完成功能：用户提交一个请求，服务端处理器在接收到这个请求后，给出一条欢迎信息，在响应页面中显示该信息。

1. **导入Jar包**
2. **注册中央调度器**
3. 全限定性类名

该中央调度器为一个Servlet，名称为DispatcherServlet。中央调度器的全限定类名在导入的Jar文件spring-webmvc-4.2.1.RELEASE.jar的第一个包org.springframework.web.servlet下可找到。

1. <load-on-startup/>

在<servlet/>中添加<load-on-startup/>的作用是，标记是否在Web服务器（这里是Tomcat）启动时会创建这个Servlet实例，即是否在Web服务器启动时调用执行该Servlet的init()方法，而不是在真正访问时才创建。

它的值必须是一个整数。

* 当值大于等于0时，表示容器在启动时就加载并初始化这个servlet，数值越小，该Servlet的优先级就越高，其被创建的也就越早；
* 当值小于0或者没有指定时，则表示该Servlet在真正被使用时才会去创建。
* 当值相同时，容器会自己选择创建顺序。

1. <url-pattern>

对于<url-pattern/>，不能写为/\*，最好也不要写为/，建议写为\*.do的形式。

1. 配置文件位置与名称

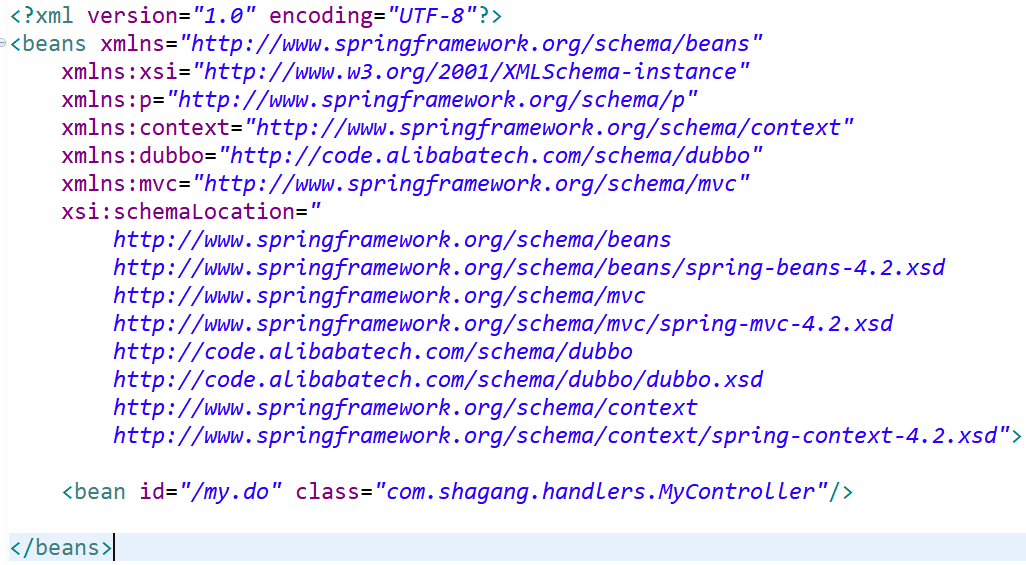
注册完毕后，可直接在服务器上发布运行。此时，默认浏览器页面，及MyEclipse控制台均会抛出FileNotFoundException异常。即默认要从项目根下的WEB-INF目录下找名称为Servlet名称-servlet.xml的配置文件。这里的“Servlet名称”指的是注册中央调度器时<servlet-name/>标签中指定的Servlet的name值。本例配置文件名为springmvc-servlet.xml。

而一般情况下，该配置文件是放在类路径下，即src目录下。所以，在注册中央调度器时，还需要为中央调度器设置查找SpringMVC配置文件路径，及文件名。



1. **创建SpringMVC配置文件**

在工程的类路径即src目录下创建SpringMVC的配置文件springmvc.xml。该文件名可以任意命名。而该xml文件的约束，则使用Spring配置文件最全约束。



1. **定义处理器**

该处理器需要实现Controller接口。



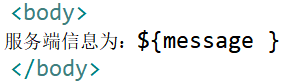
ModelAndView类中的addObject()方法用于向其Model中添加数据。Model的底层为一个Map。另外需要注意的是，addObject()方法的返回值为ModelAndView，则说明可向该Model中连续添加多个数据。

1. **注册处理器**

在springmvc.xml中注册处理器。不过，需要注意处理器的id属性值为一个请求URI。表示当客户端提交该请求时，会访问class指定的这个处理器。

1. **定义目标页面**

在WEB-INF目录下新建一个子目录jsp，其中新建一个JSP页面show.jsp。



1. **修改视图解析器的注册**



SpringMVC框架为了避免对于请求资源路径与扩展名上的冗余，在视图解析器InternalResouceViewResolver中引入了请求的前辍与后辍。而ModelAndView中只需给出要跳转页面的文件名即可，对于具体的文件路径与文件扩展名，视图解析器会自动完成拼接。

1. **修改处理器**

