1. **配置详解**
2. 建议写为\*.do形式
3. 不能写/\*（会拦截所有动态请求）

这里的url-pattern不能写为/\*，因为DispatcherServlet会将向动态页面的跳转请求，即向JSP页面的跳转请求也当作是一个普通的Controller请求。中央调度器会调用处理器映射器为其查找相应的处理器。当然是找不到的，所以在这种情况下，所有的JSP页面跳转均会报404错误。

1. 最好也不要写为/（会拦截静态资源）

最好也不要写为/，因为DispatcherServlet会将向静态资源的获取请求，例如.css、.js、.jpg、.png等资源的获取请求，当作是一个普通的Controller请求。中央调度器会调用处理器映射器为其查找相应的处理器。当然也是找不到的，所以在这种情况下，所有的静态资源获取请求也均会报404错误。

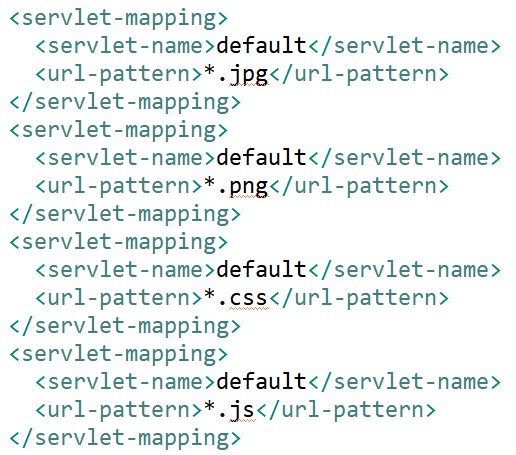
1. **静态资源访问（解决静态资源无法访问的方法）**

<url-pattern/>的值并不是说写为/后，静态资源就无法访问了。经过一些配置后，该问题也是可以解决的。

1. 使用Tomcat中名为default的Servlet

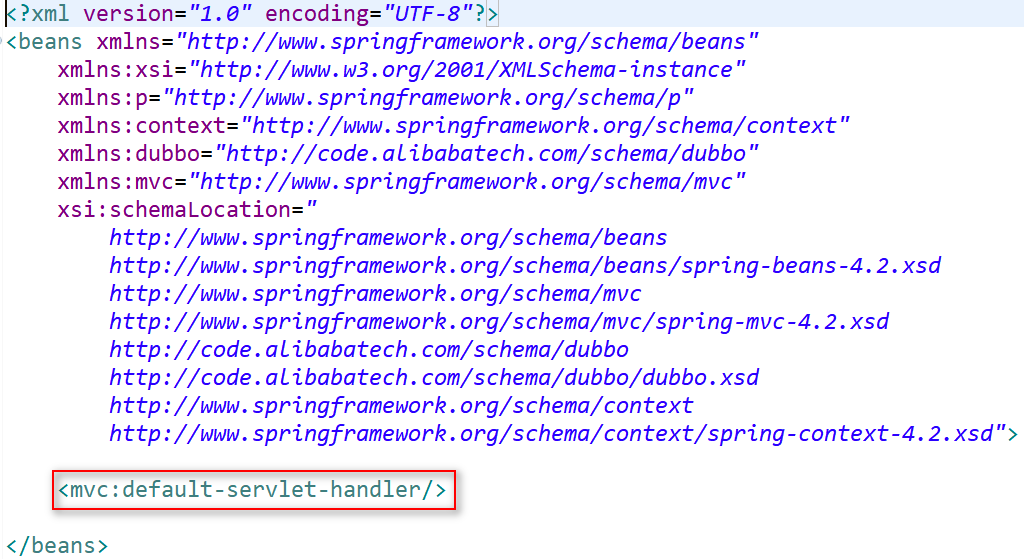
在Tomcat中，有一个专门用于处理静态资源访问的Servlet–DefaultServlet。其<servlet-name/>为default。可以处理各种静态资源访问请求。该Servlet注册在Tomcat服务器的web.xml中。在Tomcat安装目录/conf/web.xml。

我们需要做的就是直接使用它即可，即直接在web.xml中注册<servlet-mapping/>即可。



1. 使用<mvc:default-servlet-handler/>

只需要在 springmvc.xml中添加<mvc:default-servlet-handler/>标签即可。



<mvc:default-servlet-handler/>会将对静态资源的访问请求添加到SimpleUrlHandlerMapping的urlMap中，key就是请求的URI，而value则为默认Servlet请求处理器DefaultServletHttpRequestHandler对象。而该处理器调用了Tomcat的DefaultServlet来处理静态资源的访问请求。

1. 使用<mvc:resources/>

在Spring3.0.4版本后，Spring中定义了专门用于处理静态资源访问请求的处理器ResourceHttpRequestHandler。并且添加了<mvc:resources/>标签，专门用于解决静态资源无法访问问题。需要在springmvc.xml中添加如下形式的配置：



***location表示静态资源所在目录***。当然，这里的目录包含/WEB-INF/目录及其子目录。

***mapping表示对该资源的请求。注意，后面是两个星号\*\*。***

该配置会把对该静态资源的访问请求添加到SimpleUrlHandlerMapping的urlMap中，key就是真正与mapping的URI匹配的URI，而value则为静态资源处理器对象ResourceHttpRequestHandler。