

传智播客

《JavaWeb 程序开发进阶》

教学设计

课程名称： JavaWeb 进阶

授课年级： 2015 年级

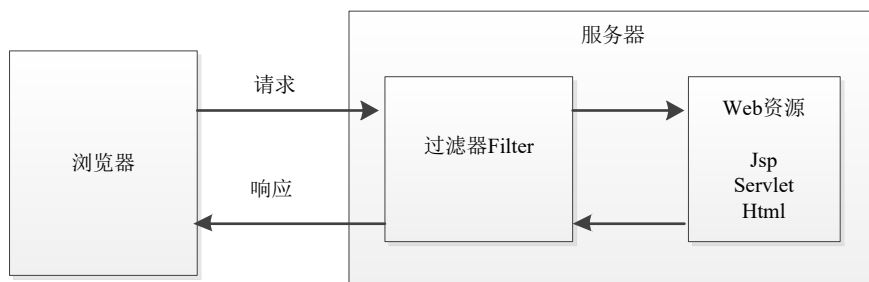
授课学期： 2015 学年第二学期

教师姓名： 某某老师

2014 年 08 月 01 日

课题名称	第4章 Filter（过滤器）	计划学时	6 课时
内容分析	在 Web 开发过程中，为了实现某些特殊的功能，经常需要对请求和响应消息进行处理。例如记录用户访问信息，统计页面访问次数，验证用户身份等。Filter 作为 Servlet2.3 中新增的技术，可以实现用户在访问某个目标资源之前，对访问的请求和响应进行相关处理。接下来，本章将针对 Filter 进行详细地讲解。		
教学目标及基本要求	<ol style="list-style-type: none"> 1、掌握如何编写一个 Filter 程序 2、熟悉 Filter 的高级应用 3、了解什么是装饰设计模式 		
教学重点	<ol style="list-style-type: none"> 1、实现第一个 Filter 程序 2、Filter 映射 3、Filter 链 4、案例-Filter 实现用户自动登陆 5、Filter 实现统一全站编码 6、Filter 实现页面静态化 		
教学难点	<ol style="list-style-type: none"> 1、什么是装饰设计模式 2、案例-Filter 实现用户自动登陆 3、Filter 实现统一全站编码 4、Filter 实现页面静态化 		
教学方式	讲师课堂教学要以 PPT 讲授为主，并结合多媒体进行教学		
教学过程	<p style="text-align: center;">第一课时</p> <p style="text-align: center;">（什么是 Filter、实现第一个 Filter 程序、Filter 映射）</p> <p>一、创设情境，导入 Filter 概念</p> <p>（1）教师通过现实中的例子，引出 Filter 的概念</p> <p>教师通过开发中遇到的实际问题进行讨论：开发中我们经常需要记录用户访问信息，统计页面访问次数，验证用户身份等，那么我们如何完成这一的功能呢？ 提出问题。</p> <p>（2）进入主题，讲解 Filter 的概念</p> <p>进入主题，引出 Filter，讲述 Filter 作为 Servlet2.3 中新增的技术，其基本功能就是对 Servlet 容器调用 Servlet 的过程进行拦截，从而在 Servlet 进行响应处理前后实现一些特殊功能。</p> <p>（3）明确学习目标</p> <ol style="list-style-type: none"> 1、要求学生了解什么是 Filter。 2、如何使用 Filter。 2、要求学生了解什么是 Filter 映射。 <p>二、进入重点知识的讲解</p> <p>（1）Filter 在 Web 应用中的拦截过程图。</p> <p>教师讲述完 Filter 的概念，参考教材结合图例向学生描述 Filter 的拦截过程。可参考如下图</p>		

例：



最后向学生介绍 Filter 接口中定义的三个方法，以及各自的作用。具体如下（可参见教材表 4-1）：

- 1、init(FilterConfig filterConfig)
- 2、doFilter(ServletRequest request,ServletResponse response,FilterChain chain)
- 3、destroy()

(2) 如何实现一个 Filter 程序？

教师结合案例，带领学生演示 Filter 实现过滤拦截的过程。可参照如下步骤：

- 1) 在 Eclipse 中创建一个名为 chapter04 的 Web 工程，然后在该工程的 Java Resources/src 目录下创建 cn.itcast.chapter04.filter 包，并在该包下创建一个 MyServlet.java 程序。
- 2) 在 web.xml 文件中对 servlet 进行如下配置。
- 3) 拦截 MyServlet 程序。在 cn.itcast.chapter04.filter 包中创建过滤器 MyFilter，该类用于拦截 MyServlet 程序。
- 4) 在 web.xml 文件中进行配置，设置 MyFilter 所能拦截的资源。
- 5) 重启 Tomcat 服务器，在浏览器的地址栏中输入 URL 地址 `http://localhost:8080/chapter04/MyServlet`，访问 MyServlet。
- 6) 查看浏览器窗口中显示的输出信息，只显示了 MyFilter 的输出信息，并没有显示 MyServlet 的输出信息，说明 MyFilter 成功拦截了 MyServlet 程序。

(3) 什么是 Filter 映射？

上节课讲解了什么是 Filter，并实现了一个 Filter 程序，了解到 Filter 拦截的资源需要在 web.xml 文件中进行配置，即 Filter 映射。然后讲解 Filter 映射的两种方式：

1、使用通配符“*”拦截用户所有请求

Filter 的 <filter-mapping> 元素可以配置过滤器所有拦截的资源，如果想让过滤器拦截所有的请求访问，那么需要使用通配符“*”来实现，具体实例如下：

```
<filter>
  <filter-name>Filter1</filter-name>
  <filter-class>cn.itcast.chapter04.filter.MyFilter </filter-class>
</filter>
```

```
<filter-mapping>
  <filter-name>Filter1</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
```

2、拦截不同方式的访问请求

在 web.xml 文件的<filter-mapping>元素中有一个特殊的子元素<dispatcher>，该元素用于指定过滤器所拦截的资源被 Servlet 容器调用的方式，<dispatcher>元素的值共有四个，具体如下：（可参见教材对应知识点）

- REQUEST：当用户直接访问页面时，Web 容器将会调用过滤器。
- 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。
- □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。
- □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。

介绍完如上知识点，教师以 FORWARD 为例，分步骤向同学演示 Filter 对转发请求的拦截效果。

（可参见教材 例 4-3& 4-5）

三、归纳总结，随堂练习，布置作业

（1）对课堂上讲解的知识点进行总结，使用博学谷系统中的随堂练习题巩固本节课的知识点。

（2）让学生课下完成动手体验，编写一个 Filter 过滤器，并练习拦截四种不同的访问请求，加深对所学知识的理解，巩固本节的学习内容。

第二课时

（Filter 链、FilterConfig 接口）

一、回顾上节课内容，讲解 Filter 链的概念

（1）对上节课留的作业进行答疑。

（2）回顾上节课内容，引出本节课主题

上节课讲解了什么是 Filter，以及 Filter 映射。本节我们主要讲解 Filter 链，以及如何获取 Filter 程序在 web.xml 文件中的配置信息呢？

（3）明确学习目标

- 1、要求学生明确什么是 Filter 链以及它的工作原理。
- 2、学会使用 Filter 链拦截 MyServlet 的同一个请求。
- 3、熟悉 FilterConfig 接口中方法的使用。

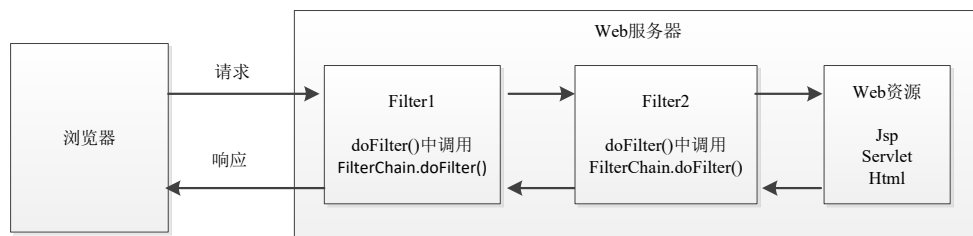
二、进行重点知识的讲解

(1) 什么是 Filter 链？

教师首先对 Filter 链的概念进行讲解：在一个 Web 应用程序中可以注册多个 Filter 程序，每个 Filter 程序都可以针对某一个 URL 进行拦截，如果多个 Filter 程序都对同一个 URL 进行拦截，那么这些 Filter 就会组成一个 Filter 链（也叫过滤器链）。

(2) Filter 链在的工作原理是什么？

教师进行讲解：Filter 链用 FilterChain 对象来表示，FilterChain 对象中有一个 doFilter() 方法，该方法作用就是让 Filter 链上的当前过滤器放行，请求进入下一个 Filter。最后结合图例向学生描述 Filter 链的拦截过程。图例如下：



需要注意的是：当 Web 服务器对请求做出响应时，也会被过滤器拦截，这个拦截顺序与之前相反，最终将响应结果发送给客户端。

介绍完如上知识点，教师可结合案例，分步骤向同学演示如何使用 Filter 链拦截 MyServlet 的同一个请求。（可参见教材 例 4-6& 4-7）

(3) FilterConfig 接口？

教师首先进行讲解 FilterConfig 接口的作用：该接口封装了 Filter 程序在 web.xml 中的所有注册信息，并且提供了一系列获取这些配置信息的方法。具体如下：（可参见教材 表 4-2）

- 1、String getFilterName ()
- 2、ServletContext getServletContext()
- 3、String getInitParameter(String name)
- 4、Enumeration getInitParameterNames()

介绍完上述方法后，教师以 getInitParameter(String name)方法为例，向同学演示该方法的使用。（可参见教材例 4-8）

三、归纳总结，随堂练习，布置作业

- (1) 对课堂上讲解的知识点进行总结，使用博学谷系统中的随堂练习题巩固本节课的知识点。
- (2) 让学生课下完成动手体验，如何使用 Filter 链拦截一个 Servlet 的同一个请求，并分别对 FilterConfig 接口中的方法进行练习，巩固本节的学习内容。

第三课时

(如何使用 Filter 实现用户的自动登录功能、装饰设计模式)

一、回顾上节课内容，引出装饰设计模式的概念

- (1) 对上节课留的作业进行答疑。
- (2) 回顾上节课内容，引出本节课主题

教师可对上节课所学知识点进行提问，让同学进行回答。教师总结上面所学的各个知识点，提出本节需要学习的知识：如何使用 Filter 实现用户的自动登录功能，以及什么是装饰设计模式和他的作用。

(3) 明确学习目标

- 1、能够使用 Filter 实现用户的自动登录功能。
- 2、熟悉装饰设计模式的原理和应用。

二、进行重点知识的讲解

(1) 如何使用 Filter 实现用户的自动登录功能？

教师引导学生进行回忆，我们之前实现的用户自动登录功能是通过什么技术实现的？然后学生进行回答，教师进行总结，虽然 Cookie 可以实现用户自动登录的功能，可是使用 Cookie 实现用户自动登录后，当客户端访问服务器的 Servlet 时，所有的 Servlet 都需要对用户的 Cookie 信息进行校验，这样势必会导致在 Servlet 程序中书写大量重复的代码。

那么为了解决上面的问题，可以在 Filter 程序中实现 Cookie 的校验。原理是因为 Filter 可以对服务器的所有请求进行拦截，因此，一旦请求通过 Filter 程序，就相当于用户信息校验通过，Servlet 程序根据获取到的用户信息，就可以实现自动登录了。

最后教师结合案例来演示如何使用 Filter 实现用户的自动登录功能。（可参见教材 例 4-9~4-14）

(2) 什么是装饰设计模式？

所谓装饰设计模式，指的是通过包装类的方式，动态增强某个类的功能。HttpServletRequestWrapper 和 HttpServletResponseWrapper 作为 request 和 response 对象的包装类，它们都采用了装饰设计模式。装饰设计模式的特点具体如下：

- □ 包装类要和被包装对象实现同样的接口。
- □ 包装类持有有一个被包装对象，例如在 HttpServletRequestWrapper 定义的构造方法中，需要传递一个 HttpServletRequest 类型的参数。
- □ 包装类在实现接口的过程中，对于不需要包装的方法原封不动地调用被包装对象的方法

来实现，对于需要包装的方法自己实现。

(3) 如何实现装饰设计模式？

了解了装饰设计模式的特点，教师结合案例，在 chapter04 工程的 cn.itcast.chapter04.decorator 包中，编写 PhoneDemo.java 程序，向学生演示装饰设计模式的实现。（可参见教材 例 4-15）

三、归纳总结，随堂练习，布置作业

(1) 让学生课下动手体验，完成使用 Filter 实现用户的自动登录功能的练习以及装饰设计模式功能的实现。

(2) 让学生课下完成动手体验，使用 Filter 实现用户的自动登录功能，以及装饰设计模式的使用。

第四课时

(Filter 实现统一全站编码、Filter 实现页面静态化)

一、回顾上节课内容，讲解 Filter 的高级应用

(1) 对上节课留的作业进行答疑。

(2) 回顾上节课的内容，引出本节课主题

在 Web 开发中，经常会遇到中文乱码问题，按照前面所学的知识，解决乱码的通常做法都是在 Servlet 程序中设置编码方式，但是，如果多个 Servlet 程序都需要设置编码方式，势必会书写大量重复的代码。为了解决上面的问题，我们可以在 Filter 中对获取到的请求和响应消息进行编码，从而统一全站的编码方式。

在实际开发中，有时为了提高程序性能、减轻数据库访问压力以及对搜索引擎的优化，可以使用 Filter 实现动态页面静态化。

(3) 明确学习目标

- 1、熟练掌握使用 Filter 实现统一全站编码的方式。
- 2、熟练掌握使用 Filter 实现动态页面静态化的功能。

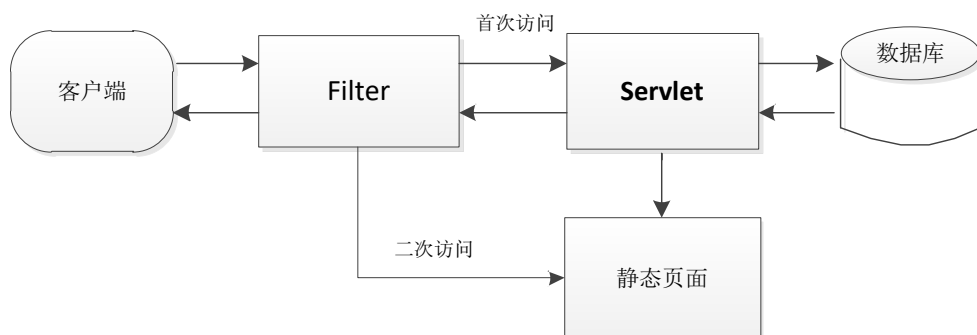
二、进行重点知识的讲解

(1) 如何使用 Filter 统一全站编码？

首先同学们已经了解为什么需要使用 Filter 实现统一全站编码方式的需求，然后教师结合案例分步骤讲解如何使用 Filter 统一全站的编码。（可参见教材 例 4-16~4-18）

(2) 什么是动态页面静态化？

教师首先介绍什么是页面静态化：页面静态化就是先于用户获取资源或数据库数据进而通过静态化处理，生成静态页面，所有人都访问这一个静态页面，而静态化处理的页面的访问速度要比动态页面快的多，因此程序性能会有大大的提升。然后结合教材中静态化页面执行流程图，具体如下：



向学生阐述静态化页面的执行过程：当客户端首次访问页面时，Filter 会自定义 response 输出缓存 HTML 源码，当客户端第二次访问页面时，就会直接访问静态页面，从而避免访问数据库。

最后教师通过一个显示图书分类的案例，分步骤向同学演示使用 Filter 实现页面静态化的步骤。（可参见教材 例 4-19~ 4-25）

三、归纳总结，随堂练习，布置作业

- (1) 让学生课下动手体验，完成使用 Filter 实现统一全站编码和 Filter 实现页面静态化的练习。
- (2) 让学生课下完成动手体验，练习使用 Filter 统一全站编码，以及使用 Filter 实现页面静态化。

第五课时

（上机练习）

上机练习主要针对本章中需要重点掌握的知识点，以及在程序中容易出错的内容进行练习，通过上机练习可以考察同学对知识点的掌握情况，对代码的熟练程度。

上机一：（考察知识点为使用 Filter 实现用户自动登录）

请按照以下要求设计一个自动登录的 Filter 程序。

要求如下：

- 1) 已知在客户端保存好了名为 " autologin " 的 cookie 用户信息，形式：username_password，即用户名与密码形式，之间用 " _ " 分隔。

	<p>2) 假定登录用户信息为：username 的值” itcast”，password 的值为” 123456”。</p> <p>3) 编写 AutoLoginFilter.java 的过滤器，实现自动登录功能。</p> <p style="text-align: center;">第六课时</p> <p style="text-align: center;">(上机练习)</p> <p>上机一：(考察知识点为装饰设计模式)</p> <p>请按照以下要求，使用装饰者设计模式，对 Bicycle 类进行简单包装。</p> <p>1) Bicycle 类表示普通自行车，它属于被包装类，MountainBike 类表示山地自行车，它是 Bicycle 类的包装类。</p> <p>2) 使用 MountainBike 类对 Bicycle 进行包装，使 Bicycle 类具有山地自行车的变速功能。</p> <p>上机二：(考察知识点为 Filter 实现统一全站的编码)</p> <p>分步骤讲解如何使用 Filter 实现统一全站的编码。</p>
<p>思考题和 习题</p>	
<p>教 学 后 记</p>	