# Filter

## 一、其它内容

1.编码：Eclipse的workspace编码(即java代码的编码)、jsp编码、html编码、响应头编码(content-type，告诉浏览器以哪种格式打开页面)。

request.setCharacterEncoding(“utf-8”)：设置post请求的编码

response.setCharacterEncoding(“utf-8”)：设置响应内容的编码

response.setContentType(“text/html;charset=utf-8”)：设置响应头的编码，只适用于输出流直接响应(PrinteWriter)，跳转(服务器跳转、客户端跳转)是不用设置这个的。

2. uri和url：

uri：是包括项目名和资源名称，不包括协议、主机名和端口号。

url：是全路径，包括协议、主机号、端口号、项目名、资源名等。

## 二、Filter(过滤器)

作用：先于servlet或者其他请求(如jsp页面等)之前执行的，可以处理请求也可以给出响应。

导包：**import** javax.servlet.Filter;

Filter是一个接口，有两个default方法(init()和destroy()方法)，和一个抽象的doFilter()方法。

**创建一个Filter类：implements Filter，重写doFilter()方法，然后在web-xml中配置(或用@WebFilter注解的方式配置括号内可写拦截路径)**。

doFilter()方法中有三个参数，分别为ServletRequest 对象、ServletResponse对象、FilterChain对象。

### 1.配置Filter

在web.xml中配置Filter的方式和配置Servlet的类似，注意Filter配置的url-pattern是拦截路径，有三种方式：

①**/\***：拦截所有的请求

②**\*.文件扩展名**：拦截指定扩展名的所有请求，如\*.jsp表示拦截所有以.jsp结尾的文件的请求。

③**/目录1/\***：拦截指定目录下的所有请求，如/menu/\*表示拦截menu目录下所有的请求，该目录可以是虚拟的也可以是实际存在的目录可以是静态中的也可以是动态中的目录。(如访问Servlet的路径/virtual/loginServlet中第一个目录是虚拟的)。

拦截路径可以配置多个，表示可以拦截多个符合要求的请求。下面表示可以拦截.jsp文件的请求，也可以拦截menu1目录下的所有请求。



### 2.Filter的生命周期

**①创建阶段：服务器(tomcat)启动时，就创建Filter的对象，调用init方法完成初始化阶段**。Servlet对象的创建在默认情况下则是第一次访问Servlet时才会创建。**注意Filter、Servlet对象创建过程的区别**。

**②运行阶段**：每次发送请求并被Filter拦截到时，调用doFilter方法。Filter也是一个单实例对象。

**③销毁阶段**：服务器(web容器)正常关闭，或者项目卸载时，销毁Filter对象，销毁之前调用destory方法。

### 3.FilterChain请求链

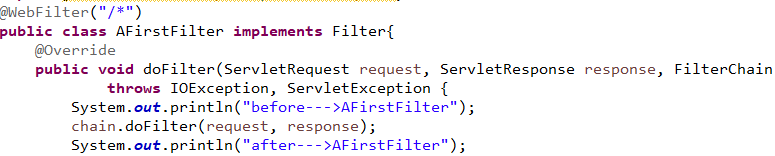
FilterChain请求链对象是用来管理请求的，请求可以来自Filter/Servlet/jsp/html等。**注意在doFilter()方法中一定不能忘了将请求放行：chain.doFilter(req,resp)**。

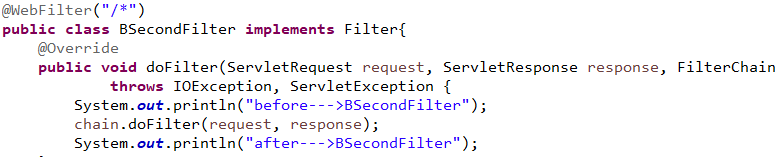
FilterChain是一个接口，其只有一个方法：doFilter(ServletRequest req，ServletResponse resp)的抽象方法。

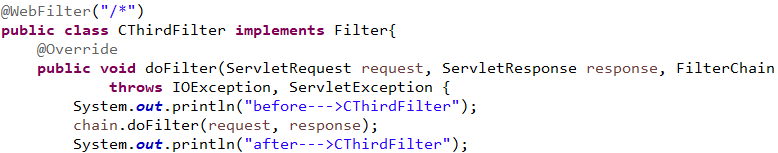
#### ⑴Filter和 doFilter的执行过程

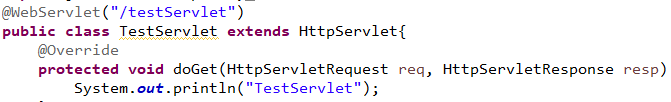
**Filter的执行顺序：**①**在web.xml中配置Filter，Filter的执行顺序和配置的顺序一致**。②**注解的方式@WebFilter配置Filter，Filter的执行顺序和Filter的类名有关，按照类名对应的字母顺序执行**。

实例：

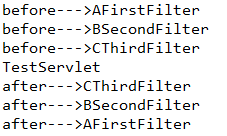


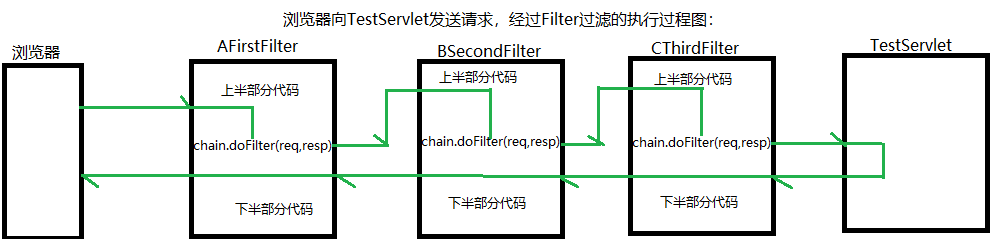






在浏览器中输入地址....../testServlet，在控制台得到的结果为：

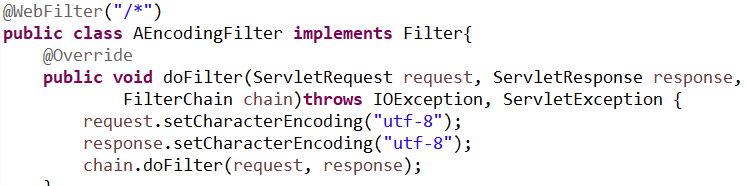




## 三、Filter的应用

**一般先将Filter写好再写其它的Servlet和jsp等页面。**

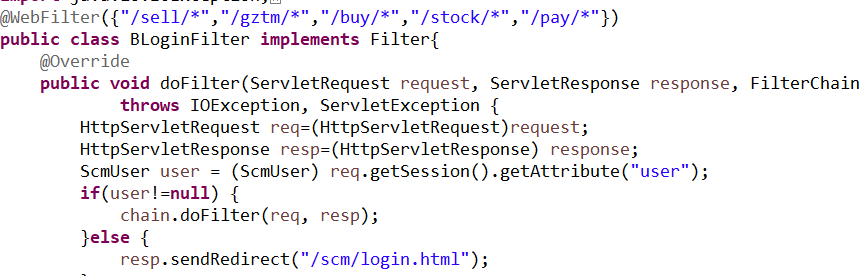
**1.编码过滤**：这个一般写在最前面，拦截所有的请求。



**2.登录过滤**：登录页面的请求不能拦截、登录的Servlet的请求不能拦截。只能拦截哪些需要登录之后才能进行操作发送的请求，如加入购物车、对购物车中的内容进行操作。

**一定要提前规划好请求的路径， 把需要登录后才能访问的请求放在同一个目录下，不需要登录就能访问的放在其他目录下**。

注解的方式设置多个拦截路径用{}括起来，每个拦截路径之间用逗号分开。



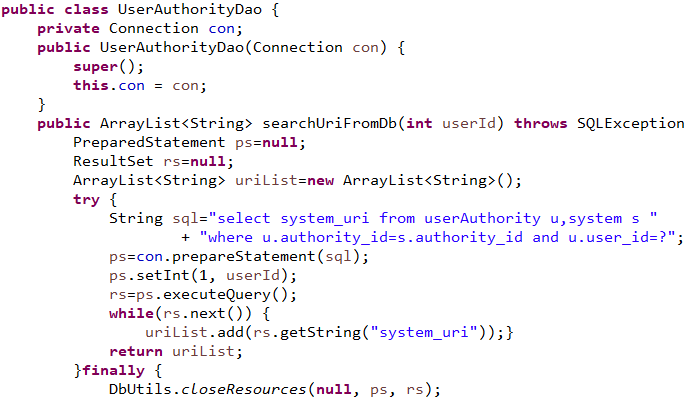
**3.权限过滤**：进行某些需要特殊权限的操作时，拦截用户是否具有某些权限，若有则放行请求。

下面代码是在Filter中查询用户的权限，老师要求改为在filter中先从session中取权限，若取不到则查询后保存在session中，不会。

①Filter代码：



②dao包代码：



③service包代码：

