请帮忙开发一个活动管理系统，要求如下：

1. 实现一个简易的类springMVC框架，必需具备以下功能：
   1. 实现DispatcherServlet模块；
   2. 实现HandleMapping模块；
   3. 实现基于注解的Handler（Controller）模块；
   4. 基于已实现的MVC框架实现用户登录/登出功能；
   5. spring的AOP机制，实现所有请求的log记录功能。

答：

代码位于 SpringMVC\_HW文件中，实现了DispatcherServlet、HandleMapping、自定义注解和简易AOP功能。流程如下

1. 启动时加载web.xml，取得properties字段，从而拿到了要扫描的包名，即com.tplink.hujie.myspringmvc
2. 扫描com.tplink.hujie.myspringmvc包下面所有的类，类名输出到classNames[]；
3. 对classNames[]中的每一个类名，检测该类有没有加自定义的@MyController,

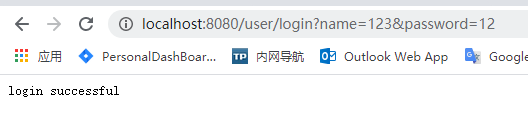
@MyService注解，如果有，通过反射机制,实例化,并且放到IOC容器中；

1. 扫描注解,将切面里的切点扫描出来,添加需要代理的类到proxy这个HashMap里
2. 对ioc容器中的每一个bean，通过反射获取其属性（包括私有属性），如果有@MyAutowired注解，注入其所需依赖（如果类在proxy中，则注入代理的类）；
3. 初始化HandlerMapping(将url和method对应上)，获取controller的所有方法，如果有@MyRequestMapping注解，则把注解中的url和该方法对应起来，通过invoke调用该方法。

环境配置：配置本地tomcat。

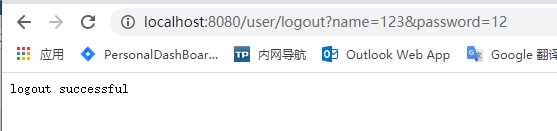
运行结果：

测试登录





测试退出：



1. 思考题：Spring MVC提供拦截器机制和servlet的Filter机制有什么不同，各自的优缺点和使用场景是什么？

相同点：

二者都是AOP程序思想的体现，都能实现权限检查、日志记录。

不同点：

1. 拦截器是基于Java的反射机制的，而过滤器Filter是基于函数回调
2. 拦截器不依赖与servlet容器，过滤器Filter依赖与servlet容器
3. 拦截器只能对部分请求起作用，而过滤器Filter则可以对几乎所有的请求起作用。
4. 拦截器可以访问spring容器上下文、值栈里的对象，而过滤器Filter不能访问
5. 在controller的生命周期中，拦截器可以多次被调用，而过滤器Filter只能在容器初始化时被调用一次
6. 拦截器能够深入到方法前后、异常抛出前后等，既可以用于Web程序，也可以用于Application、Swing程序中
7. Filter主要是针对URL地址做一个编码的事情、过滤掉没用的参数、安全校验