**Filter简介**

**Filter也称之为过滤器，它是Servlet技术中最实用的技术，Web开发人员通过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，从而实现一些特殊的功能。例如实现URL级别的权限访问控制、过滤敏感词汇、压缩响应信息等一些高级功能。**

**它主要用于对用户请求进行预处理，也可以对HttpServletResponse进行后处理。使用Filter的完整流程：Filter对用户请求进行预处理，接着将请求交给Servlet进行处理并生成响应，最后Filter再对服务器响应进行后处理。**



**如何借助Filter实现拦截功能?**

**Servlet API提供了一个Filter接口，编写的过滤器必须实现该接口。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过滤的目的.**

**Filter开发两步走**

**①编写java类实现Filter接口，并实现其doFilter方法。**

**②在web.xml文件中对编写的filter类进行注册，并设置它所能拦截的资源。**

**案例1:对Filter的整个生命周期了解的一个案例,加上初始化参数**

**Filter的生命周期(Filter的创建和销毁由web服务器控制)**

**①服务器启动的时候，web服务器创建Filter的实例对象，并调用其init方法，完成对象的初始化功能。filter对象只会创建一次，init方法也只会执行一次。**

**②拦截到请求时，执行doFilter方法。可以执行多次。**

**③服务器关闭时，web服务器销毁Filter的实例对象。**

**web.xml配置常用的各节点介绍：**

**<filter>指定一个过滤器。**

**<filter-name>用于为过滤器指定一个名字，该元素的内容不能为空。**

**<filter-class>元素用于指定过滤器的完整的限定类名。**

**<init-param>元素用于为过滤器指定初始化参数，它的子元素<param-name>指定参数的名字，<param-value>指定参数的值。这些参数可以中Filter实现类里拿到!在过滤器中，通过FilterConfig接口对象来访问初始化参数。**

**<filter-mapping>元素用于设置一个 Filter 所负责拦截的资源。一个Filter拦截的资源可通过两种方式来指定：Servlet 名称和资源访问的请求路径**

**<filter-name>子元素用于设置filter的注册名称。该值必须是在<filter>元素中声明过的过滤器的名字**

**<url-pattern>设置 filter 所拦截的请求路径(过滤器关联的URL样式)**

**<dispatcher>指定过滤器所拦截的资源被 Servlet 容器调用的方式，可以是REQUEST,INCLUDE,FORWARD和ERROR之一，默认REQUEST。用户可以设置多个<dispatcher>子元素用来指定 Filter 对资源的多种调用方式进行拦截。**

**<dispatcher>子元素可以设置的值及其意义**

**REQUEST：当用户直接访问页面时，Web容器将会调用过滤器。如果目标资源是通过RequestDispatcher的include()或forward()方法访问时，那么该过滤器就不会被调用。**

**INCLUDE：如果目标资源是通过RequestDispatcher的include()方法访问时，那么该过滤器将被调用。除此之外，该过滤器不会被调用。**

**FORWARD：如果目标资源是通过RequestDispatcher的forward()方法访问时，那么该过滤器将被调用，除此之外，该过滤器不会被调用。**

**ERROR：如果目标资源是通过声明式异常处理机制调用时，那么该过滤器将被调用。除此之外，过滤器不会被调用。**

**Filter对象——FilterConfig**

**用户在配置filter时，可以使用<init-param>为filter配置一些初始化参数，当web容器实例化Filter对象，调用其 init方法时，会把封装了filter初始化参数的filterConfig对象传递进来。**

**因此开发人员在编写filter时，通过 filterConfig对象的方法，就可获得：**

**①String getFilterName()：得到filter的名称。**

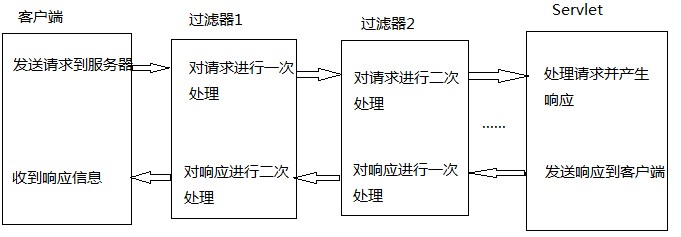
**②String getInitParameter(String name)：返回在部署描述中指定名称的初始化参数的值。如果不存在返回null.**

**③Enumeration getInitParameterNames()：返回过滤器的所有初始化参数的名字的枚举集合。**

**④public ServletContext getServletContext()：返回Servlet上下文对象的引用。**

**过滤器链——FilterChain**

**一个web应用中有很多个过滤器对某些web资源进行拦截，那么这组过滤器就称为过滤器链。过滤器的执行顺序和<filter-mapping>有关（谁在前先执行谁）和过滤器在web.xml文件中的配置顺序无关。**



**案例2:Filter链示例**