



Servlet & JSP

学习笔记

林信良 编著

- 深入浅出讲述Web容器
- 涵盖SCWCD考试范围
- 实务教学经验集合
- Tomcat源代码分析
- 逐步重构的案例实训
- 课后测验与实训



内附CD

- 案例源代码
- 代码运行所需库文件
- 环境配置工具



清华大学出版社

TP312/3586D

2010

Servlet & JSP

学习笔记

林信良 编著

清华大学出版社

北 京

内 容 简 介

本书以“在线书签”项目贯穿全书,随着每一章的讲述都在适当的时候将 Servlet & JSP 技术应用于“在线书签”程序之中,并作适当修改,以了解完整的应用程序构建方法。

本书内容包括简单的 Web 应用程序,开发简单的 Servlet & JSP,容器管理,JSP 的使用,整合数据库等相关内容,本书适合 Servlet & JSP 初学者,以及广大的 Servlet & JSP 技术应用人员。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Servlet & JSP 学习笔记/林信良 编著. —北京:清华大学出版社,2010.4

ISBN 978-7-302-22153-1

I. S… II. 林… III. ①JAVA 语言—程序设计 ②JAVA 语言—主页制作—程序设计

IV. ①TP312 ②TP393.092

中国版本图书馆 CIP 数据核字(2010)第 031947 号

责任编辑:刘金喜 鲍 芳

封面设计:久久度文化

版式设计:孔祥丰

责任校对:胡雁翎

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京鑫丰华彩印有限公司

装 订 者:三河市李旗庄少明装订厂

经 销:全国新华书店

开 本:185×260 印 张:25.75 字 数:449 千字

附光盘 1 张

版 次:2010 年 4 月第 1 版 印 次:2010 年 4 月第 1 次印刷

印 数:1~5000

定 价:48.00 元

、 本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:034986-01

序

在完成本书之前，意外翻出了这张车票，94.08.15 从高雄到台北的座位证！一时间还想起不起这张座位证是哪来的，反倒是我老婆提醒了我，不过我却想起了更久之前的事情……

大学时代参加的社团是社会服务团，寒、暑假时会到一些地方举办营队，在学期中，即将参与营队的队友们必须负责各自的课程、准备教材、设计教具、验收教案等，出营队时则上台实行课程。

除了社团之外，自己平常也爱写些有的没的。大学时代正值 WWW 兴起之时，自己学会如何写 HTML，也常将学计算机时的心得写下来，放到 Web 上，像是如何安装 Apache、CGI 留言板之类的，说来写作的习惯应该是从那时养成的。

在大学最后一年考完硕士研究生入学考试之后，我在 BBS 的 Job 版上发现了几个短期打工需求，有天接到一个电话，问我想不想写书。虽然主题只是网页制作，但第一次要写一本完整的书，合同载明页数必须有 400 页以上，着实有很大的压力，甚至还因此失眠了好几次，所幸在当兵前夕完成了这本书，成了我的第一本著作。事后在市场上发现，这本书还有四次印刷，心里还蛮感到安慰的。

当兵期间所属的单位是学校，平时除了连队勤务或卫哨之外，所做的事就是协助教官编写教材、教案，上课担任助教等。退伍后的第一份工作是在高雄，公司的业务之一是出版计算机图书，因为早有写作及出版图书的经验，自然也在公司的名义之下写了几本书。

2003 年 3 月底，开始将一些东西以“良葛格学习笔记”的名称放在网络上，随着时间的积累，伴随着网络传播的力量，越来越多的人知道了这个网站的存在，也在网络上结交了许多朋友，且因此得以在 Javaworld@TW 前站长林上杰(Browser)先生的介绍下，认识了碁峰编辑江佳慧(Novia)小姐，出版了第二本印有个人名号的书籍。



想到这里，发现在我过去的经验中，怎么都跟上台、写作、课程有关？还有一点不知道是否也有相关，我岳父岳母也都是老师……

这就是看到 94.08.15 从高雄到台北的座位证时，突然涌出的一连串回忆……

1994 年 8 月 15 日是什么日子？隔两天就是“2005 Java TWO 社区大会”！这张车票是为了参加 Java TWO 大会而买的。这是我第一次参加 Java TWO 大会，目的之一是为了看看许多网络上认识的但未曾谋面的朋友，另一个原因是基峰也在大会上设摊，其中有卖我的书，想去看看反应如何……在大会上，碰到了王森(Moli)先生，他跟我说：“想要请你帮忙写个教材……”不过那时场面很混乱，反正就是一堆人哈啦来哈啦去的，话题很难继续，直到后来出现了个 Moli 先生想参加我的 MSN，哈啦过后，才确定这件事是真的！

之后又因为一连串的因缘机会，开始了我江湖卖艺……呃……讲课的日子！时光匆匆，岁月如梭，转眼来到了 2009 年 3 月，Novia 小姐问我有没有新的写作计划？我想了一下，这些日子以来，许许多多的授课经验积累了不少的想法，也了解了不少学员在实际学习时所遇到的问题，不如写下来吧！而这些写下来的东西就成了你眼前的这本书(篇幅有限，这本书只针对 Servlet / JSP)！

我不太知道人有没有宿命，但回首时总会发现许多的巧合，过去的种种经历，好像是在为了将来的某个事件而准备似的。当然，你也可以说，这是因为回忆是选择性地挑选拼凑而成的。无论如何，这些事情过去总得发生过，未来的你才有的拼凑。

一张车票引发了一连串的记忆，也终于知道要怎么写这本书的序了，这本书就是这么来的……



导 读

本书中许多范例都使用了完整的程序来展现，如果是用以下方式示范程序代码：

FirstServlet

HelloServlet.java

```
package cc.openhome;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloServlet extends HttpServlet {  ←❶ 继承 HttpServlet
    @Override
        protected void doGet(HttpServletRequest request, ←❷ 重新定义 doGet()
                                HttpServletResponse response)
                                throws ServletException, IOException {
        PrintWriter out = response.getWriter();
        out.println("Hello!World!");
    }
}
```

左上角的 **FirstServlet** 表示可以在光盘中找到对应的 **FirstServlet** 项目，右上角的 **HelloServlet.java** 表示可以在项目中找到 **HelloServlet.java** 文件。如果程序代码中出现标号与提示文字，表示后续的内文中会有对应于标号及提示的更详细说明。

如果使用以下方式呈现程序代码，则表示它是一个完整的程序内容，不是项目的一部分，主要用来展现如何编写一个完整的文件。

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<html>
<head>
    <title>SimpleJSP</title>
</head>
<body>
    <h1><%= new java.util.Date() %></h1>
```

```
</body>
</html>
```

如果使用以下方式呈现程序代码，则表示它是个程序代码片段，主要展现程序编写时需要特别注意的片段。

```
// ...
public void _jspService(HttpServletRequest request,
                        HttpServletResponse response)
    throws java.io.IOException, ServletException {
// ...
try {
    response.setContentType("text/html;charset=UTF-8");
    //...
    out = pageContext.getOut();
    // ...
} catch (Throwable t) {
    // ...
} finally {
    // ...
}
```

由于受书籍页面宽度的限制，有些过长的程序代码可能会在一行容纳不下，不得不隔行表示，此时会使用箭头符号表示两行实际上是必须连接在一起的。例如：

JDBCDemo

context.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Context antiJARLocking="true" path="/JDBCDemo">
    <Resource name="jdbc/demo"
        auth="Container" type="javax.sql.DataSource"
        maxActive="100" maxIdle="30" maxWait="10000" username="root"
        password="123456" driverClassName="com.mysql.jdbc.Driver"
        url="jdbc:mysql://localhost:3306/demo?
        useUnicode=true&characterEncoding=UTF8"/>
</Context>
```

在上面的程序片段中，在 url 属性的设置中，完整的设置其实是 jdbc:mysql://localhost:3306/demo?useUnicode=true&character Encoding=UTF8，当中不可以中断。

在书中会出现类似以下的特殊段落：

提示 针对课程中所提到的概念，提供一些额外的资料或思考方向，暂时忽略这些提示对课程的进行并没有影响，但有时间的话，应针对这些提示多作阅读、思考或讨论。

注意 针对课程中所提到的概念，以特殊段落的方式特别呈现出必须注意的一些使用方式、陷阱或避开问题的方法，看到这个特殊段落时请集中精神阅读。

本书以“在线书签”项目贯穿全书，随着每一章的进行，都会在适当的时候将新学到的技术应用到“在线书签”程序之中，并作适当的修改，以了解完整的应用程序基本上是如何建构出来的。

书附光盘中包括本书中所有的范例，提供了 NetBeans 及 Eclipse 两个项目结构，附录 A、B 分别包括了 NetBeans 与 Eclipse 的入门说明。

本书也说明了如何在 Web 应用程序中整合数据库，列举范例时所使用的数据库为 MySQL，附录 C 包括了 MySQL 的入门简介。

本书涵盖了 SCWCD 考试范围(除了第 11 章之外，因为 JDBC 不是 SCWCD 的考试范围)，每章最后的“重点复习”是针对该章的重要提示，可在考前复习时使用。每章都会有“课后练习”，与认证相关的是选择题，分为单选、多选题两种形式。

实现题则是与每个章节相关的程序练习。为了对应课文中的“在线书签”，实现题中也有个“访客留言板”项目贯穿全书。书附光盘不提供课后练习解答及实现题的参考程序代码，若需要，可以到作者网站下载：<http://openhome.cc> 或从 <http://www.tupwk.com.cn/downpage> 下载。

目 录

第 1 章 简介 Web 应用程序	1
1.1 Web 应用程序架构	2
1.1.1 Web 网站的组成	2
1.1.2 HTTP 通信协议	5
1.1.3 静态页面和动态页面	8
1.1.4 客户端程序和服务器端程序	10
1.2 Servlet/JSP 简介	11
1.2.1 Servlet 的运行概念	11
1.2.2 Servlet 与 JSP 的关系	14
1.2.3 Java EE 简介	17
1.3 重点复习	19
课后练习	20
第 2 章 开发简单的 Servlet & JSP	23
2.1 从 Servlet 到 Web 容器	24
2.1.1 从 Servlet 的编写认识 HttpServlet	24
2.1.2 设置部署描述文件	27
2.1.3 Web 应用程序文件组织	29
2.1.4 WAR 的创建与部署	30
2.2 在 HelloServlet 之后	31
2.2.1 Web 容器的作用	32
2.2.2 MVC 与 Model 2 简介	34
2.2.3 简单的 Model 2 程序	37
2.3 重点复习	40
课后练习	42
实训题	44
第 3 章 HTTP 请求、响应处理	45
3.1 HttpServletRequest 的使用	46
3.1.1 请求信息的取得	46

3.1.2	中文处理问题	50
3.1.3	请求转发与包含	53
3.1.4	ServletRequest 属性	54
3.2	HttpServletResponse 的应用	56
3.2.1	对浏览器进行响应	56
3.2.2	sendRedirect() 与 sendError()	58
3.2.3	ServletOutputStream 的使用	59
3.3	综合练习/在线书签	61
3.3.1	在线书签应用程序分析	61
3.3.2	查看书签	62
3.3.3	添加书签	64
3.4	重点复习	68
	课后练习	70
	实训题	72
第 4 章	ServletConfig 与 ServletContext	73
4.1	关于 ServletConfig	74
4.1.1	Servlet 生命周期	74
4.1.2	Servlet 初始参数的设置和取得	76
4.2	关于 ServletContext	79
4.2.1	ServletContext 简介	79
4.2.2	ServletContext 初始参数	82
4.2.3	ServletContextListener	82
4.2.4	ServletContext 属性	84
4.3	综合练习/在线书签	85
4.3.1	书签记录文件的加载	86
4.3.2	修改“查看书签”功能	87
4.3.3	修改“添加书签”功能	88
4.3.4	动态“添加书签”窗体	91
4.4	重点复习	94
	课后练习	96
	实训题	99

第 5 章 会话管理	101
5.1 会话管理的基本原理	102
5.1.1 隐藏字段的使用	102
5.1.2 Cookie 的使用	105
5.1.3 URL 重写的使用	110
5.2 HttpSession 会话管理	112
5.2.1 HttpSession 的使用	112
5.2.2 HttpSession 会话管理原理	116
5.2.3 HttpSession 与 URL 重写	118
5.3 重点复习	120
课后练习	122
实训题	124
第 6 章 过滤器与封装器	125
6.1 过滤器服务	126
6.1.1 过滤器的概念	126
6.1.2 过滤器的实现	128
6.1.3 过滤器的设置	131
6.2 搭配封装器	133
6.2.1 请求封装器	133
6.2.2 响应封装器	137
6.3 综合练习 / 在线书签	141
6.3.1 书签服务对象的实现	141
6.3.2 在线书签应用程序的重构	144
6.3.3 登录验证过滤器的实现	147
6.3.4 字符过滤器的使用	149
6.4 重点复习	150
课后练习	152
实训题	155
第 7 章 监听器、部署描述设置	157
7.1 监听器	158
7.1.1 生命周期监听器	158
7.1.2 属性操作监听器	162

7.1.3 会话属性类监听器	163
7.2 部署描述设置	166
7.2.1 Servlet 的设置	166
7.2.2 设置欢迎与错误处理页面	169
7.2.3 MIME 与扩展名的对应	170
7.3 重点复习	171
课后练习	173
实训题	175
第 8 章 使用 JSP	177
8.1 从 JSP 到 Servlet	178
8.1.1 JSP 生命周期	178
8.1.2 指示元素	181
8.1.3 声明、Scriptlet 与表达式元素	184
8.1.4 注释元素	189
8.1.5 隐含对象	189
8.1.6 错误处理	192
8.2 标准标签	196
8.2.1 <jsp:include>与<jsp:forward>标签	197
8.2.2 <jsp:useBean>、<jsp:setProperty>与<jsp:getProperty>简介	198
8.2.3 深入<jsp:useBean>、<jsp:setProperty>与<jsp:getProperty>	200
8.2.4 Model 1 简介	204
8.2.5 其他标准标签	206
8.3 表达式语言(EL)	207
8.3.1 EL 简介	207
8.3.2 使用 EL 访问属性	210
8.3.3 EL 隐含对象	213
8.3.4 EL 运算符	214
8.3.5 EL 函数的自定义	215
8.4 综合练习/在线书签	218
8.4.1 使用 JSP 实现“查看书签”页面	218
8.4.2 使用 JSP 实现“添加书签”页面	220
8.4.3 调整 web.xml 与其他源代码	223
8.5 重点复习	226

课后练习	229
实训题	231
第 9 章 使用 JSTL 与自定义标签	233
9.1 JSTL 的使用	234
9.1.1 JSTL 简介	234
9.1.2 流程处理标签	236
9.1.3 错误处理标签	239
9.1.4 网页导入、重定向标签	241
9.1.5 属性处理与输出标签	242
9.1.6 URL 处理标签	245
9.2 Tag File 自定义标签	246
9.2.1 Tag File 简介	247
9.2.2 处理标签属性与主体	249
9.2.3 TLD 文件	252
9.3 Simple Tag 自定义标签	253
9.3.1 Simple Tag 简介	254
9.3.2 架构与生命周期	256
9.3.3 处理标签属性与主体	259
9.3.4 与父标签的沟通	263
9.3.5 TLD 文件	268
9.4 Tag 自定义标签	268
9.4.1 Tag 简介	269
9.4.2 架构与生命周期	271
9.4.3 重复执行标签主体	273
9.4.4 处理主体运行结果	275
9.4.5 与父标签的沟通	278
9.5 综合练习/在线书签	281
9.5.1 在“查看书签”页面使用 JSTL	282
9.5.2 在“添加书签”页面使用 JSTL	282
9.6 重点复习	283
课后练习	288
实训题	291

第 10 章 Web 容器安全管理	293
10.1 Web 容器的安全管理	294
10.1.1 Java EE 安全的基本概念	294
10.1.2 声明式基本验证	297
10.1.3 容器基本验证原理	302
10.1.4 声明式窗体验证	303
10.1.5 容器窗体验证原理	304
10.1.6 使用 HTTPS 保护数据	305
10.1.7 编程安全管理	309
10.2 综合练习/在线书签	310
10.2.1 使用容器窗体验证	310
10.2.2 设置服务器对应文件	312
10.3 重点复习	313
课后练习	315
实训题	317
第 11 章 整合数据库	319
11.1 使用 JDBC 连接数据库	320
11.1.1 JDBC 简介	320
11.1.2 数据库的连接	324
11.1.3 使用 DataSource 取得连接	328
11.1.4 Statement、ResultSet 的使用	332
11.1.5 PreparedStatement 的使用	337
11.2 综合练习/在线书签	339
11.2.1 重构/创建 DAO	340
11.2.2 重构/修改 BookmarkService	344
11.2.3 设置 JNDI 部署描述	346
11.2.4 添加“删除书签”功能	347
11.3 重点复习	349
课后练习	351
实训题	353
第 12 章 模式与重构	355
12.1 设计模式	356

12.1.1	Template Method 模式(Gof 设计模式)	356
12.1.2	Intercepting Filter 模式(Java EE 设计模式)	357
12.1.3	Model-View-Controller 模式(架构模式)	358
12.2	从重构中认识模式	360
12.2.1	Business Delegate 模式	360
12.2.2	Service Locator 模式	361
12.2.3	Transfer Object 模式	361
12.2.4	Front Controller 模式	362
12.3	重点复习	363
	课后练习	365
附录 A	NetBeans 入门	367
A.1	NetBeans IDE 与 Tomcat 的安装	368
A.2	Servlet 的编写	370
A.3	Web 应用程序的部署和运行	373
A.4	NetBeans 工程的关闭和打开	375
附录 B	Eclipse 入门	377
B.1	Eclipse 与 Tomcat 的安装	378
B.2	Servlet 的编写	380
B.3	Web 应用程序的导出和执行	383
B.4	Eclipse 工程的关闭、删除和导入	386
附录 C	MySQL 入门	389
C.1	MySQL 的安装和设置	390
C.2	MySQL 的数据类型	391
C.3	数据库和数据表的创建	393
C.4	CRUD 操作	394

简介 Web 应用程序

学习目标：

- ❖ 了解 Web 应用程序的基本架构
- ❖ 了解 HTTP 中 GET 与 POST 的差别
- ❖ 了解静态页面和动态页面
- ❖ 了解客户端程序和服务器端程序
- ❖ 了解 Servlet/JSP 运作原理
- ❖ 了解 Java EE 基本架构

1.1 Web 应用程序架构

网络应用程序基本上是客户(Client)/服务器(Server)架构, 你所需要的文件或相关资源都放置在服务器上, 用户通过客户端应用程序, 通常是个图形用户界面(Graphic User Interface, GUI)程序, 通过指定资源的位置来进行网络连接, 而后通过协议来沟通所要进行的动作或要求所需的资源。

可将这样的结构应用到 Web 应用程序架构上。服务器上所放置的文件基本上是使用 HTML(HyperText Markup Language, 超文本标识语言)格式来存放的, 它主要用于描述文件在客户端显示时的页面绘制或行为方式, 客户端应用程序通常是个浏览器(Browser), 浏览器使用 URL(Uniform Resource Locator, 统一资源定位符)指定文件的所在位置, 通过 HTTP(HyperText Transfer Protocol, 超文本传输协议)与服务器端沟通, 发送各种请求以取得不同的 HTML 响应。

HTML、URL 与 HTTP 的基本概念, 为本节接下来所要说明的内容。

1.1.1 Web 网站的组成

Web 的内容以文件的方式放置在 Web 网站上, 让世界各地的用户得以通过网络连接来取得文件。为了让全球各地的用户取得文件之后, 能够理解当中的内容, 或者更具体地说, 让客户端应用程序取得内容后, 可以了解如何显示页面及作出对应的行为反应, 文件格式必须先加以规范, 而这个规范文件内容的格式就是 HTML。

基本上, HTML 是以标签(Tag)的方式来定义文件结构与当中的每个元素的。例如, 一个简单的 HTML 范例如下所示:

```
<html>
<head>
  <title>HTML 范例文件</title>
</head>
<body>
  <center>
    哈啰! 请输入...<br>
    <br>
    <form method="get" action="sample.do" name="sample">
      名称: <input type="text" name="name"><br><br>
      <input type="button" value="送出">
    </form>
```