# Servlet Filter Listener

# 一 过滤器（Filter）

## 1 什么是过滤器

Filter属于Servlet的技术，通过Filter可以对web服务器管理的所有web资源，如jsp, Servlet, 静态图片文件或静态 html 文件等进行拦截，从而实现一些特殊的功能。

过滤器能够对Servlet容器的请求和响应对象进行检查和修改。

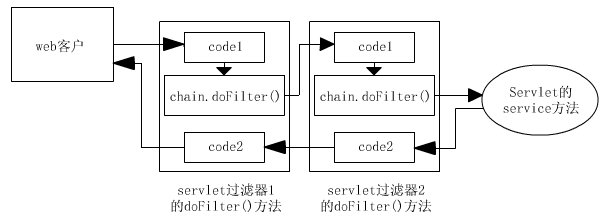
过滤器本身并不产生请求和响应对象，它只能提供过滤作用。

过滤器能够在web资源被调用之前检查Request对象，修改Request Header和Request内容；在Servlet被调用之后检查Response对象，修改Response Header和Response内容。

## 2 为什么使用过滤器

访问资源时，针对不同的资源可能会执行很多相同的逻辑操作，如编码的设置，登录权限的判断等，借助Filter中的代码复用，可以提高开发效率

## 3 过滤器执行流程



# 二 过滤器基本用法

## 1使用步骤

1)创建Filter类

自定义Filter类实现Filter接口，重写相关方法

2)Filter注册

在web.xml中对Filter进行注册，设置拦截的资源，一般设置为”/\*”

## 2 过滤器的生命周期

1）init

Tomcat服务器启动时，如果应用中存在filter，会创建filter对象，并调用其初始化方法。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

2）doFilter

通过chain.doFilter进行实际的过滤操作，可以访问后续的资源，访问资源后，会回到doFilter方法中

3）当应用重新加载，或者关闭tomcat服务器时，会销毁filter对象

## 3 过滤器链

在一个web应用中，可以开发编写多个Filter，这些Filter组合起来称之为一个Filter链

在doFilter方法中，开发人员如果调用了chain.doFilter方法，web服务器会检查FilterChain对象中是否还有其他的filter，如果有，则调用依次调用其余的filter，如果没有，则调用访问的目标资源。

注意：访问filter的先后顺序，由filter在web.xml中的注册顺序决定

过滤器配置

<filter>

<display-name>Filter2</display-name>

<filter-name>Filter2</filter-name>

<filter-class>com.rr.filter.Filter2</filter-class>

</filter>

<filter-mapping>

<filter-name>Filter2</filter-name>

<url-pattern>\*.jsp</url-pattern>

</filter-mapping>

<filter>

<display-name>Filter1</display-name>

<filter-name>Filter1</filter-name>

<filter-class>com.rr.filter.Filter1</filter-class>

</filter>

<filter-mapping>

<filter-name>Filter1</filter-name>

<url-pattern>/\*</url-pattern>

<!--

dispatch

REQUEST 默认值

FORWARD 转发的时候调用过滤器

ERROR

INCLUDE

-->

<dispatcher>REQUEST</dispatcher>

</filter-mapping>

# 三 应用

1 统一编码

2 访问权限控制