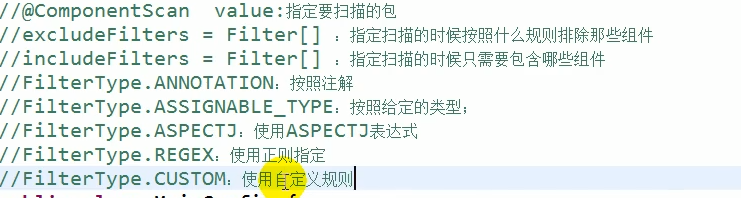
### Spring注解驱动开发

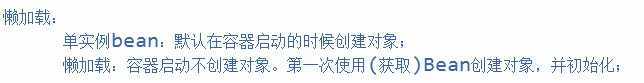
1. 扫描规则



1. 创建单实例多实例



1. 懒加载

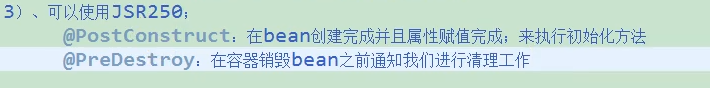


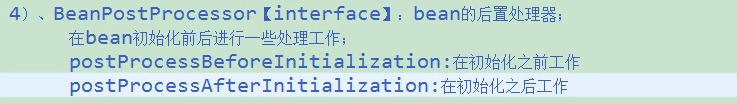
1. Bean的生命周期

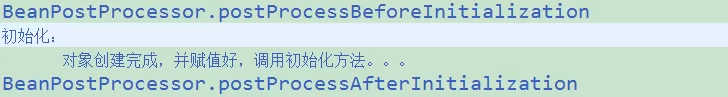






  
后置处理器





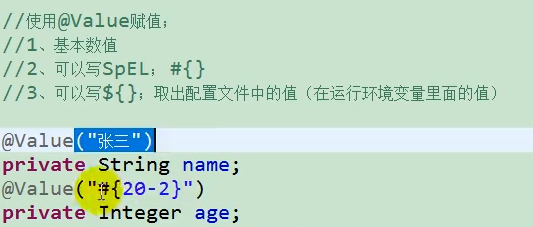
BeanPostProcessor原理



Spring底层对后置处理器的处理

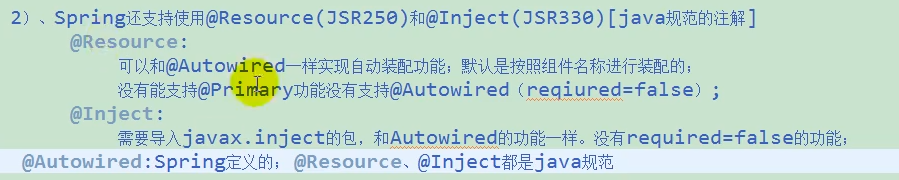


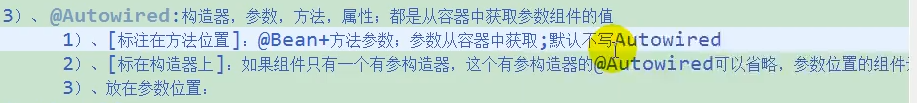
1. 属性赋值：注解配置的值会覆盖，配置类配置的@Bean注解时赋的值



1. 自动装配

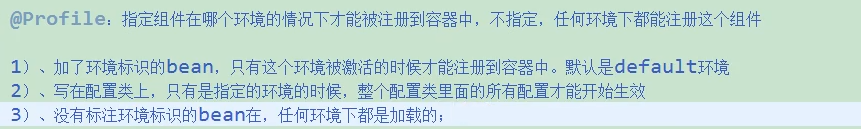






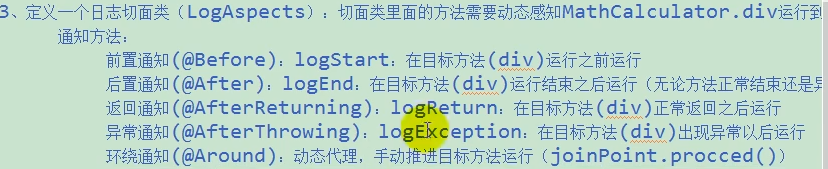
1. Profile

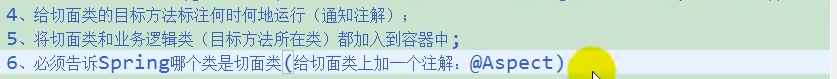




1. AOP面向切面编程

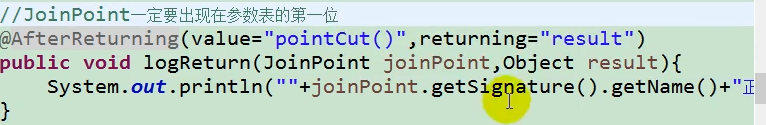












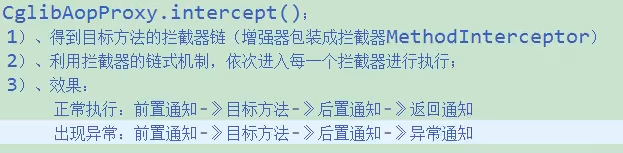
1. AOP原理







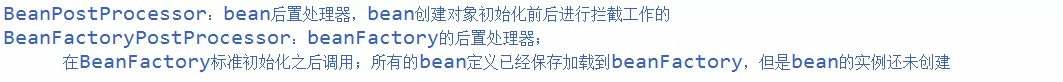
----------

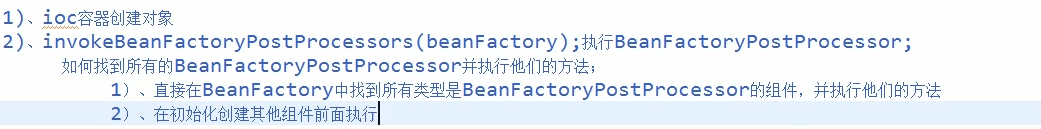


1. 事务配置



1. 扩展原理







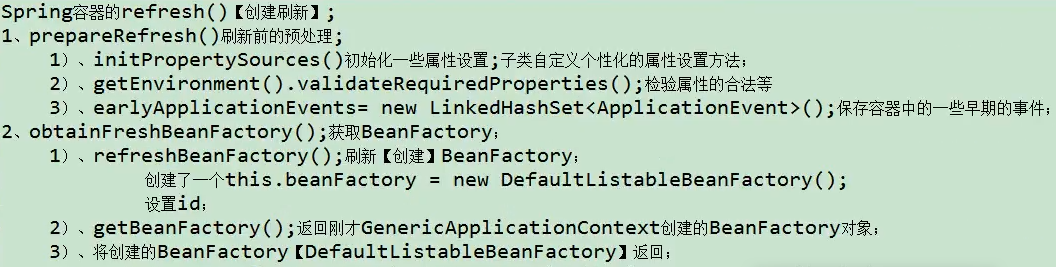


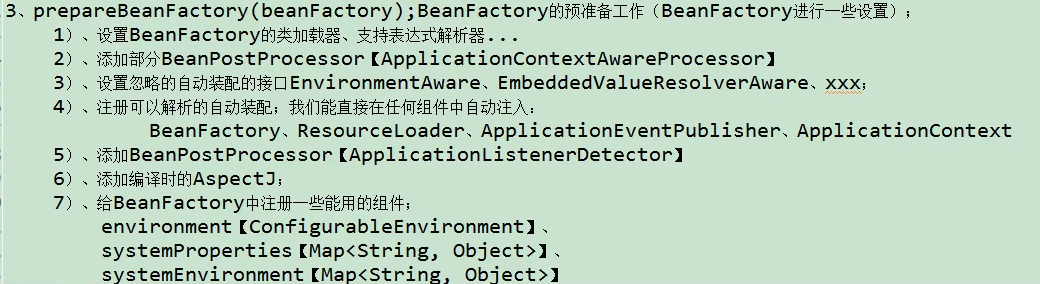


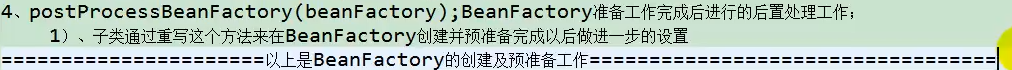


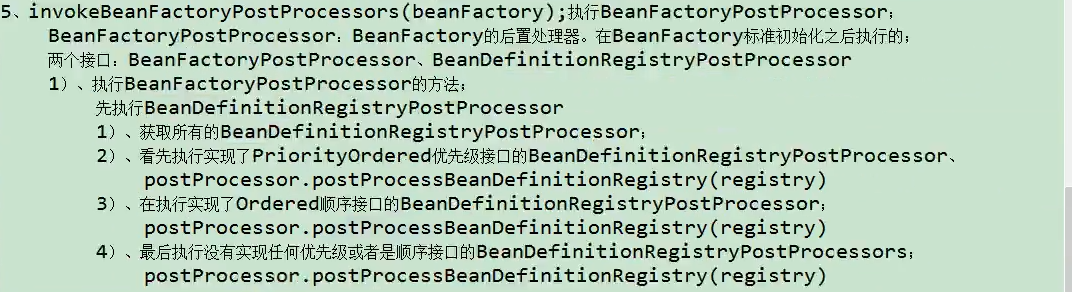


Spring容器创建过程

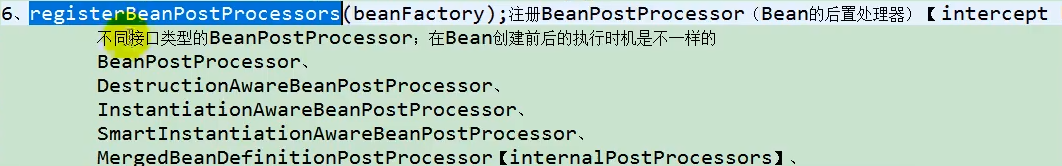


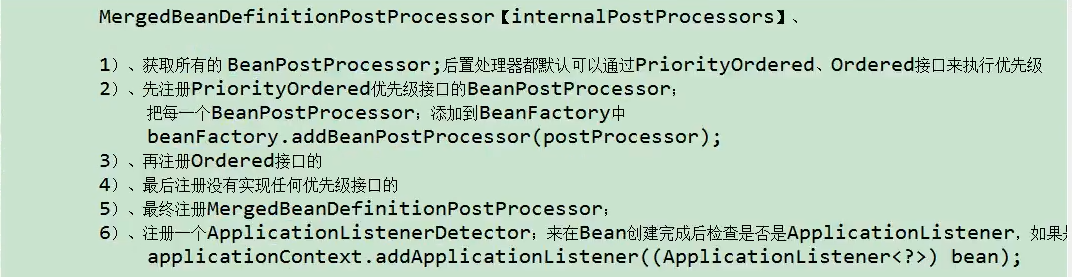


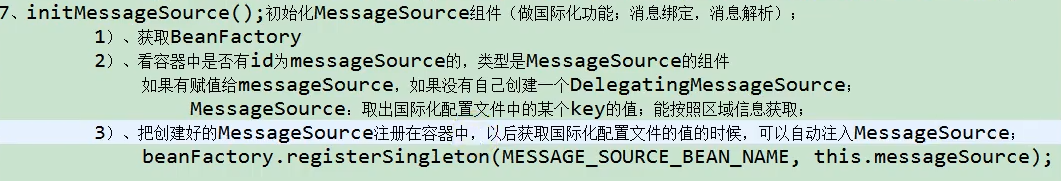




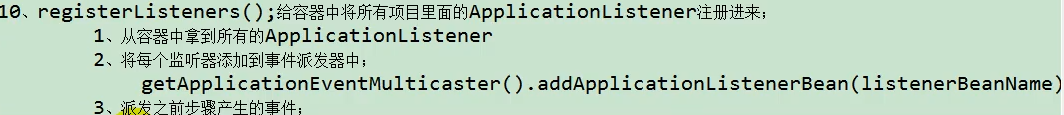


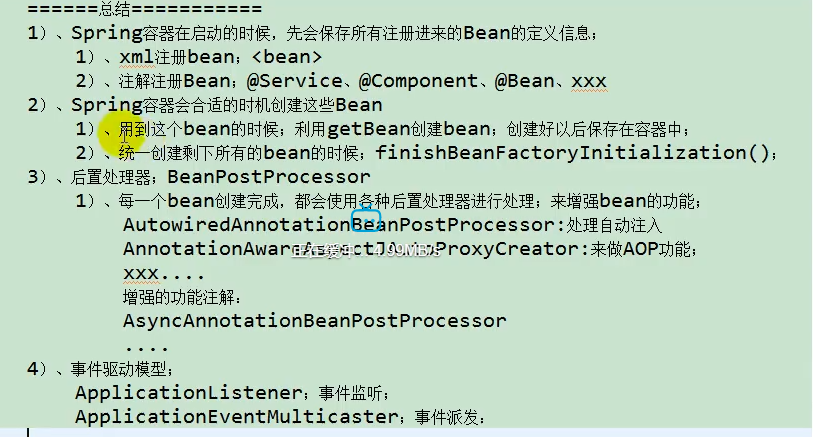








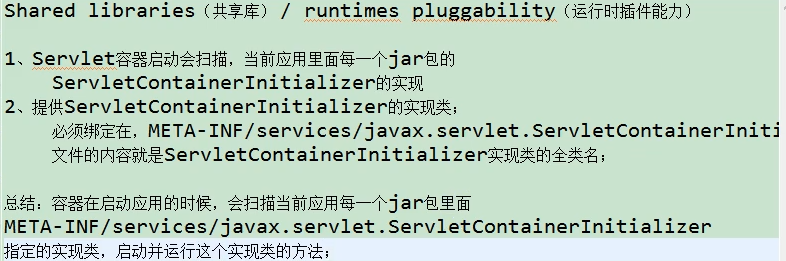




1. 在web.xml中配置监听器。注意：**监听器>过滤器>serlvet，配置的时候要注意先后顺序**

|  |
| --- |
| **分类：**  **按监听的对象划分，可以分为**   * **ServletContext对象监听器** * **HttpSession对象监听器** * **ServletRequest对象监听器**     **按监听的事件划分**   * **对象自身的创建和销毁的监听器** * **对象中属性的创建和消除的监听器** * **session中的某个对象的状态变化的监听器** |

1. Servlet3：META-INF文件需要放在src目录下

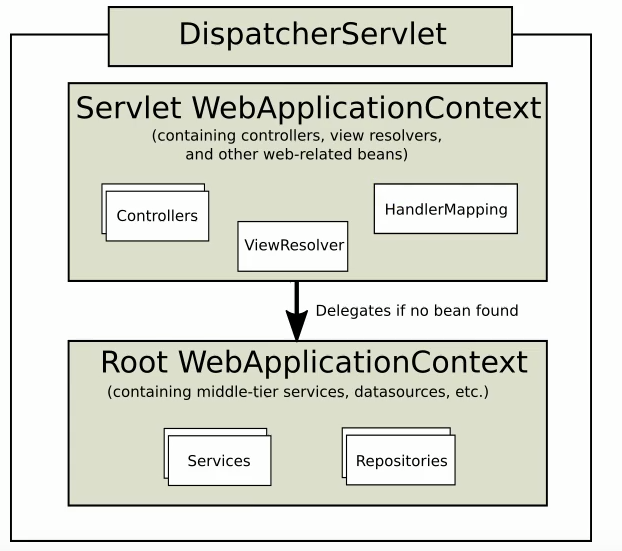




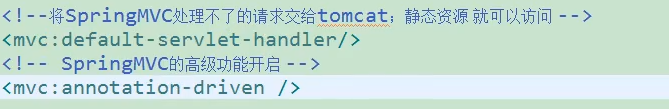
1. SpringMVC整合分析



1. 两个IOC容器管理的组件分类：
   1. SpringMVC - IOC子容器：Servlet WebApplication
   2. Spring - IOC父容器：Root WebApplicationContext



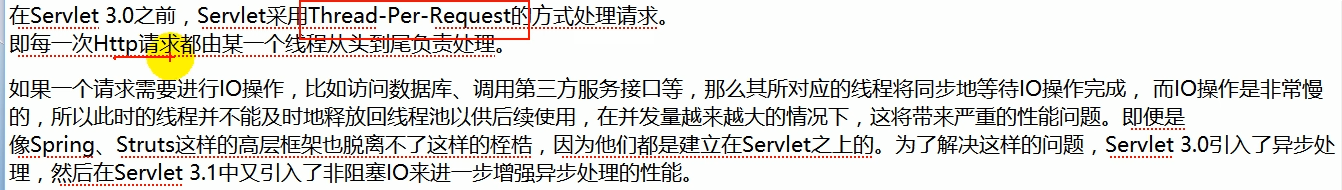
1. SpringMVC：default-servlet-handler

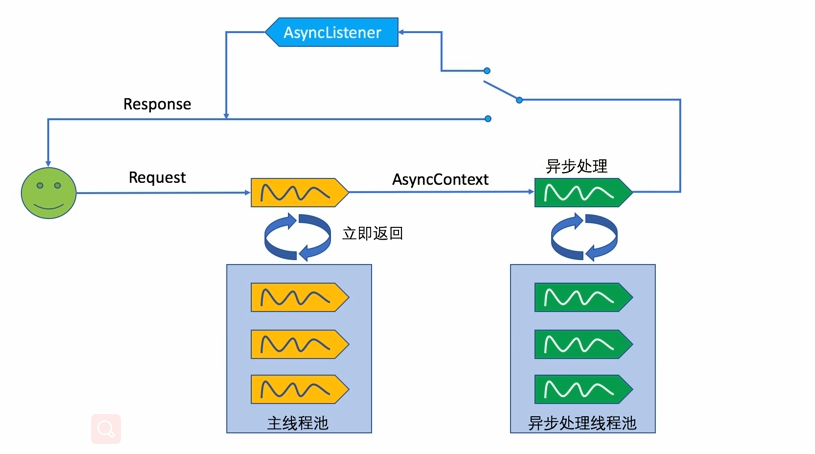


1. 坑总结，bug
   1. 创建Maven的web项目时，需要需要自己创建文件目录
   2. 使用tomcat部署时要是用：war结尾的
   3. 注解要检查好
   4. 静态资源不要放在WEB-INF目录下
2. SpringMVC定制配置



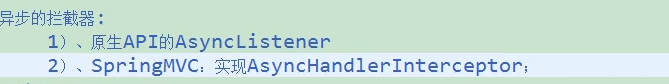
1. Servlet异步请求





SpringMVC异步处理：之Callable



  
之DefrredResult

