线

 导出jar包。在pom.xml加入如下代码:

<build>

<plugins>

<plugin>

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

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

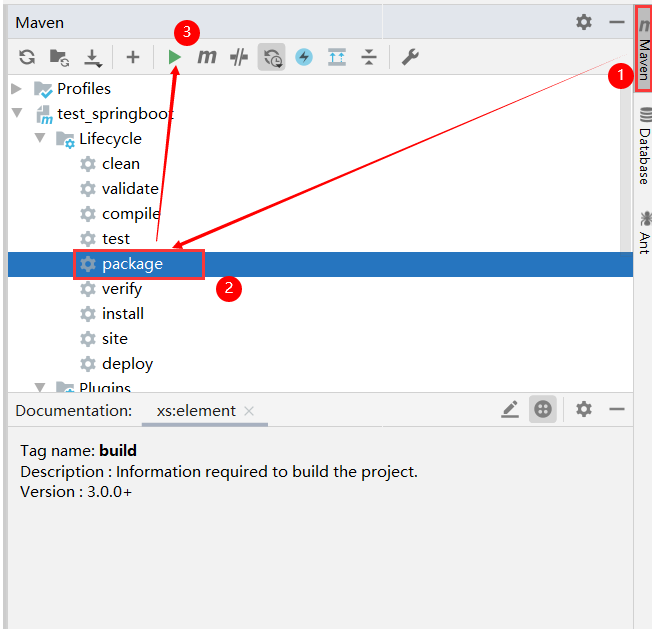
</plugin>

</plugins>

</build>



 运行package。



 运行结果:

