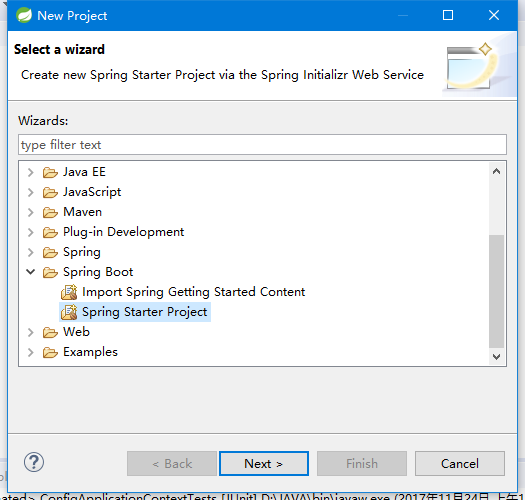
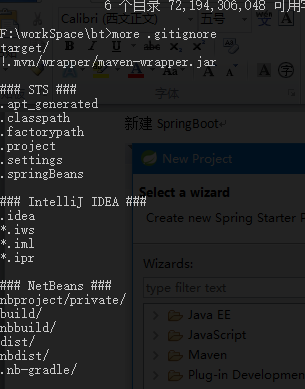
新建SpringBoot



此文件不会把target、sts、intellij文件放到git上



在工作空间的项目下



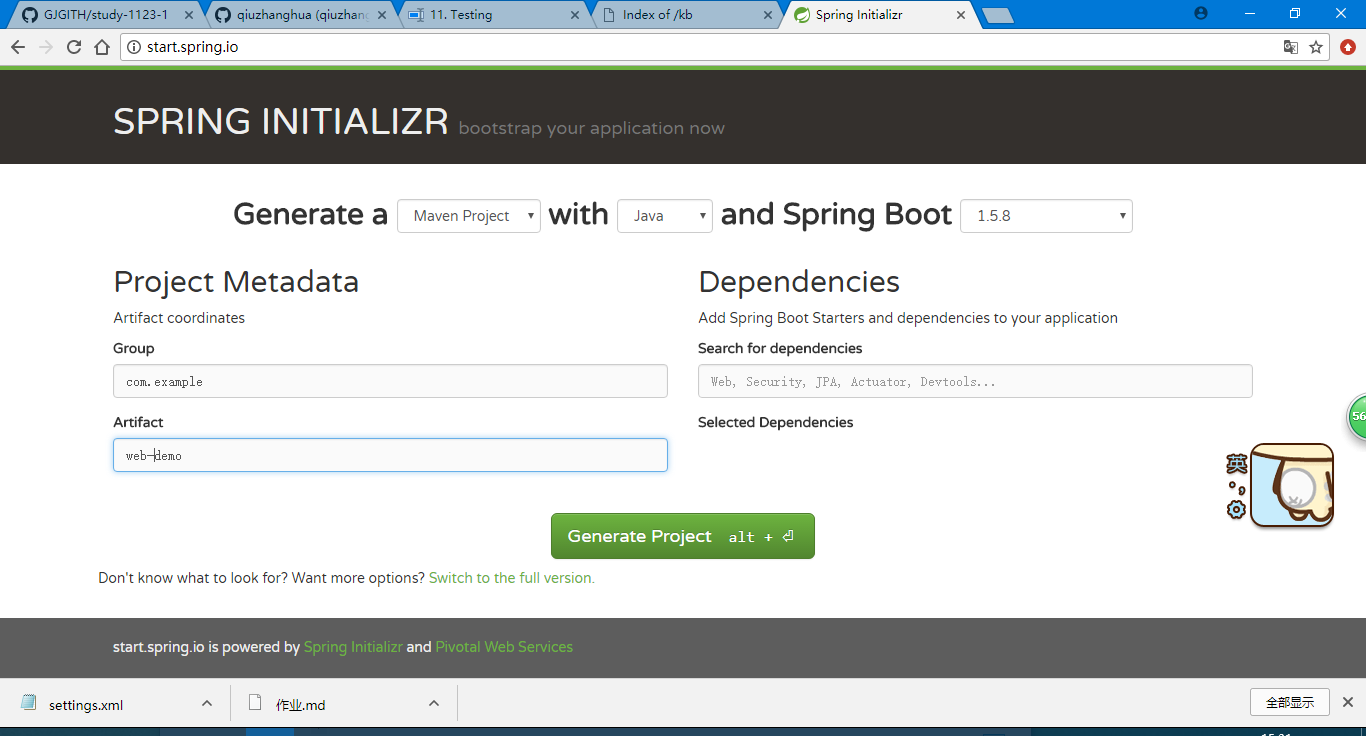
jar文件打包 跳过测试文件

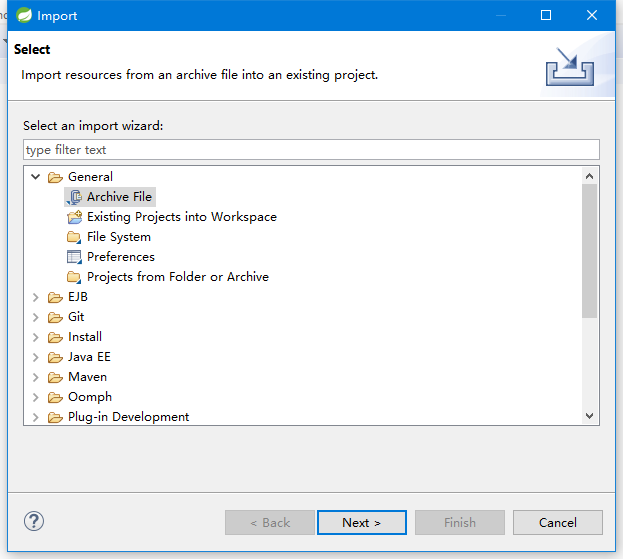


运行springboot的程序

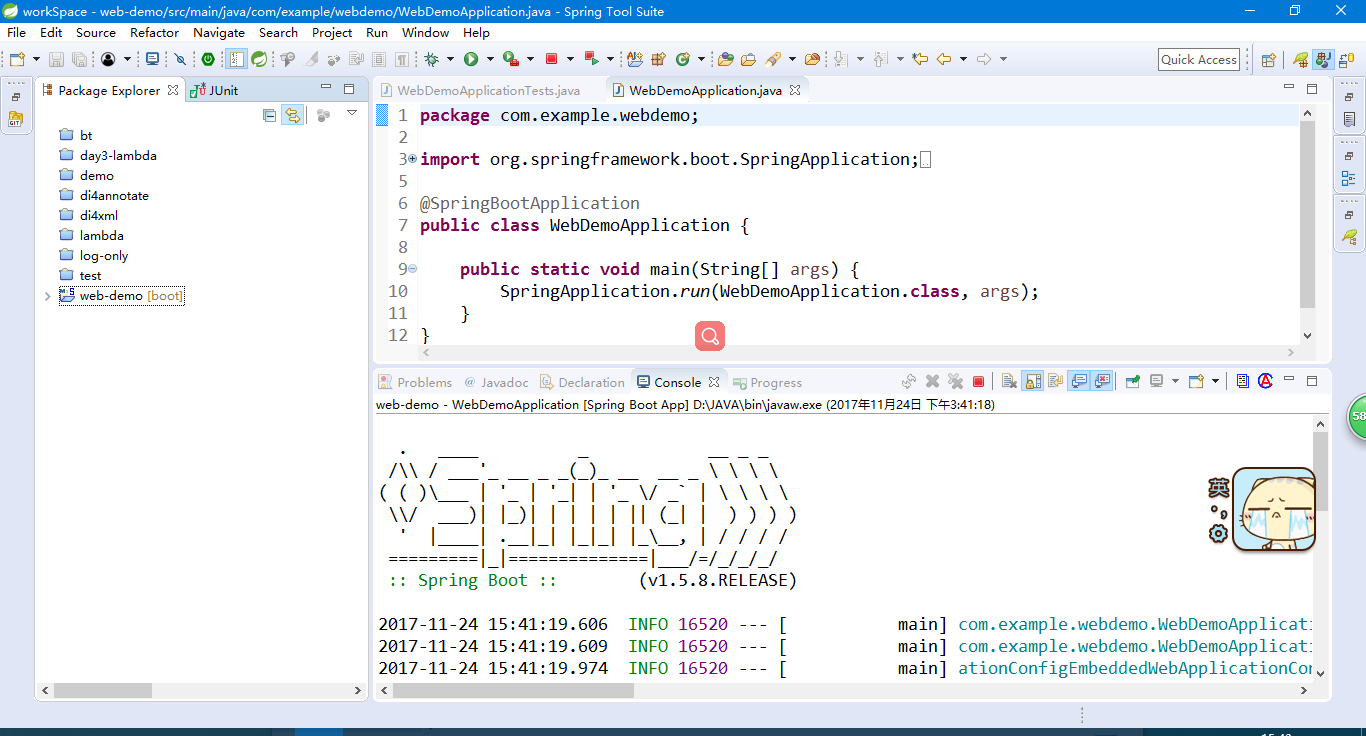


新建springboot

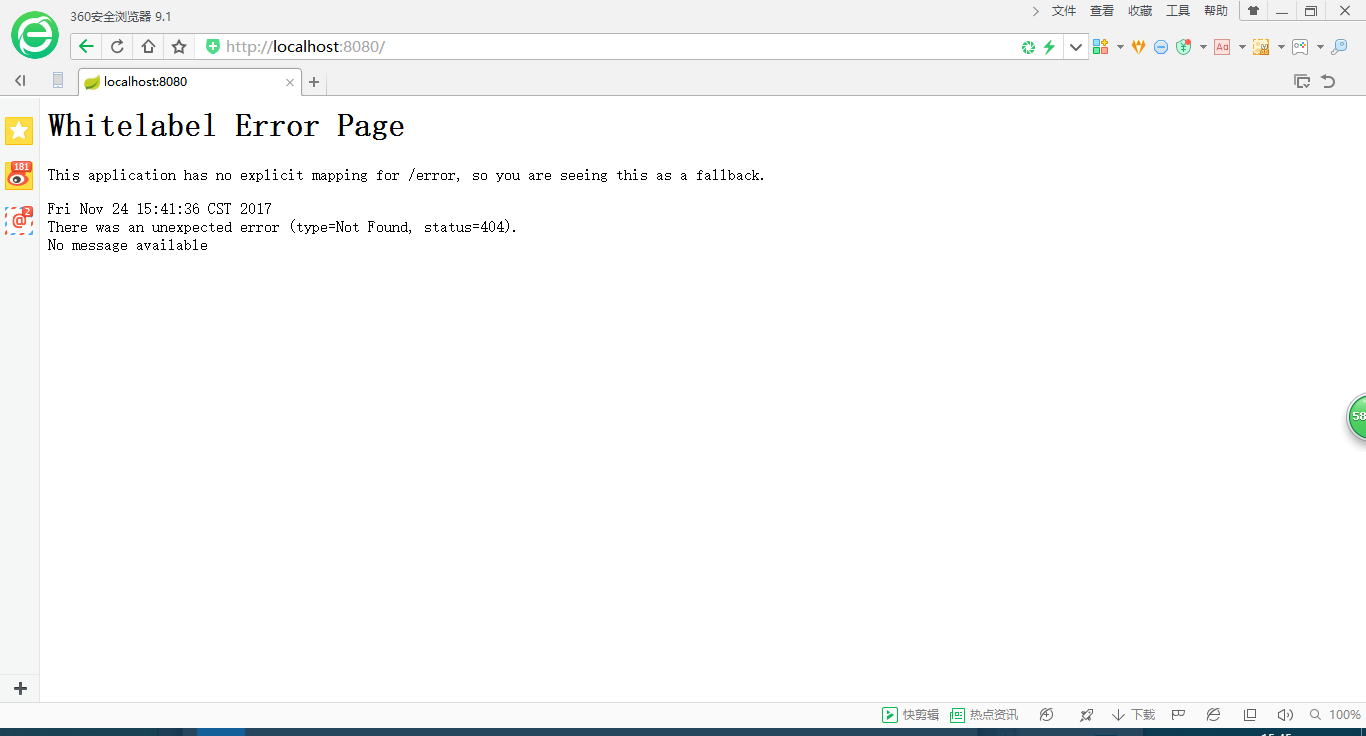




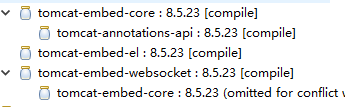
运行main文件



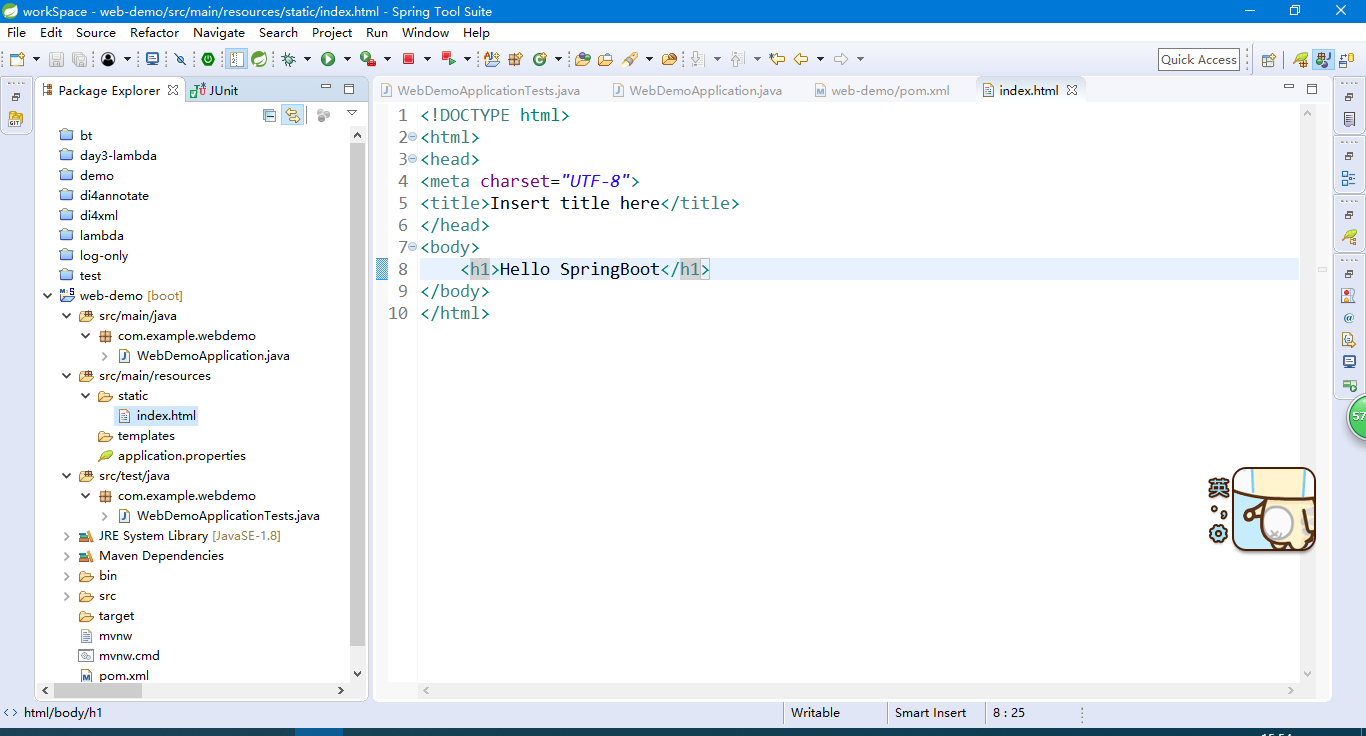
地址栏输入localhost:8080



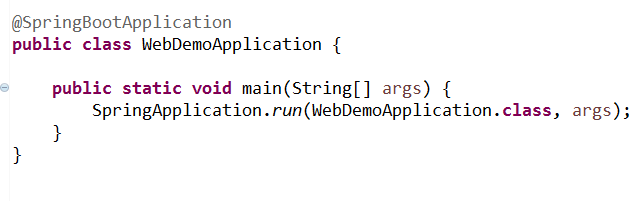
内嵌tomcat



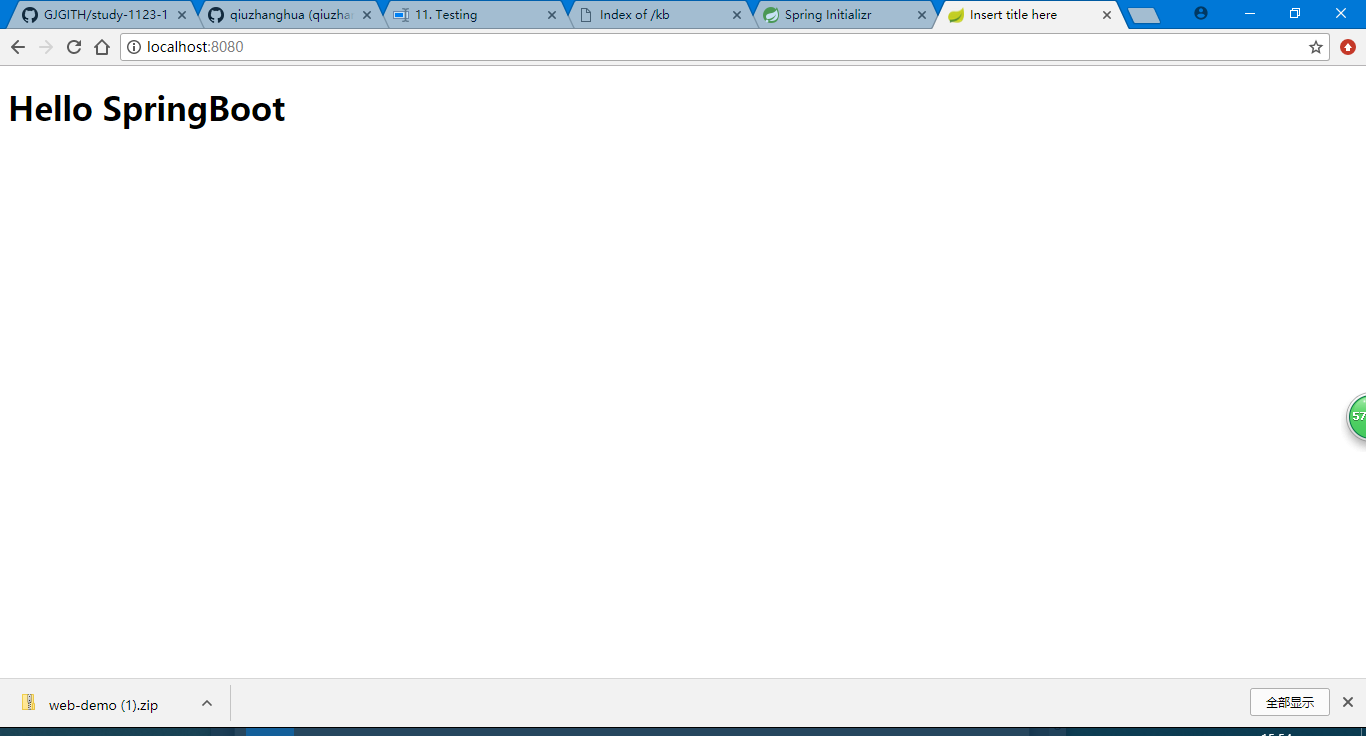
新建Html



运行main文件



地址栏输入localhost:8080



## 常用注解：

* @RestController

@ResponseBody和@Controller的合集

继承自@Controller（在springMVC使用@Controller标识当前类是控制器servlet）

@Controller用于标注控制层组件(如struts中的action)

在SpringBoot中替换了@Controller@RequestMapping

注解提供路由信息。它告诉Spring任何来自"/"路径的HTTP请求都应该被映射到 home 方法。

* @Configuration

类级别的注解，

一般这个注解，用来标识main方法所在的类

相当于将类变成xml文件，用于里边的bean被调取

@ComponentScan

 表示将该类自动发现（扫描）并注册为Bean，可以自动收集所有的Spring组件，包括@Configuration类。

经常使用@ComponentScan注解搜索beans，并结合@Autowired注解导入

如果没有配置的话，Spring Boot会扫描启动类所在包下以及子包下的使用了@Service,@Repository等注解的类。

@Component

把普通pojo实例化到spring容器中，相当于配置文件中的<bean id="" class=""/>

* @ImportResource

用于加载XML配置文件

* @AutoWired
  1. 用于把配置好的Bean拿来用，完成属性、方法的组装
  2. 它可以对类成员变量、方法及构造函数进行标注，完成自动装配的工作。
  3. 当加上（required=false）时，就算找不到bean也不报错。
  4. 一般结合@ComponentScan注解，来自动注入一个Service或Dao级别的Bean
* @Import
  1. 用来导入其他的配置类
* @Repository
  1. 使用@Repository注解可以确保DAO或者repositories提供异常转译
  2. Repository修饰的DAO或者repositories类会被ComponetScan发现并配置，同时也不需要为它们提供XML配置项。
* @Bean

用@Bean标注方法等价于XML中配置的bean。

* @Inject

等价于默认的@Autowired，只是没有required属性；

周末作业

1. 写Spring Framework Reference 和Spring Boot Reference的读书笔记，以md文件格式存放到github上

2. 新建Spring Boot项目(模拟一个上课的场景，写服务(签到、查询，统计等)，不用存库，不用web界面，使用命令行界面就可以，要有单元测试，要github提交)。

3. Java 8的作业要完成。

4. 完成一个上台阶的问题，一个人每次可以上1、2或者3级台阶，问10级台阶有多少种上法。在stream-java8的基础上修改和提交

。fn=fn-1 +fn-2 +fn-3

f1=1

f2=2

f3=4

f4=7