Spring Aop、拦截器、过滤器的区别

Filter过滤器：**拦截web访问url地址。**

Interceptor拦截器：**拦截以 .action结尾的url，拦截Action的访问。**

Spring AOP拦截器：**只能拦截Spring管理Bean的访问（业务层Service）**

----------------------------------------------------------------------------

Spring AOP

Spring AOP，是AOP的一种实现，**使用的是代理模式。**

Filter

Filter(过滤器)是J2EE的规范，Servlet2.3开始引入/实现的是**职责链模式**。Filter可以用来设置字符集、控制权限、控制转向等等。**Filter也是AOP的一种实现。**

Interceptor

Interceptor (拦截器)，**是Struct2中的概念**。**同样是AOP的一种实现。**

拦截器和过滤器的区别：

　　1、**拦截器是基于Java的反射机制，过滤器是基于java的函数回调**

　　2、**拦截器不依赖于servlet容器，而过滤器依赖于servlet容器**

　　3、**拦截器只能对action请求起作用，过滤器几乎对所有的请求起作用**

　　4、**拦截器可以访问action上下文，值栈里的对象，而过滤器不能访问**

　　5、**在action生命周期中，拦截器可以被多次调用，过滤器只能在servlet溶初始化是调用一次**

　　6、**拦截器可以获取IOC容器中的各个bean，过滤器不行，在拦截器中注入一个service可以调用逻辑业务**

拦截器：

Servlet中的过滤器Filter是实现了统一设置编码，简化操作；同时还可进行逻辑判断，如用户是否已经登陆、有没有权限访问该页面等等工作。它是随你的web应用启动而启动的，只初始化一次，以后就可以拦截相关请求，只有当你的web应用停止或重新部署的时候才销毁

过滤器：

拦截器是在面向切面编程中应用的，就是在你的service或者一个方法前调用一个方法，或者在方法后调用一个方法。是基于JAVA的反射机制。拦截器不是在web.xml，比如struts在struts.xml中配置，

总结：

1.过滤器：所谓过滤器顾名思义是用来过滤的，在Java web中，你传入的request,response提前过滤掉一些信息，或者提前设置一些参数，然后再传入servlet或者struts的action进行业务逻辑，比如过滤掉非法url（不是login.do的地址请求，如果用户没有登陆都过滤掉）,或者在传入servlet或者struts的action前统一设置字符集，或者去除掉一些非法字符（聊天室经常用到的，一些骂人的话）。filter 流程是线性的， url传来之后，检查之后，可保持原来的流程继续向下执行，被下一个filter, servlet接收等.

2.java的拦截器 主要是用在插件上，扩展件上比如 hivernate spring struts2等 有点类似面向切片的技术，在用之前先要在配置文件即xml文件里声明一段的那个东西。

