

Broadview®
www.broadview.com.cn

JSP 网络编程

傅进勇 邓少烽 李波 编著

学习编程

编程

网络

学习

JSP

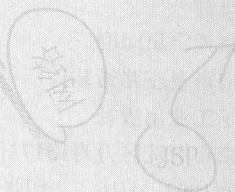


电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

JSP网络编程 学习笔记

傅进勇 邓少烽 李波 编著



電子工業出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书以JSP 2.0为基础,结合JSP和Servlet的最新规范,从基本的语法和规范入手,以恰当的实例为导向,对实例代码进行层层图解的方式,由浅入深地介绍JSP开发的基本理论知识及Web应用系统开发的实践经验。全书内容分为7篇,共23章,第一篇为“JSP应用开发基础”,主要讲述JSP技术背景情况、JSP技术的基本原理及开发环境的搭建;第二篇为“Servlet核心技术”,主要讲述Servlet的技术特点、请求和响应、数据共享、会话跟踪等技术;第三篇为“JSP技术基础”,主要讲述JSP的基本语法和内建对象及JSP和Servlet的集成,JavaBean和表达式语言;第四篇为“数据库访问技术”,主要讲述JDBC技术及JSP和Servlet如何通过JDBC访问数据库,以及如何改进数据库的访问和目前流行的Hibernate、iBATIS及Spring集成访问的支持;第五篇为“标签语言和表达式语言”,主要讲述JSP的标签技术,JSP提供的标准标签库JSTL的用法及用户如何自定义自己的标签库;第六篇为“Web应用高级专题”,主要讲述Servlet过滤器、JSP异常处理、JSP日志、认证和安全、部署等内容;第七篇为“Web应用开发实例”,围绕一个电子商务网站,从需求分析、架构选取、数据存储、开发、测试及部署等过程中阐述Web开发的步骤实例讲解。

本书适合于具有一定Java基础、欲入门JSP开发及Struts等Java Web开发的程序员阅读,本书知识点丰富,也可作为专业Java开发程序员的必备参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

JSP网络编程学习笔记 / 傅进勇, 邓少烽, 李波编著. 北京: 电子工业出版社, 2008.5
ISBN 978-7-121-05495-2

I. J… II. ①傅…②邓…③李… III. JAVA语言—主页制作—程序设计 IV. TP393.092

中国版本图书馆CIP数据核字(2007)第185041号

责任编辑: 高洪霞

印 刷: 北京天宇星印刷厂

装 订: 三河市皇庄路通装订厂

出版发行: 电子工业出版社

北京市海淀区万寿路173信箱 邮编 100036

开 本: 860×1092 1/16 印张: 38 字数: 949 千字 彩插: 2

印 次: 2008年5月第1次印刷

印 数: 5000 册 定价: 65.00 元(赠光盘1张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至zlts@phei.com.cn, 盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线:(010) 88258888。



本书独一无二的优势

本书由具有多年电信、金融、保险行业经验的IBM工程师和华为工程师执笔写作，他们在J2EE领域具有多年开发研究经验，并具有多年J2EE培训讲解经验，而且均有多套书籍面世发表，尤其熟悉Java Web开发理论知识体系。两位工程师的娴熟笔法，使得本书深入、广博、细腻，会将JSP Web开发技术展现得淋漓尽致。

本书内容新颖，基于目前JSP和Servlet最新版本，并结合J2EE最新规范，考虑JSP和Servlet历史版本兼容写作而成，使得读者不论基于JSP 1.2编程还是基于JSP 2.0编程，本书内容均适宜。本书内容全面，覆盖JSP和Servlet的核心技术点和重要知识点，囊括JDBC数据库操作，Hibernate、iBATIS、Spring流行框架技术，以及开源Web框架Struts、JavaServer Face技术；除此之外，本书将JSP Web应用开发的成套综合知识SOAP Web服务、Web应用分层及设计模式、国际化、异常处理、日志记录、调试技巧、测试部署等内容作重点讲解，这些专题基本囊括Java领域Web开发的各个层面。

本书和市场上其他类似书籍相比，具有下面与众不同的特色。

★形象

本书对书中的源代码采取绘图的讲解方式，使抽象变形象，让高深的编程理论“赤裸裸”地展现在你的面前，使你能够轻而易举地领悟编程奥秘。这是本书最大的特色。

★趣味

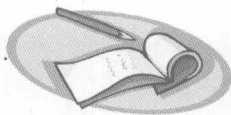
本书用拉链做素材，拉开学习的帷幕，用情趣似的学习笔记形式带你走进轻松的学习课堂，并且在书中让人感觉到烦躁的地方搭配以相关的图片，让你享受到学习中的乐趣，探索中的喜悦，胜利后的兴奋。

★实用

本书的实用性较强，以经验为后盾，以实践为导向，以实用为目标，深入浅出地讲解JSP开发中的种种问题。特别是，在讲解时非常注重实践与理论的形象结合，为了能使读者通过读懂源代码来掌握JSP技术，在实例举证时还给源代码绘制了大量的流程图。本书的所有源代码都已调试通过，并且放在了本书所附带的光盘中，读者拿来即可使用。

★全面

本书内容全面，从基本的语法入手，以恰当的实例为导向，由浅入深地讲解ASP技术的基本理论知识，所讲解的内容几乎囊括了JSP技术的所有知识点，涉及JSP异常处理、认证与安全等高级知识。



前言 Foreword



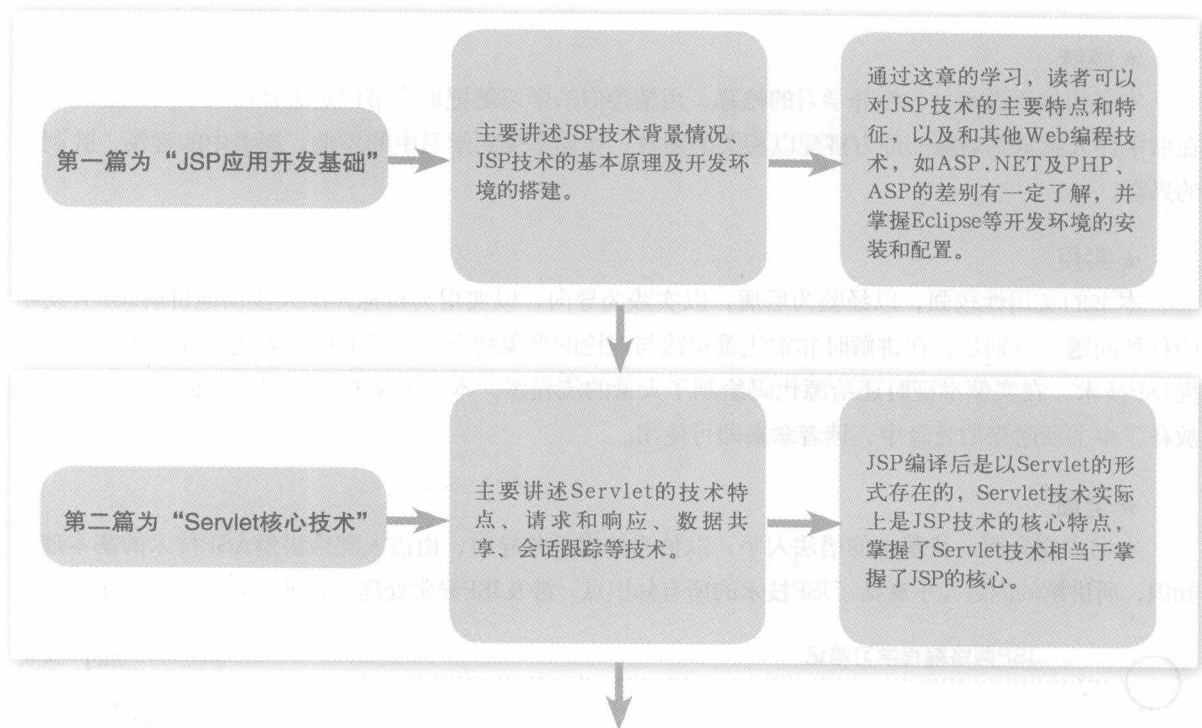
Java 2 Enterprise Edition(J2EE)技术自从推出以来得到广泛认可和应用,在J2EE技术中发展最迅猛的当属Java Server Page (JSP) 技术,在JSP 2.0规范中,为JSP的易用性进行了重新设计,使得不懂Java的页面设计人员也能够开发JSP应用,在开源社区的推动下,Struts、Hibernate、Spring等大量开源框架的出现为JSP技术的应用增添了更大的灵活性和空间。

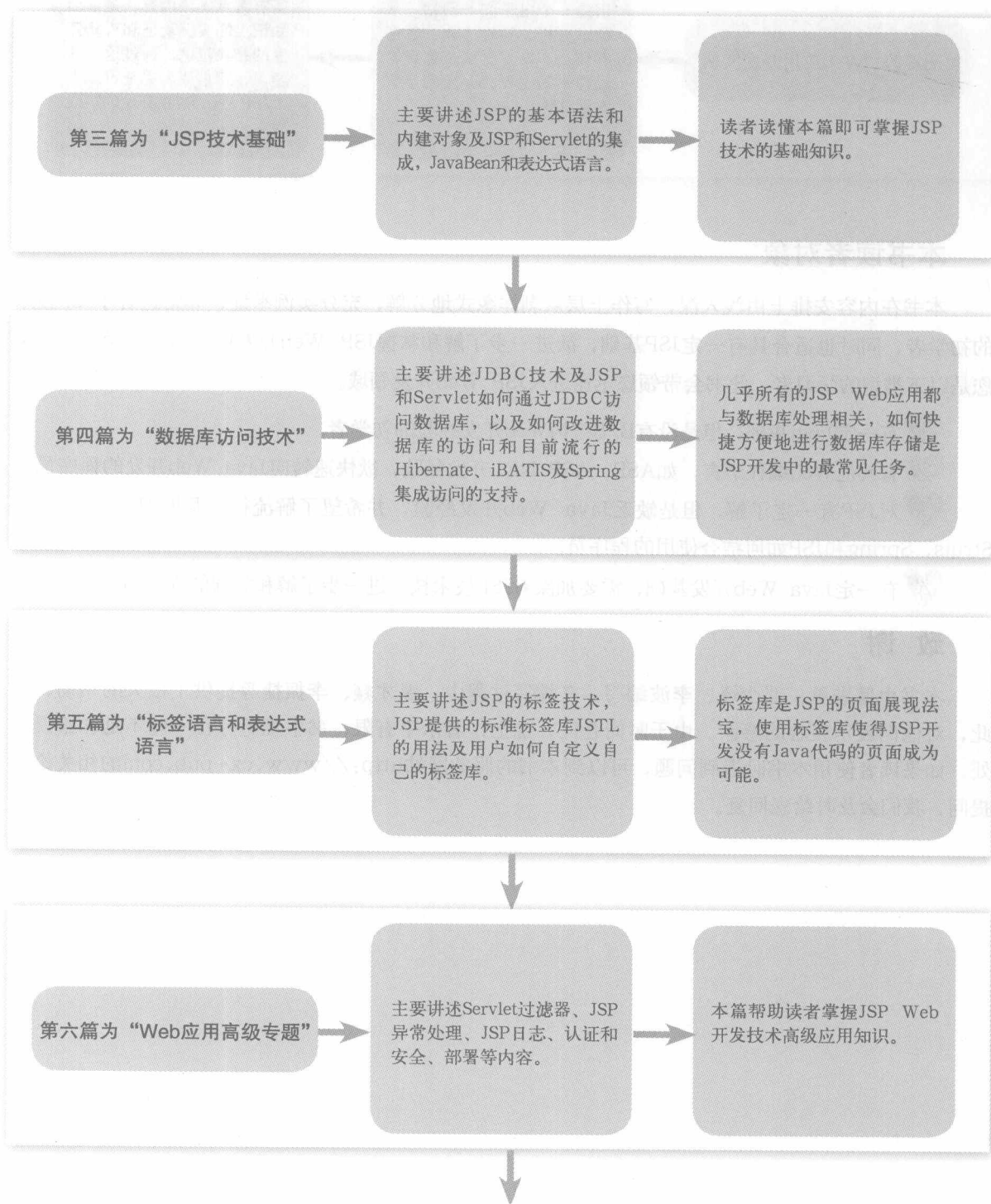
本书充分考虑到JSP Web应用开发的知识和技能需要,以及读者的知识结构,将JSP基本理论知识和Web应用开发经验知识融为一体,内容上做到全面而深入,囊括JSP和Servlet核心技术和精髓,以及Web应用开发整体解决方案知识,从浅入深,实例举证,以图文并茂的源代码图解和融实践于一体的理论分析为导向,对每一个知识点进行剥洋葱式的分析讲解,使得读者在了解JSP基础理论知识的同时获得成套Web应用系统的开发经验,带领读者慢慢步入JSP技术的殿堂,了解JSP工作原理的奥秘。

本书的内容结构

本书以JSP 2.0和Servlet 2.4为基础,结合J2EE的最新规范,从基本的语法和规范入手,以恰当的实例为导向,由浅入深地介绍JSP开发的基本理论知识及Web应用系统开发的实践经验。结构如下:

本书共分7篇23章





第七篇为“Web应用开发实例”

围绕一个电子商务网站，从需求分析、架构选取、数据存储、开发、测试及部署等过程中阐述Web开发的步骤实例讲解。

内容覆盖本书JSP Web开发所有重点领域内容，是前面知识点的实例验证和Web开发的经典范例，通过这章的学习，相信读者完全可以走入JSP Web应用开发的实践殿堂。

本书读者对象

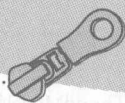
本书在内容安排上由浅入深，写作上层层剥洋葱式地分解，充分实例举证，非常适合于JSP技术的初学者，同时也适合具有一定JSP基础，欲进一步了解和掌握JSP Web开发技术的中级学者。如果您是以下类型的学习者，此书会带领您迅速进入JSP Web开发领域。

- ① 有一定Java基础，但是没有B/S结构Web开发经验的初学者。
- ② 有其他Web编程语言，如ASP、ASP.NET开发经验，欲快速转向Java Web开发的程序员。
- ③ 对JSP有一定了解，但是缺乏Java Web开发经验，并希望了解流行开源框架Hibernate，Struts，Spring和JSP如何结合使用的程序员。
- ④ 有一定Java Web开发基础，需要加深对JSP技术核心进一步了解和掌握的程序员。

致 谢

本书由傅进勇、邓少烽、李波编写，在编写过程中，温才燊、李振捷等提供了很大的帮助，在此，对他们表示衷心的感谢。由于时间仓促，加上作者水平有限，书中难免存在一些不足和错误之处，如果读者使用本书时遇到问题，可以到本书的服务网站<http://www.cx-pub.com>的相关论坛提问，我们会及时给您回复。

目录 Content



第一篇 JSP应用开发基础

第1章 为什么要使用Servlet&JSP 01

- 1.1 Web网站 02
 - 1.1.1 Web的运作机制 03
 - 1.1.2 客户端和Web服务器 03
- 1.2 HTML和HTTP 05
 - 1.2.1 HTML标记语言 05
 - 1.2.2 脚本语言JavaScript 07
 - 1.2.3 CSS样式表 10
 - 1.2.4 什么是HTTP协议 11
 - 1.2.5 何谓URL 12
 - 1.2.6 HTML表单和Web请求 15
 - 1.2.7 GET和POST 16
- 1.3 动态页面 17
 - 1.3.1 静态页面和动态页面 17
 - 1.3.2 Web容器的功能 19
 - 1.3.3 Servlet的概念及其运作 20
 - 1.3.4 JSP的概念及其运作 21
- 1.4 小结 25

第2章 搭建运行和开发环境 26

- 2.1 Servlet和JSP开发环境概述 27
- 2.2 安装和配置JDK 28
 - 2.2.1 JDK的安装 28
 - 2.2.2 JDK的配置 29
- 2.3 安装和配置Tomcat 31
 - 2.3.1 安装Tomcat服务器 32
 - 2.3.2 测试安装是否成功 33
- 2.4 开发第一个应用程序 35
 - 2.4.1 创建Form提交表单 35
 - 2.4.2 创建表单处理Servlet 36
 - 2.4.3 创建JSP显示页面 37
 - 2.4.4 部署第一个Web应用 37
- 2.5 小结 42

第二篇 Servlet核心技术

第3章 Servlet基础 43

- 3.1 Servlet介绍 44

- 3.1.1 什么是Servlet 44
- 3.1.2 Web容器如何找到Servlet 45
- 3.1.3 Servlet生命周期 47
- 3.2 Servlet的基本结构 48
 - 3.2.1 init方法 48
 - 3.2.2 service方法 50
 - 3.2.3 doGet、doPost、doXXX方法 51
 - 3.2.4 destroy方法 53
 - 3.2.5 Servlet结构综合 54
- 3.3 Servlet相关的主要对象 56
- 3.4 Servlet的基本配置 59
 - 3.4.1 Servlet的名字及路径配置 61
 - 3.4.2 初始化参数 61
 - 3.4.3 启动装入优先级 62
 - 3.4.4 URL到Servlet的映射 65
- 3.5 开发部署一个简单的Servlet 67
- 3.6 小结 72

第4章 请求和响应 73

- 4.1 Servlet的运行模型 74
- 4.2 Servlet请求 75
 - 4.2.1 Servlet请求处理过程 75
 - 4.2.2 Request请求的生命周期 76
 - 4.2.3 获取请求的参数 77
 - 4.2.4 获取请求的信息 80
- 4.3 Servlet响应 82
 - 4.3.1 使用Response返回响应 82
 - 4.3.2 输出文本和HTML 85
 - 4.3.3 了解ContentType属性 88
 - 4.3.4 输出一个非文本文件 89
 - 4.3.5 设置Response Header 91
- 4.4 乱码问题简介 92
 - 4.4.1 理解字符编码 93
 - 4.4.2 为何Java程序容易出现编码问题 98
 - 4.4.3 请求和响应中的编码处理 99
- 4.5 重定向和请求转发 102
 - 4.5.1 使用sendRedirect进行重定向 102
 - 4.5.2 使用RequestDispatcher转发 104
 - 4.5.3 重定向和请求转发的区别 107
- 4.6 小结 107

第5章 数据共享	109
5.1 Servlet范围数据共享	110
5.1.1 Servlet范围共享的方法	110
5.1.2 Servlet共享安全和注意SingleThreadModel的 使用	111
5.2 在RequestDispatcher请求转发间的数据 共享	114
5.3 Web程序范围数据共享	116
5.3.1 ServletContext的概念和作用范围	116
5.3.2 使用ServletContext设置初始参数	117
5.3.3 用ServletContext共享数据	119
5.3.4 使用ServletContextAttributeListener监听器	120
5.3.5 一个简单的ServletContextAttributeListener	122
5.3.6 ServletContext的线程安全	125
5.4 小结	127
第6章 会话跟踪	128
6.1 会话跟踪的概念	129
6.1.1 跨请求会话跟踪需求	129
6.1.2 会话跟踪实现机制和常用技术	130
6.2 Cookie会话跟踪	133
6.2.1 Cookie的概念和用途	133
6.2.2 Servlet Cookie机制	135
6.2.3 Cookie例子	138
6.3 HttpSession会话跟踪	139
6.3.1 Session的工作原理	140
6.3.2 HttpSession接口方法	142
6.3.3 Session的线程安全	143
6.3.4 Session的范围及其生命周期设置	143
6.3.5 例子：记录用户访问次数	145
6.3.6 Session和Cookie技术比较	147
6.3.7 Session、Context、Request作用范围总结	148
6.4 Session属性事件监听	149
6.4.1 通知给专门的对象—— HttpSessionAttributeListener	149
6.4.2 通知给被绑定的对象—— HttpSessionBindingListener	153
6.5 使用Session技术的购物车示例	155
6.6 小结	162

第三篇 JSP技术基础

第7章 JSP基本结构	163
7.1 JSP基础	164
7.1.1 JSP和Servlet的关系	164
7.1.2 JSP的生命周期	165

7.1.3 JSP例子及组成结构	166
7.2 页面指令元素	170
7.2.1 page页面指令	170
7.2.2 include指令	175
7.2.3 taglib指令	177
7.3 脚本元素	178
7.3.1 声明 (Declaration)	178
7.3.2 表达式 (Expression)	178
7.3.3 脚本片断 (Scriptlets)	179
7.4 标准动作元素	181
7.4.1 <jsp:include>	181
7.4.2 <jsp:forward>	183
7.4.3 <jsp:param>	183
7.4.4 <jsp:plugin>、<jsp:params>和<jsp:fallback>	184
7.5 小结	187
第8章 JSP内建对象	188
8.1 内建对象介绍	189
8.1.1 请求对象request	190
8.1.2 响应对象response	191
8.1.3 输出流对象out	192
8.1.4 会话对象session	193
8.1.5 全局对象application	193
8.1.6 页面上下文对象pageContext	194
8.1.7 页面对象page	196
8.1.8 Servlet初始化参数对象config	196
8.1.9 异常对象exception	199
8.2 JSP也可以处理HTML表单	200
8.2.1 创建HTML表单	200
8.2.2 JSP页面处理和输出	202
8.3 小结	205

第9章 在JSP中使用JavaBean

9.1 为什么要使用JavaBean	208
9.2 JavaBean的基本概念	208
9.3 在JSP中使用JavaBean	211
9.3.1 获取JavaBean对象	212
9.3.2 输出属性值	214
9.3.3 设置属性值	215
9.4 小结	223

第10章 JSP和Servlet的集成

10.1 JSP开发的两种模式	225
10.1.1 模式一：JSP + JavaBean	225
10.1.2 模式二：JSP + Servlet + JavaBean	226
10.1.3 两种模式的比较	228

10.1.4 JSP和Servlet的选择	229	13.1 连接池和数据源	301
10.2 使用MVC编程模式	229	13.1.1 连接池的需求和实现原理	301
10.2.1 MVC模式原理及应用	229	13.1.2 JNDI和数据源简介	302
10.2.2 使用JSP和Servlet实现MVC模式	231	13.1.3 配置连接池为数据源	303
10.3 MVC模式实例	232	13.1.4 使用数据源	305
10.4 小结	240	13.2 存取二进制文件	306
第11章 表达式语言	241	13.2.1 二进制文件的存取过程	306
11.1 表达式语言概述	242	13.2.2 在数据库中存取图片	308
11.1.1 表达式语言简介	242	13.3 实现分页显示	311
11.1.2 启用或停止表达式语言功能	243	13.3.1 实现分页显示	311
11.2 基本语法	246	13.3.2 多种分页实现技术的优劣比较	315
11.2.1 表达式语言的组成	246	13.3.3 分页显示的JavaBean实现	317
11.2.2 “.”和“[]”操作符	248	13.4 小结	321
11.2.3 算术操作符	253	第14章 Hibernate数据持久化	322
11.2.4 关系操作符	254	14.1 认识Hibernate	323
11.2.5 条件操作符	256	14.1.1 Hibernate和JDBC的比较	323
11.2.6 逻辑操作符	257	14.1.2 Hibernate体系结构和数据库处理	324
11.2.7 empty操作符	258	14.2 体验Hibernate实例	332
11.2.8 ()操作符及操作符优先级	258	14.2.1 Hibernate安装	332
11.3 隐含对象	259	14.2.2 建立数据库表	335
11.4 保留字	261	14.2.3 建立Java工程	335
11.5 自定义函数	261	14.2.4 创建Hibernate配置文件	336
11.6 小结	264	14.2.5 创建持久化类	337
第四篇 数据库访问技术		14.2.6 创建对象-关系映射文件	338
第12章 JDBC数据库访问技术	265	14.2.7 使用Hibernate操作数据库	340
12.1 JDBC和数据库驱动器程序	266	14.3 小结	345
12.1.1 JDBC概述	266	第五篇 标签语言和表达式语言	
12.1.2 数据库驱动器程序	267	第15章 标准标签库JSTL	346
12.2 JDBC处理数据库方法	269	15.1 JSTL介绍	347
12.2.1 加载并注册数据库驱动	270	15.1.1 标签库与JSTL	347
12.2.2 建立到数据库的连接	271	15.1.2 JSTL标签的种类	347
12.2.3 Statement和PreparedStatement	274	15.1.3 安装JSTL标签库	348
12.2.4 ResultSet结果集	277	15.2 一般用途的标签	352
12.2.5 事务处理	282	15.2.1 <c:out>	352
12.3 连接不同的数据库	284	15.2.2 <c:set>	354
12.3.1 连接MySQL数据库	284	15.2.3 <c:remove>	356
12.3.2 MySQL中文乱码处理	285	15.2.4 <c:catch>	357
12.3.3 连接MSSQL数据库	286	15.3 条件标签	358
12.3.4 连接Oracle数据库	287	15.3.1 <c:if>	358
12.4 使用JDBC操作数据库示例	288	15.3.2 <c:choose>、<c:when>和<c:otherwise>	359
12.5 小结	299	15.4 迭代标签	361
第13章 高级数据库应用	300	15.4.1 <c:forEach>	361
		15.4.2 <c:forEachTokens>	365

15.5 URL相关的标签	366
15.5.1 <c:import>	366
15.5.2 <c:redirect>	369
15.5.3 <c:url>	370
15.5.4 <c:param>	372
15.6 函数标签	372
15.6.1 fn:startsWith	373
15.6.2 fn:endsWith	373
15.6.3 fn:indexOf	374
15.6.4 fn:replace	375
15.6.5 fn:substring	375
15.6.6 fn:split	376
15.7 其他种类的标签	376
15.7.1 本地化	376
15.7.2 访问XML的内容	377
15.7.3 访问数据库	379
15.8 小结	380
第16章 JSP自定义标签扩展	381
16.1 标签类型及结构	382
16.2 自定义标签接口	384
16.2.1 Tag接口	384
16.2.2 IterationTag接口	387
16.2.3 BodyTag接口	388
16.2.4 SimpleTag接口	390
16.3 标签开发实例	392
16.3.1 实现Tag接口	392
16.3.2 从TagSupport继承	394
16.3.3 带属性的标签	395
16.3.4 可处理标签体的标签	396
16.4 标签库配置	399
16.4.1 创建标签库描述文件	399
16.4.2 Web中使用标签	401
16.5 小结	403

第六篇 Web应用高级专题

第17章 使用Servlet过滤器	404
17.1 Servlet过滤器	405
17.1.1 Servlet过滤器简介	405
17.1.2 Servlet过滤器配置	406
17.1.3 实现一个Servlet过滤器	408
17.1.4 部署Servlet过滤器	412
17.2 过滤器的应用场合	413
17.2.1 使用过滤器集中处理请求	

和进行访问控制	413
17.2.2 使用过滤器记录用户访问日志	421
17.2.3 使用过滤器解决中文编码问题	424
17.3 小结	429
第18章 JSP异常处理	430
18.1 Java语言的异常处理机制	431
18.1.1 Java异常处理概述	431
18.1.2 使用Java异常处理	432
18.2 JSP异常处理机制	441
18.2.1 Java Web容器处理异常的机制	441
18.2.2 Servlet类处理异常	443
18.3 JSP应用中处理异常的方式	443
18.3.1 以配置方式处理异常	443
18.3.2 以编程方式处理异常	447
18.4 小结	450
第19章 JSP与日志记录	451
19.1 JSP中记录日志方式	452
19.1.1 通过Servlet容器记录日志	452
19.1.2 使用标记库记录日志	453
19.2 使用Log4j日志器	455
19.2.1 Log4j日志器概述	455
19.2.2 配置Log4j的一般步骤	457
19.2.3 向root日志记录器添加日志存储器	460
19.2.4 在日志记录器的存储器中使用布局	462
19.3 小结	464
第20章 Web设计优化	465
20.1 Web层设计	466
20.1.1 Web层设计的难点	466
20.1.2 清晰的层次划分	468
20.1.3 保持JSP页面代码纯净	469
20.1.4 使用MVC模式	471
20.2 应用DAO模式访问数据库实例	475
20.2.1 创建数据库	477
20.2.2 创建数据传递类	477
20.2.3 建立DAO接口	478
20.2.4 实现DAO接口	479
20.2.5 实现DAO工厂类	480
20.2.6 编写测试JSP页面验证DAO模式部署	481
20.3 小结	483
第21章 Web认证与安全	484
21.1 概述	485
21.1.1 为什么要认证	485

21.1.2 授权	486	23.8 登录管理页面	554
21.1.3 数据完整性和机密性	486	23.8.1 表示层实现	555
21.2 Web认证与信息加密	488	23.8.2 业务逻辑实现	557
21.2.1 为Tomcat建立SSL	488	23.8.3 实现数据访问逻辑	559
21.2.2 HTTP基本认证方式	491	23.9 在线购物页面	561
21.2.3 基于表单的认证方式	495	23.9.1 表示层实现	561
21.3 JSP授权方式	497	23.9.2 业务逻辑实现	564
21.3.1 使用容器授权	497	23.9.3 实现数据访问逻辑	568
21.3.2 基于Servlet过滤器的授权	497	23.10 购物篮管理页面	571
21.4 使用Java认证和授权API	500	23.10.1 表示层实现	572
21.4.1 使用JAAS创建LoginModule	502	23.10.2 业务逻辑实现	573
21.4.2 创建JAAS配置文件	505	23.10.3 实现数据访问逻辑	577
21.4.3 在Servlet中使用JAAS	507	23.11 商品查询页面	579
21.4.4 在JSP中使用JAAS	509	23.11.1 表示层实现	580
21.5 小结	510	23.11.2 业务逻辑实现	582
第22章 Web应用开发及部署	511	23.11.3 数据访问层实现设计	583
22.1 Web应用常见开发目录结构	512	23.12 结账与订单管理页面	584
22.2 Eclipse中开发JSP	512	23.12.1 表示层实现	584
22.2.1 安装并配置Eclipse	513	23.12.2 业务逻辑实现	587
22.2.2 创建JSP页面	518	23.12.3 数据访问层实现设计	591
22.2.3 创建Servlet	521	23.13 小结	592
22.2.4 配置web.xml	522		
22.3 使用Eclipse工具进行调试	522		
22.3.1 设置断点	523		
22.3.2 检查变量和设置观察器	525		
22.3.3 单步跟踪代码	526		
22.4 Web部署	527		
22.5 小结	528		
第七篇 Web应用开发实例			
第23章 综合实例开发	529		
23.1 需求分析	530		
23.1.1 功能需求分析	530		
23.1.2 流程概述	532		
23.1.3 数据库设计	534		
23.2 模块设计	537		
23.3 对象模型设计	540		
23.4 日志管理	541		
23.4.1 Log4j日志器配置管理	541		
23.4.2 表示层和控制层日志管理	543		
23.5 国际化与本地化实现	545		
23.6 性能考虑	549		
23.7 Hibernate配置	552		

第一章

为什么要使用Servlet&JSP

拉开崭新的学习帷幕

JSP是J2EE平台体系中最重要技术，是Java Web开发的核心。本章将阐述Servlet和JSP在网站开发中的作用，以及与其他动态页面技术的比较和区别。

通过本章的学习，读者将对Web有个总体认识，明白Web的运作过程和所用的技术，对Web容器、Servlet和JSP有个初步认识，懂得Web容器在开发Web程序中的地位和作用，知道Servlet、JSP的概念，以及它们和Web容器是如何协作实现Web服务的。

本章内容包括：

- ★ Web的概念和Web的运作机制。
- ★ Web应用的两个重要技术——HTTP和HTML。
- ★ 静态页面和动态页面。
- ★ Web容器的概念。
- ★ 初步认识Servlet和JSP。



1.1 Web网站

这里的Web是World Wide Web (缩写为WWW) 的简称, 也称为万维网。简单来说, Web是建立在Internet (国际互联网, 也称因特网) 之上的一种应用, 它用URI (Uniform Resource Identifier, 统一资源标识符) 来标识分布在世界各地机器上的资源, 如文本、图像等, 客户可以根据URI来访问这些机器上的资源, 以实现全球网络资源的共享。



注意:

人们经常混淆Internet (国际互联网) 和World Wide Web (万维网) 的概念。实际上, Internet指的是用TCP/IP协议联系起来的一个全球性互联网络, 而Web是建立在Internet之上的应用之一, 使用HTTP协议进行通信, 如图1-1所示。其他的Internet应用还有电子邮件服务、FTP服务等。

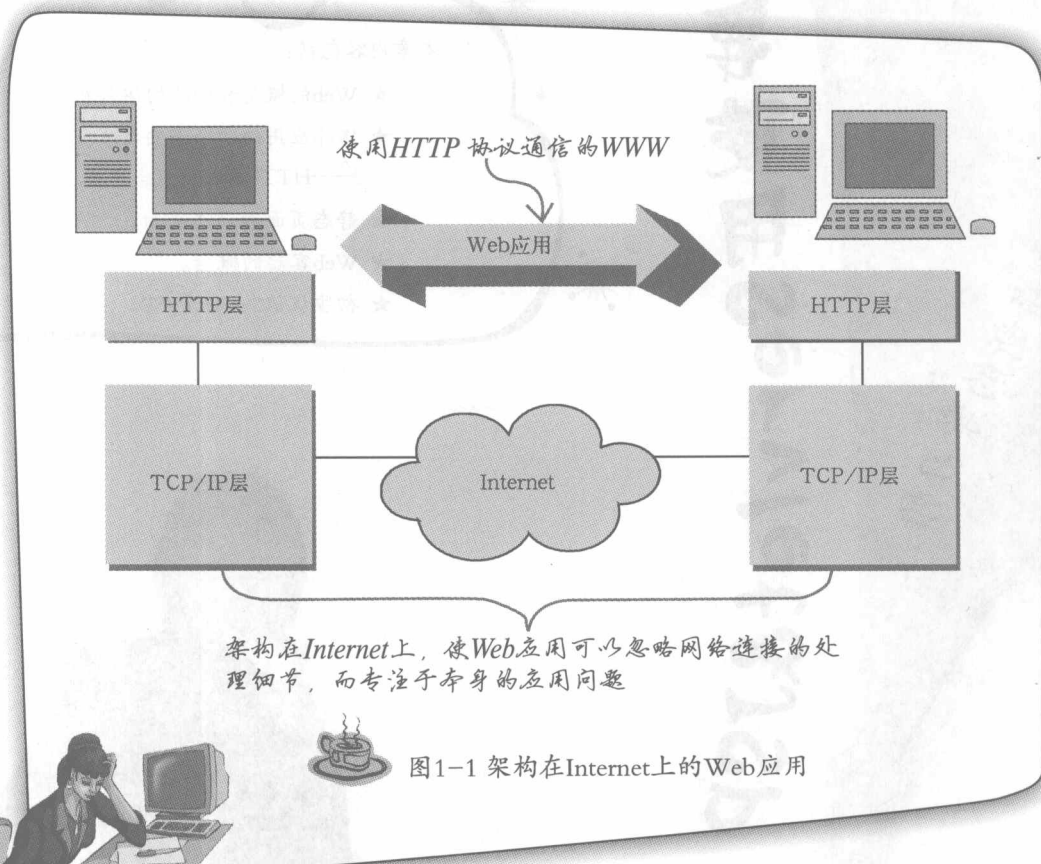


图1-1 架构在Internet上的Web应用

从技术上看, Web利用HTTP协议, 采用客户端/服务器 (Client/Server, 简称C/S) 的模式进行交互。提供Web服务的服务器一般称为Web网站, 或者简称网站; 而客户端最常见的就是浏览器, 如Internet Explorer、Firefox 等。

1.1.1 Web的运作机制

Web是以客户端/服务器的模式，通过HTTP协议来交互的。具体交互方式如下：

★ 一次通信过程包括请求和响应两个步骤。请求由客户端发起，请求信息中带有所需资源的URI和其他所需的信息。

★ 服务器收到请求后，根据URI和其他信息来决定如何处理，如可能找到资源文件并直接返回，也可能交给某个服务程序处理后再返回信息。

★ 如果成功，服务器返回响应信息和资源内容；如果出现问题，如资源不存在、客户无权限、服务器处理过程出现错误等，服务器则返回错误消息通知客户。

★ 不管获取资源是否成功，经过上述的请求/响应的步骤后，这次HTTP通信都会结束。后续的通信将发起新一轮的请求/响应，与本次无关。

运作过程如图1-2所示。

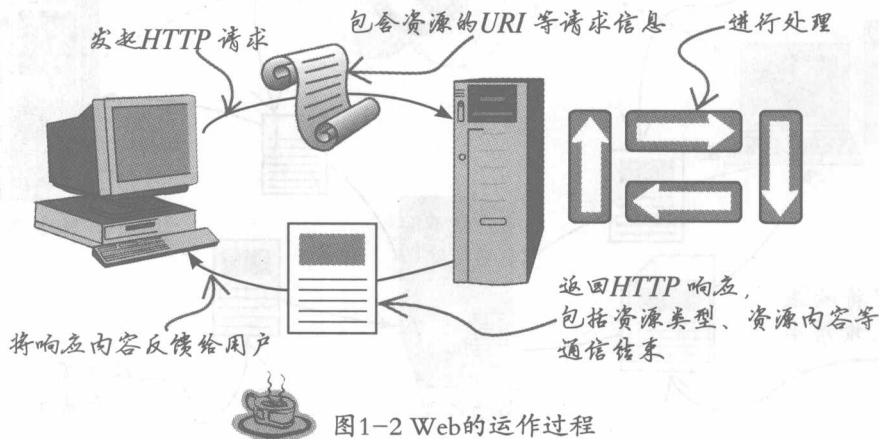


图1-2 Web的运作过程

资源的种类多种多样，因此服务器在返回资源内容时，会在响应信息中附带资源类型的说明，如说明这是图像、HTML页面还是一般的二进制文件等，客户程序会根据资源类型而采用相应的方式反馈给用户。如当客户程序是浏览器时，如果资源类型是HTML页面，它会将其解释出来并显示给用户；如果是一般的二进制文件，则可能弹出对话框让用户选择保存的地点。而当客户程序是下载工具时，则可能对任何资源类型都将它保存起来。

HTML页面，或者简称页面，是Web中应用得最广泛的一类资源。HTML页面也称为超文本(Hypertext)，顾名思义，它是传统纯文本内容的扩充。本质上它只是一个普通的文本文件，但它用HTML (Hypertext Markup Language, 超文本标记语言) 写成，可以在页面中定义文字、图片等丰富多彩的内容。浏览器接收到页面后，会解释它并将定义的内容显示出来。页面中可以定义一种称为超链接(HyperLink)的内容，它包含其他资源的URI，用户单击它就能使浏览器转向该资源，以实现资源间的链接。

1.1.2 客户端和Web服务器

由前面内容知道，Web通信——或者说HTTP协议，是使用客户端/服务器模式的，这样就有客

户端和服务器的概念。但它们不是指具体的机器，而是指实现客户端或服务器端功能的软件。

客户端软件种类很多，如最常用的浏览器、下载工具等，甚至是用户自己定制的程序。只要它能发送HTTP请求，并且能正确处理HTTP响应信息，就算是客户端。Web服务器也如此，能接受HTTP请求，并能返回HTTP响应的就可称之为服务器。本书中如无特别说明，客户端、服务器指的也是软件概念。

一般情况下，一台机器只充当一个角色。但既然客户端、服务器只是软件概念，一台机器能同时运行多个程序，当然也能同时充当这两个角色。Web代理正是这样的例子，如图1-3所示。

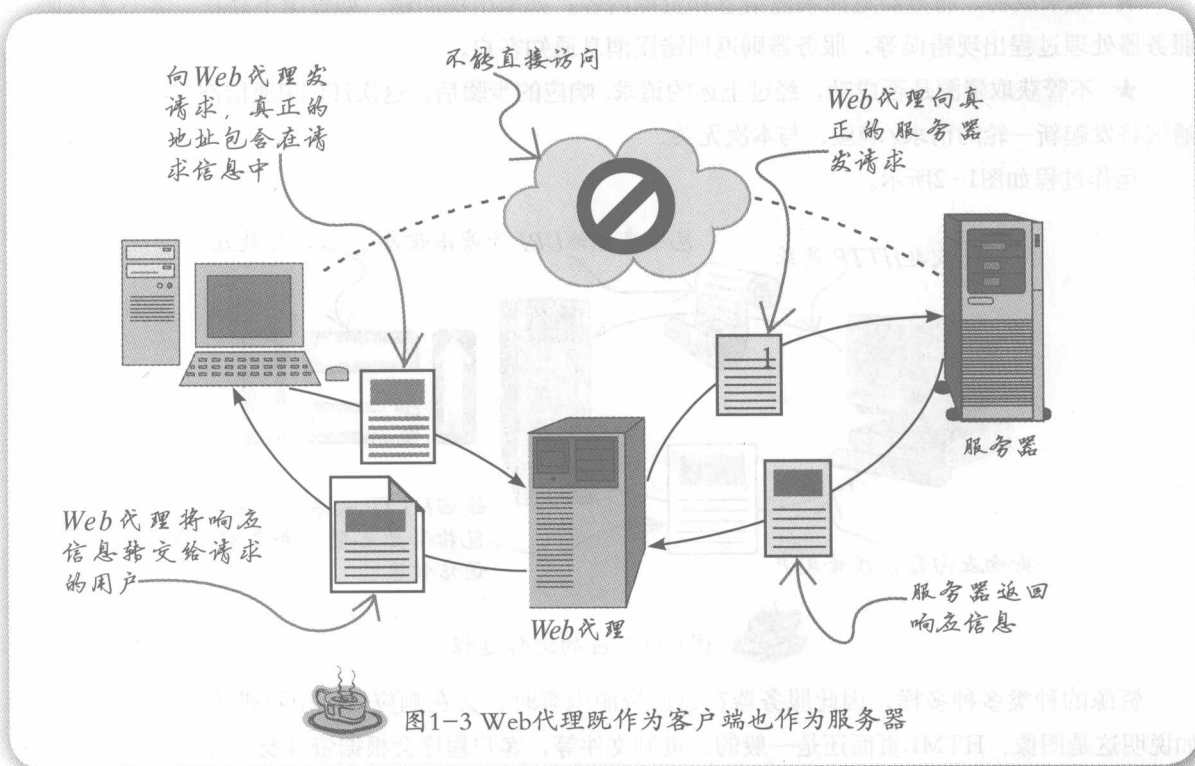


图1-3中，客户程序，如浏览器，需要从某服务器获取资源，但因为某些原因不能直接访问，如这个网站被屏蔽了，于是，它请求Web代理协助。

浏览器发送HTTP请求给Web代理，在请求信息中包含实际要访问的服务器地址。这时，浏览器充当客户角色，而Web代理则充当服务器的角色。

代理服务器可以和真正的服务器通信，于是，它根据请求信息，向真正的Web服务器发出请求，Web服务器接受请求并返回响应信息。这时，代理服务器扮演的是客户角色，而Web服务器则处于服务器的位置。

最后，代理服务器将收到的响应信息返回给浏览器，过程完成。在整个过程中，代理服务器既充当客户端，又充当服务器。

当然，Web代理和浏览器、Web代理和真正的Web服务器这两次通信是互相独立的HTTP通信过程。在浏览器的角度来看，“Web代理和真正的服务器通信”这一步可以抽象成图1-2中的“进行处理”这部分。

注意

为方便讲解，如无特殊说明，本章后面用“地址”代替URI，用“浏览器”代表客户端程序。相信这更符合习惯，不会影响阅读。



1.2 HTML和HTTP

在上一节了解到，页面（或超文本）是一类很重要且应用最广泛的资源，它可以将文字、图像、超链接等丰富的内容呈现给用户。它之所以能实现这么强大的功能，是HTML和浏览器配合的结果。页面用HTML编写，可以定义各种格式的文字、图像、超链接等内容，而浏览器解释这些内容并将它显示给用户。

此外，Web是使用HTTP协议（Hypertext Transfer Protocol，超文本传输协议）进行通信的。HTTP这个名字可能让人认为它只能传输超文本类型的资源。实际上这个命名是因为该协议诞生时，Web主要用于页面传输；但实际上它也能传输其他如图像、文件等非页面数据。读者如有很丰富的网络使用经验，对这一点也会深有体会。

本节简单介绍HTML和HTTP，目的是给读者一个大概的认识，以方便后面的学习。读者如果有深入学习的需要，可以参考其他更详细的书籍和资料。

1.2.1 HTML标记语言

HTML是一种利用标签，用嵌套方式来描述内容的语言。“嵌套方式”是什么意思呢？如图1-4所示，它是由如图1-5所示例子使用的标签组成的树形结构。

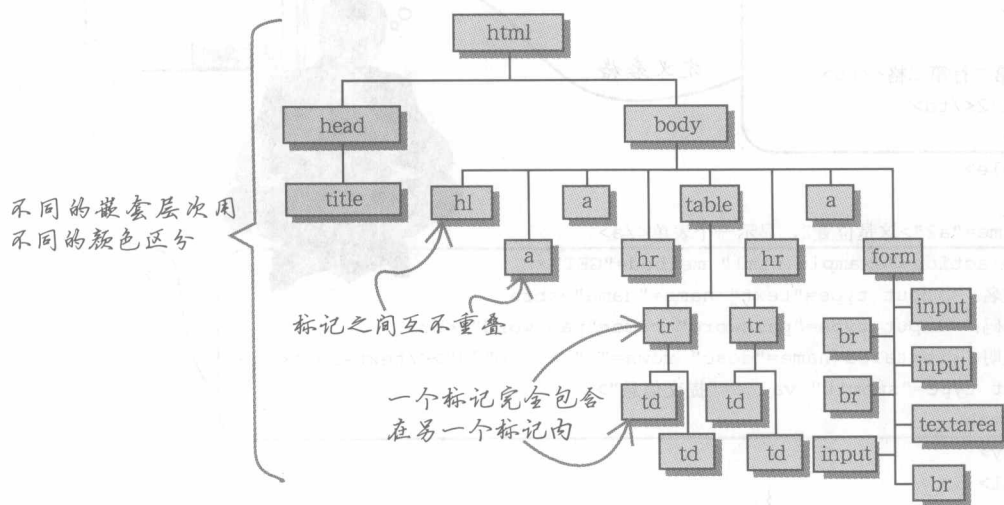


图1-4 图1-5所示例子使用的标签组成的树形结构