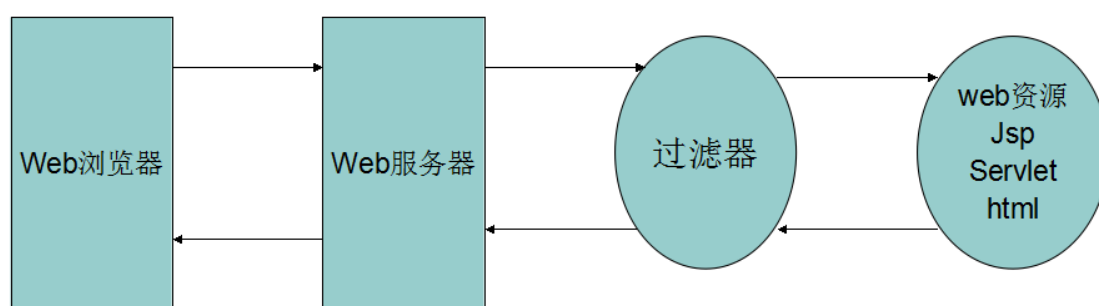


Java Web 三大组件之过滤器（Filter）

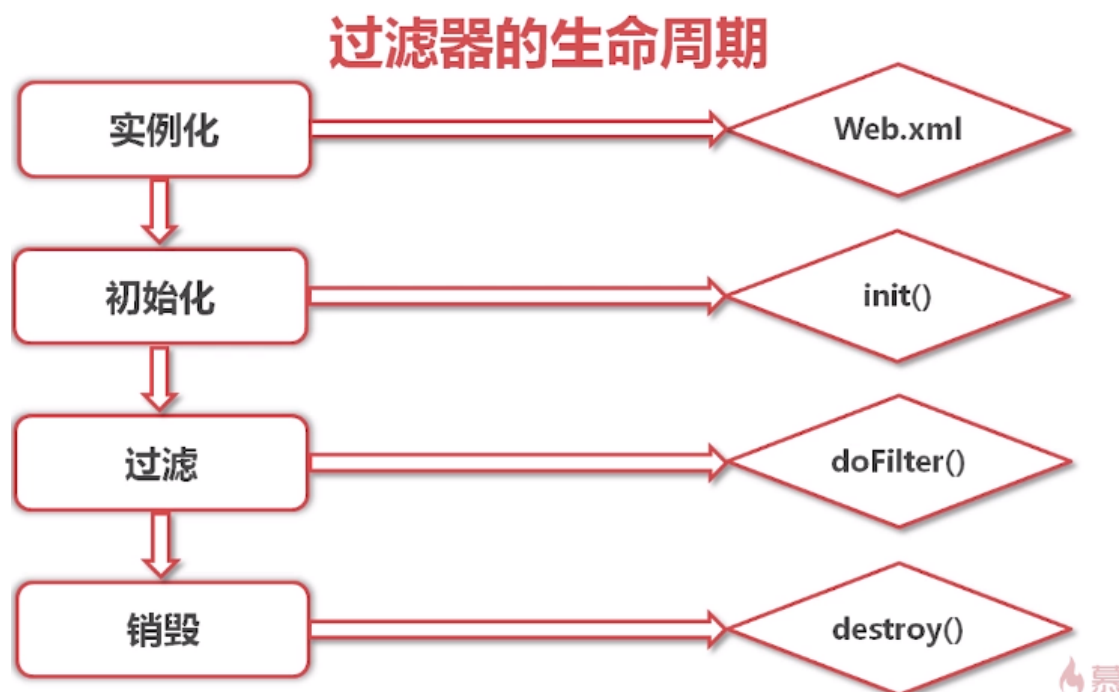
什么是过滤器？有什么用？

过滤器 JavaWeb 三大组件之一，它与 Servlet 很相似。不过过滤器是用来拦截请求的，而不是处理请求的。过滤，顾名思义，就是留下我们想要的，丢掉我们不需要的。例如：某个网站的用户通过/login.jsp 登录成功后会跳转到/pages/user/index.jsp 页面。但是如果用户并没有通过/login.jsp 页面登录，而是直接请求/pages/user/index.jsp 页面，显然这样做是没道理的，应该阻止这种行为。过滤器就是负责处理这种情况的，过滤器收到/pages/user/index.jsp 请求的时候，会将该请求先拦截下来，然后判断是不是处于登录状态的用户，如果是就允许这种请求，放行。如果是非登录状态，则不会放行这个请求，相当于过滤掉了。

过滤器的工作原理



过滤器的生命周期



Filter 的创建和销毁由 **WEB 服务器** 负责。web 应用程序启动时，web 服务器将创建 **Filter 的实例对象**，并调用其 **init** 方法，完成对象的初始化功能，从而为后续的用户请求作好拦截的准备工作，**filter 对象只会创建一次**，**init 方法也只会执行一次**。通过 **init** 方法

的参数，可获得代表当前 filter 配置信息的 FilterConfig 对象。Web 容器调用 destroy 方法销毁 Filter。**destroy 方法在 Filter 的生命周期中仅执行一次。**在 destroy 方法中，可以释放过滤器使用的资源。**每一次请求都会执行一次 doFilter 方法。**

Filter 的应用

- 1、定义 XxxFilter 类，实现 Filter 接口，重写 init()、doFilter()、destroy()方法。多数情况下只用具体实现 doFilter()方法即可。
- 2、在 web.xml 文件中使用<filter>和<filter-mapping>元素对编写的 filter 类进行注册，并设置它所能拦截的资源。Servlet3.0 提供@WebFilter 注解将一个实现了 javax.servlet.Filter 接口的类定义为过滤器，这样我们在 web 应用中使用过滤器时，也不再需要在 web.xml 文件中配置过滤器的相关描述信息了。

2 过滤器的入门

Java Web的三大组件（1）都需要交给web服务器运行（2）在web.xml文件中配置（3）

1. Servlet: javax.servlet.Servlet 通过HTTP协议接收客户端的请求，并且做出响应的一个Java应用程序。
2. Filter过滤器: javax.servlet.Filter 是一个接口，过滤请求，实现请求的拦截或者放行，并且添加新的功能。
3. Listener 监听器: javax.servlet.XxxListener，用来监听Web容器中各种域的事件

2.2 过滤器的作用：

1. 作用：

通过Filter技术，对Web服务器管理的所有Web资源：例如Jsp, Servlet, 静态图片文件或静态 html 文件等进行拦截，并且添加功能，从而实现一些“共同”的功能。

2. 例如：

- 1) 实现URL级别的权限访问控制
- 2) 过滤敏感词汇
- 3) 自动登录
- 4) 压缩响应信息等一些功能。

2.3 过滤器编写步骤：

特点：过滤器不是用户主动调用的，而是根据规则自己执行

1. 编写一个 java 类，实现 Filter 接口，并实现其中的所有方法

2. 在 web.xml 文件中配置 Filter

3. 把 Filter 部署到 tomcat 服务器运行

4) 运行结果：

1.请求的时候运行一次过滤器

2.运行了 Web 资源

3.响应的时候运行一次过滤器

2.4 Filter是如何实现拦截的？

Filter接口中有一个doFilter方法，当开发人员编写好Filter，并配置对哪个Web资源进行拦截后，Web服务器每次在调用Web资源的service方法之前，都会先调用一下filter的doFilter方法，因此，在该方法内编写代码可达到如下目的：

1) 调用目标资源之前，执行一段代码，request执行过滤任务。

2) 是否调用目标资源（即是否让用户访问Web资源）。

Web服务器在调用doFilter方法时，会传递一个filterChain对象进来，filterChain对象是filter接口中最重要的一個对象，它也提供了一个doFilter方法，开发人员可以根据需求决定是否调用此方法，调用该方法，则Web服务器就会调用Web资源的service方法，即Web资源就会被访问，否则Web资源不会被访问。

3) 调用目标资源之后，执行一段代码，response执行过滤任务。

2.5 过滤器的生命周期：

1. 过滤器加载的时机：

过滤器在Web服务器启动的时候就加载了，因为要拦截Web资源，所以必须在所有Web资源启动之前就加载过滤器。

2. 生命周期的方法：

1). init方法：

在创建完过滤器对象之后被调用。只执行一次

注：过滤器在Web服务器中也是单例模式

2). doFilter方法：

执行过滤任务方法。执行多次。

3). destroy方法：

Web服务器停止或者Web应用重新加载，销毁过滤器对象。