



实例源代码·程序软件包·行业规范

JSP

应用与开发技术(第2版)

马建红 李占波 主 编
韩 颖 王瑞娟 姬莉霞 副主编



清华大学出版社

JSP 应用与开发技术

(第2版)

主 编 马建红 李占波
副主编 韩 颖 王瑞娟



清华大学出版社
北 京

内 容 简 介

JSP(Java Server Pages)是由 Sun Microsystems 公司(已被甲骨文公司收购)倡导的、许多公司参与建立的一种动态网页技术标准。JSP 被赋予了 Java 技术的强大功能,能够为用户提供功能强大的技术支持,同时,JSP 继承了 Java 的优势,可以建立安全的、跨平台的动态网站。

本书详细讲解了 JSP 的基本语法和 Web 程序设计方法。全书共 16 章,可分为 7 个部分。第 1~3 章是第 1 部分,介绍了 JSP 应用技术的前导知识和开发环境的搭建;第 4~6 章是第 2 部分,详细讲解了 JSP 技术的基本语法;第 7 章和第 8 章是第 3 部分,是 JSP 应用开发的进阶,讲述了 JavaBean、表单处理以及文件的操作;第 9~11 章是第 4 部分,以 MySQL 数据库为例详细讲解了 JSP 中使用数据库的操作;第 12 章和第 13 章是第 5 部分,详细讲述了 Servlet 技术;第 14 章和第 15 章是第 6 部分,介绍了表达式语言(EL)、标准标签库(JSTL)及自定义标签库;第 16 章是第 7 部分,通过实例讲述了 Web 开发的实际应用。另外,本书还给出了 6 个实验,以指导读者进行上机操作。

本书所附光盘中收录了相关实例运行的开源程序、实例源代码,读者可参照光盘说明进行调试运行。此外,读者还可以通过 www.tupwk.com.cn 下载本书的电子课件。

本书语言简洁,内容丰富,既可作为 JSP 初学者的入门教材,也可作为高等院校相关专业的教材和辅导用书,而且对 JSP 开发人员的自学也具有较高的参考价值。

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

JSP应用与开发技术/马建红,李占波 主编. —2版. —北京:清华大学出版社,2014

ISBN 978-7-302-35687-5

I. ①J… II. ①马… ②李… III. ①JAVA语言—主页制作工具 IV. ①TP312 ②TP393.092

中国版本图书馆 CIP 数据核字(2014)第 180832 号

责任编辑:王 定

封面设计:杜丽雅

版式设计:孔祥峰

责任校对:成凤进

责任印制:刘海龙

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 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:清华大学印刷厂

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:37.75 字 数:942 千字
(附光盘 1 张)

版 次:2011 年 2 月第 1 版 2014 年 9 月第 2 版 印 次:2014 年 9 月第 1 次印刷

印 数:1~4000

定 价:59.80 元

产品编号:053928-01

前言

P R E F A C E

JSP(Java Server Pages)是目前动态网站开发技术中最典型的一种,它继承了 Java 语言的优势,是一种与平台无关的开发技术,而 Java 技术也赋予了 JSP 为用户提供强大功能的技术支持。JSP 实现了动态页面与静态页面的分离,脱离了硬件平台的束缚,提高了执行效率而成为因特网上的主流开发技术,越来越受到编程者的关注和喜爱。

JSP 虽然综合性地包括了 Java 和 HTML 这两类语法,但不能通过简单地使用 JSP,让它集显示、业务逻辑和流程控制于一身,因为用这种方式开发出来的 Web 应用程序是非常难以维护的。所以对 JSP 使用观念的建立,以及 JavaBean、数据库、Servlet 等技术的了解运用是利用 JSP 开发复杂的商业级网站的重点。为了让读者在学习的过程中能够彻底掌握相关概念,除了对基本语法的介绍外,本书同时也将重点集中在面向对象的观点和 JSP 程序架构方面。

《JSP 应用与开发技术(第 2 版)》仍保持了第 1 版实用、新颖和经验总结的特点,从基本的语法和规范入手,以经验为后盾,以实用为目标,以实例为导向,以实践为指导,深入浅出地讲解了 JSP 开发中的种种问题,以精简的内容介绍了 JSP 的语法和 Servlet、JDBC、MySQL、EL、标签库、MVC 等技术。本书每一章节的实例读者都可以直接使用,实例讲解过程条理清晰、循序渐进,符合程序设计的自然思路,读者学完一个章节,也就相应地掌握了相关的 JSP 开发思想和技术,并且通过几个较完整的综合实例,让读者对 JSP+Servlet+JavaBean+JDBC 这样的 Web 架构有一个整体认识。和第 1 版相比,本书不仅采用了最新版本的开发工具,还添加了 JSP 和 MySQL 最新版本中出现的新特性,调整了相关实例,补充了大量的习题,进一步充实了上机实验部分,尽可能地展现出本书的实用性。另外,原有的章节全部进行了细致的修订,部分内容还进行了重大改动。

本书共 16 章,可以分成 7 个部分。

第 1 部分介绍了 JSP 编程基础与环境配置,包括 3 章:第 1 章, JSP 概述;第 2 章, JSP 动态网页设计基础,和第 1 版相比使用了 HTML 最新版本 HTML5,新增了 CSS 知识;第 3 章, JSP 的开发和运行环境,和第 1 版相比增加了最新版的 Eclipse Java EE 集成开发环境和 MyEclipse 开发环境。

第 2 部分介绍了 JSP 应用开发基础,包括 3 章:第 4 章, JSP 基本语法;第 5 章, JSP 内置对象;第 6 章,使用 Cookie 记录信息。

第 3 部分介绍了 JSP 应用开发进阶,包括 2 章:第 7 章, JavaBean 和表单处理;第 8 章, JSP 中的文件操作。这部分主要介绍了 JavaBean 分离表示与实现,使用 JSP 处理 HTML 表单,使用 Java 类库里的 I/O 类,开发具备文件存取功能的网页程序。和第 1 版不同的是,更换了容

易使用的 JspSmartUpload 包实现文件的上传。

第 4 部分介绍了 JSP 数据库编程基础,包括 3 章:第 9 章,数据库操作基础;第 10 章,应用 JDBC 进行数据库开发;第 11 章,JSP 与 JavaBean 应用实例。和第 1 版一样介绍了 MySQL 数据库的无界面基本操作,但不同的是,第 2 版中 MySQL 的版本升级到 mysql-5.1.25-rc-win32,并且添加了和版本相匹配的 MySQL 带界面的基本操作,以 MySQL 数据库为例详细讲解了 JSP 中使用数据库的操作,包括 JDBC、连接池、分页处理等技术;更换了第 1 版第 11 章中简单的留言板实例,改为一个完整的电商系统,还对 Model1 模式的应用给予详细的讲解,项目模块功能更强大。

第 5 部分讲述了 Servlet 技术,包括 2 章:第 12 章,Servlet 基础;第 13 章,使用 Servlet 过滤器和监听器。第 2 版中重点调整了监听器接口,增加了新实例。

第 6 部分讲述了标签库的应用,包括 2 章:第 14 章,表达式语言(EL)和标准标签库(JSTL);第 15 章,自定义标签库。第 2 版中对每部分内容都进行了细致的修订,增加了 EL,更加完善了标签库的知识体系结构。

第 7 部分即第 16 章,Web 应用开发实践。和第 1 版不同的是第 2 版中更新了实例,运用 MVC 技术,采用三层架构,并采用了自定义标签和 EL 等技术展示了一个门户网站综合实例,使读者对这些技术的使用有更深刻的了解。

本书采用由浅入深、循序渐进的方法,全面、系统地介绍了 JSP 程序设计的原理、方法和技术,还提供了大量的 JSP 应用开发实例,给出了相应的实用技巧、操作步骤及优化思路,使读者可以很快地进行实际开发。每章的最后还提供了习题,让读者能够检验自己对各章内容的学习、消化程度,并巩固所学的知识。

本书由马建红、李占波任主编,韩颖、王瑞娟、姬莉霞任副主编,参与编写的人员还有张晗、卫权岗、宰光军、何保锋、尹辉、程凤娟、秦兴桥、赵玉娟、王永皎等。此外,林楠、李翠霞、史苇航在整理材料方面给予了编者很大的帮助,在此对他们表示衷心的感谢。

由于时间仓促,加之水平有限,书中不足之处在所难免,敬请读者批评指正。

编者



CONTENTS

第1章 JSP 概述	1
1.1 软件编程体系简介	2
1.1.1 C/S 结构编程体系	2
1.1.2 B/S 结构编程体系	3
1.2 企业应用开发架构	3
1.2.1 两层架构	3
1.2.2 三层架构	3
1.2.3 N 层架构	4
1.2.4 开发架构比较	4
1.3 JSP 概述	4
1.3.1 什么是 JSP	5
1.3.2 JSP 技术原理	6
1.3.3 JSP 和其他动态网站 开发技术	7
1.3.4 J2EE 简介	10
1.4 JSP 知识体系及学习之路	13
1.4.1 JSP 知识体系	13
1.4.2 JSP 程序员学习路径	13
1.5 小结	15
1.6 习题	15
1.6.1 选择题	15
1.6.2 判断题	16
1.6.3 填空题	16
1.6.4 简答题	16
1.6.5 拓展实践题	16
第2章 JSP 动态网页设计基础	17
2.1 HTML 技术	18
2.1.1 HTML5 基本结构	18
2.1.2 HTML 常用标签	20

2.1.3 表单	28
2.1.4 XML 与 XHTML	36
2.2 CSS 技术	38
2.2.1 CSS 基本语法	38
2.2.2 在 HTML 文档中使用 CSS 的方法	40
2.2.3 常用 CSS 属性	42
2.3 JavaScript 技术	50
2.3.1 JavaScript 语法	50
2.3.2 JavaScript 使用方式	51
2.3.3 JavaScript 代码实例	52
2.4 小结	57
2.5 习题	57
2.5.1 选择题	57
2.5.2 判断题	58
2.5.3 填空题	58
2.5.4 简答题	58
第3章 JSP 的开发和运行环境	59
3.1 JSP 的开发和应用平台介绍	60
3.1.1 Caucho 公司的 Resin 平台	60
3.1.2 Apache 公司的 Tomcat 平台	60
3.1.3 BEA 公司的 WebLogic 平台	61
3.1.4 IBM WebSphere Application Server 平台	61
3.2 Eclipse Java EE 集成开发环境	62
3.2.1 安装和配置 JDK	62
3.2.2 Tomcat 服务器	64
3.2.3 Eclipse Java EE 开发环境搭建	68
3.3 Eclipse 集成开发环境配置	73

3.4 MyEclipse 开发环境	75	4.6.5 编程题	125
3.4.1 MyEclipse 简介与下载	76	第 5 章 JSP 内置对象	127
3.4.2 MyEclipse 安装与使用	76	5.1 JSP 内置对象概述	128
3.5 小结	82	5.2 request 对象	129
3.6 习题	82	5.2.1 request 对象常用方法	129
3.6.1 选择题	82	5.2.2 request 对象应用实例	130
3.6.2 判断题	83	5.3 response 对象	137
3.6.3 填空题	83	5.3.1 response 对象常用方法	137
3.6.4 简答题	83	5.3.2 response 对象应用实例	138
第 4 章 JSP 基本语法	85	5.4 out 对象	146
4.1 JSP 文件的结构	86	5.4.1 out 对象方法成员与 数据输出	146
4.1.1 创建第一个 JSP 文件	86	5.4.2 缓冲区操作	147
4.1.2 分析 JSP 文件的组成元素	87	5.4.3 out 对象应用实例	148
4.2 JSP 的脚本元素	88	5.5 session 对象	151
4.2.1 隐藏注释(Hidden Comment)	88	5.5.1 session 的概念	151
4.2.2 HTML 注释	89	5.5.2 session 对象的 ID	152
4.2.3 声明语句(Declaration)	91	5.5.3 session 的有效期限	152
4.2.4 脚本段(Scriptlets)	94	5.5.4 访问 session 中的数据	152
4.2.5 表达式(Expression)	95	5.5.5 其他 session 对象的 常用方法	153
4.3 JSP 指令元素	96	5.5.6 session 对象应用实例	154
4.3.1 page 指令	96	5.6 application 内置对象	160
4.3.2 include 指令	101	5.6.1 存取 application 中的数据	161
4.3.3 taglib 指令	103	5.6.2 使用 application 对象 取得信息	162
4.4 JSP 动作元素	104	5.6.3 application 对象应用实例	162
4.4.1 <jsp:include>	104	5.7 其他 JSP 内置对象	164
4.4.2 <jsp:forward>	107	5.7.1 pageContext 对象	164
4.4.3 <jsp:param>	110	5.7.2 config 对象	166
4.4.4 <jsp:useBean>、<jsp:setProperty> 和<jsp:getProperty>动作	111	5.7.3 page 对象	169
4.4.5 <jsp:plugin>	120	5.7.4 exception 对象	169
4.5 小结	122	5.8 小结	169
4.6 习题	123	5.9 习题	170
4.6.1 选择题	123	5.9.1 选择题	170
4.6.2 判断题	124	5.9.2 判断题	172
4.6.3 填空题	125		
4.6.4 简答题	125		

5.9.3 填空题	172
5.9.4 简答题	172
5.9.5 编程题	172
第6章 使用 Cookie 记录信息	175
6.1 Cookie 的概念和特性	176
6.1.1 什么是 Cookie	176
6.1.2 Cookie 的常见用途	177
6.1.3 对 Cookie 进行适当设置	178
6.2 在 JSP 中使用 Cookie	179
6.2.1 创建 Cookie	179
6.2.2 读写 Cookie	179
6.2.3 Cookie 中的主要方法	180
6.2.4 几个操作 Cookie 的常用方法	181
6.3 Cookie 对象的应用实例	183
6.4 Cookie 的安全问题	192
6.5 小结	193
6.6 习题	194
6.6.1 选择题	194
6.6.2 判断题	195
6.6.3 填空题	196
6.6.4 简答题	196
6.6.5 编程题	196
第7章 JavaBean 和表单处理	197
7.1 非 MVC 模式(Model1)	198
7.1.1 Model1 的特点	198
7.1.2 Model1 的应用范围	199
7.2 MVC 编程模式(Model2)	199
7.2.1 什么是 MVC 模式	199
7.2.2 MVC 模式在 Web 编程中的应用	200
7.3 剖析 JavaBean	201
7.3.1 什么是 JavaBean	202
7.3.2 JavaBean 的特征	203
7.3.3 创建一个 JavaBean	204
7.4 在 JSP 中使用 JavaBean	206

7.4.1 调用 JavaBean	206
7.4.2 访问 JavaBean 属性	206
7.4.3 设置 JavaBean 属性	206
7.4.4 JavaBean 的生命周期	207
7.4.5 类型自动转换规则	211
7.5 使用JavaBean处理表单数据	211
7.5.1 JSP 处理与 form 相关的常用标签简单实例	212
7.5.2 设置中文编码	217
7.5.3 Post 与 Get 的差异	218
7.6 小结	219
7.7 习题	219
7.7.1 选择题	219
7.7.2 判断题	220
7.7.3 填空题	221
7.7.4 简答题	221
7.7.5 编程题	221
第8章 JSP 中的文件操作	223
8.1 数据流和 File 类	224
8.1.1 数据流	224
8.1.2 File 类	224
8.2 读写文本文件	227
8.3 文件的浏览	229
8.4 创建和删除目录	230
8.5 文件的上传和下载	231
8.6 使用jspSmartUpload上传包	235
8.7 小结	238
8.8 习题	238
8.8.1 选择题	238
8.8.2 判断题	239
8.8.3 填空题	239
8.8.4 简答题	239
8.8.5 编程题	239
第9章 数据库操作基础	241
9.1 关系数据库及 SQL	242
9.2 在Windows上安装MySQL	243

9.3 MySQL 的常用操作	247	10.1.1 JDBC 的用途	264
9.3.1 设置环境变量	247	10.1.2 JDBC 的典型用法	264
9.3.2 启动 MySQL 数据库	248	10.1.3 JDBC 体系结构	265
9.3.3 连接 MySQL	248	10.1.4 驱动器类型	265
9.3.4 退出 MySQL	248	10.1.5 安装驱动器	267
9.3.5 增加用户	249	10.2 JDBC 连接数据库的方法	267
9.3.6 删除授权	249	10.3 使用 JDBC 操作数据库	269
9.3.7 备份数据库	250	10.3.1 使用 JDBC 访问	
9.3.8 恢复数据库	250	数据库的过程	269
9.3.9 备份表	250	10.3.2 使用 Statement 执行	
9.3.10 恢复表	251	SQL 语句	272
9.3.11 查看、创建、删除和选择		10.3.3 PreparedStatement 接口	281
数据库命令	251	10.3.4 CallableStatement 对象	285
9.3.12 导入命令	252	10.3.5 使用 ResultSet 处理	
9.4 常用查询的例子	252	结果集	287
9.4.1 查询时间	253	10.4 Java 与 SQL 的数据	
9.4.2 查询当前用户	253	类型转换	292
9.4.3 查询数据库版本	253	10.5 使用 JDBC 连接不同的	
9.4.4 查询当前使用的数据库	253	数据库	294
9.4.5 使用 AUTO_INCREMENT	254	10.5.1 连接 Oracle 数据库	294
9.4.6 列的最大值	254	10.5.2 连接 DB2 数据库	294
9.4.7 拥有某个字段的		10.5.3 连接 SQL Server 数据库	294
组间最大值的行	256	10.5.4 连接 Sybase 数据库	295
9.4.8 使用用户变量	256	10.5.5 连接 Access 数据库	295
9.5 MySQL GUI Tools	256	10.6 连接池	295
9.5.1 MySQL GUI Tools 安装	256	10.6.1 连接池的实现原理	295
9.5.2 MySQL GUI Tools		10.6.2 在 Tomcat 上配置数据源	
基本使用方法	258	与连接池	296
9.6 小结	260	10.6.3 配置连接池时需要注意	
9.7 习题	261	的问题	299
9.7.1 选择题	261	10.7 存取二进制文件	299
9.7.2 填空题	261	10.7.1 图像文件存取到	
9.7.3 编程题	262	数据库的过程	300
第 10 章 应用 JDBC 进行		10.7.2 声音文件存取到	
数据库开发	263	数据库的过程	304
10.1 JDBC 概述	264		

10.7.3	视频文件存取到数据库的过程	309
10.8	实现分页显示	313
10.8.1	分页显示技术的优劣比较	313
10.8.2	分页显示的 JavaBean 实现	314
10.9	小结	322
10.10	习题	322
10.10.1	选择题	322
10.10.2	判断题	323
10.10.3	填空题	323
10.10.4	简答题	324
10.10.5	编程题	324
第 11 章	JSP 与 JavaBean	
	应用实例	325
11.1	需求和设计	326
11.1.1	功能介绍	326
11.1.2	文件结构	327
11.1.3	数据库设计	328
11.2	使用 JavaBean 封装数据库的访问	330
11.3	项目页面实现	335
11.3.1	用户模块设计与实现	335
11.3.2	管理员模块设计与实现	353
11.4	小结	366
第 12 章	Servlet 基础	367
12.1	Servlet 介绍	368
12.1.1	什么是 Servlet	368
12.1.2	Servlet 技术特点	369
12.1.3	JSP 与 Servlet 的关系	369
12.1.4	Servlet 的工作原理	370
12.1.5	Servlet 常用接口和类	371
12.2	开发部署一个简单的 Servlet	372
12.2.1	创建 Servlet 文件	373

12.2.2	Servlet 的配置文件	375
12.3	Servlet 实现相关的接口和类	377
12.3.1	GenericServlet	378
12.3.2	HttpServlet	378
12.3.3	Servlet 实现相关实例	379
12.4	Servlet 请求和响应相关	383
12.4.1	HttpServletRequest 接口	383
12.4.2	HttpServletResponse 接口	385
12.4.3	Servlet 请求和响应相关实例	386
12.5	Servlet 配置相关	388
12.5.1	ServletConfig 接口	389
12.5.2	获取 Servlet 配置信息的例子	389
12.6	Servlet 中的会话追踪	394
12.6.1	HttpSession 接口	394
12.6.2	HttpSession 应用实例	396
12.7	Servlet 上下文	398
12.7.1	ServletContext 接口	398
12.7.2	ServletContext 接口的应用实例	399
12.8	Servlet 协作	401
12.8.1	RequestDispatcher	401
12.8.2	forward() 控制页面跳转	401
12.8.3	include() 控制页面包含	403
12.9	Servlet 异常处理	404
12.9.1	声明式异常处理	404
12.9.2	程序式异常处理	408
12.10	Servlet 应用实例	411
12.11	小结	421
12.12	习题	421
12.12.1	选择题	421
12.12.2	判断题	423
12.12.3	填空题	423
12.12.4	简答题	424

12.12.5 编程题	424
第 13 章 使用 Servlet 过滤器和监听器	425
13.1 过滤器在 Web 开发中的应用	426
13.1.1 过滤器概述	426
13.1.2 Filter API	427
13.1.3 Filter 接口	427
13.1.4 FilterConfig 接口	428
13.1.5 FilterChain 接口	428
13.1.6 编写过滤器类	429
13.1.7 过滤器的部署	430
13.1.8 对请求数据进行处理 的过滤器	434
13.1.9 对响应内容进行压缩的 过滤器	440
13.2 Servlet 监听器	445
13.2.1 监听器接口	445
13.2.2 ServletRequestListener 接口	446
13.2.3 ServletRequestAttributeListener 接口	448
13.2.4 ServletContextListener 接口	448
13.2.5 ServletContextAttributeListener 接口	449
13.2.6 HttpSessionAttributeListener 接口	452
13.2.7 HttpSessionBindingListener 接口	454
13.3 小结	460
13.4 习题	461
13.4.1 选择题	461
13.4.2 判断题	462
13.4.3 填空题	462
13.4.4 简答题	462

13.4.5 编程题	463
第 14 章 JSTL 标准标签库	465
14.1 EL 表达式语言	466
14.1.1 EL 与 EL 隐含对象	466
14.1.2 在 EL 中访问 JSP 隐含 对象的 get×××()方法	471
14.1.3 用 EL 访问 JavaBean 中 的属性	472
14.2 JSTL 标签库简介	473
14.3 设置 JSTL 运行环境	473
14.3.1 JSTL 的安装	474
14.3.2 JSTL 应用示例	475
14.4 使用核心标签	475
14.4.1 表达式操作	476
14.4.2 建立 URL	480
14.4.3 条件控制	484
14.4.4 迭代-运行循环	486
14.5 使用 JSTL 的数据库标签	490
14.5.1 指定数据源	490
14.5.2 进行查询或更新操作	491
14.5.3 对返回的结果进行处理	494
14.5.4 其他 SQL 标签库的标签	495
14.6 i18n 与国际化	496
14.6.1 国际化设置标签	496
14.6.2 消息标签库	497
14.6.3 数字、日期格式化	499
14.7 函数标签	503
14.8 小结	503
14.9 习题	504
14.9.1 选择题	504
14.9.2 判断题	505
14.9.3 填空题	506
14.9.4 简答题	506
14.9.5 编程题	506
第 15 章 自定义标签库	507
15.1 自定义标签体系介绍	508

15.1.1	标签的形式	509	16.3	系统功能结构	539
15.1.2	标签类相关接口和类	509	16.4	系统功能描述	539
15.1.3	标签库描述文件	514	16.4.1	游客用户浏览模块	539
15.1.4	在 Web 部署描述符中 引入标签库文件	515	16.4.2	管理员登录模块	541
15.1.5	在页面中使用标签	516	16.4.3	管理员管理模块	542
15.1.6	标签在 Web 页面中的 作用	517	16.5	数据库设计	545
15.2	传统标签的开发	517	16.5.1	数据库逻辑结构设计	545
15.2.1	带属性标签的开发	517	16.5.2	数据库表的设计	545
15.2.2	带 Body 标签的开发	521	16.5.3	数据库相关脚本	546
15.2.3	嵌套标签的开发	524	16.6	系统实现	548
15.2.4	迭代标签的开发	527	16.6.1	模块公用类	548
15.3	Simple 标签的开发	531	16.6.2	JavaBean	553
15.3.1	SimpleTag 接口	531	16.6.3	Servlet	557
15.3.2	Simple 标签的开发示例	532	16.6.4	自定义标签	561
15.4	小结	533	16.6.5	前台界面的实现	567
15.5	习题	534	16.6.6	后台管理页面的实现	571
15.5.1	选择题	534	16.7	小结	576
15.5.2	判断题	534	16.8	习题	576
15.5.3	填空题	534	附录	实验	577
15.5.4	简答题	534	实验一	JSP 应用开发基础	577
15.5.5	编程题	534	实验二	JSP 应用开发基础	580
第 16 章	Web 应用开发实践	537	实验三	JSP 应用开发进阶	585
16.1	信息发布平台	538	实验四	JSP 数据库编程基础	587
16.2	系统需求分析	538	实验五	Servlet 技术实验	589
			实验六	Web 应用开发	591

第1章

JSP 概述

本章主要对 JSP 技术进行概要介绍,并给出了一些学习 JSP 的建议。为了让读者在开始学习之前能对 JSP 技术有一个清晰与完整的概念,本章首先介绍了软件编程体系、企业应用开发架构,然后通过编写一个简单的 JSP 页面实例让读者对 JSP 技术有一个直观的感性认识。接着通过介绍 JSP 的技术原理以及与其他主流动态网页技术的比较,进一步了解 JSP 技术是一种功能强大、可以实现跨平台操作的动态网页开发技术。本章对 JSP 知识体系的剖析有助于读者学习和掌握 JSP 知识体系中的各个模块,对 JSP 技术有一个总体性的了解。

本章学习目标

- ◎ 了解软件编程体系
- ◎ 了解企业应用开发架构
- ◎ 掌握 JSP 的基本概念
- ◎ 掌握 JSP 的知识体系
- ◎ 了解 JSP 的学习之路

1.1 软件编程体系简介

目前,在应用开发领域中主要分为两大编程体系,一种是基于浏览器的 B/S(Browser/Server)结构,另一种是 C/S(Client/Server)结构。应用程序开发体系如图 1-1 所示。

开发基于 C/S 结构项目,传统的开发环境有 Visual Basic、Visual C++ 以及 Delphi 等,随着 Java 体系以及 .NET 体系的普及,目前更流行 .NET 编程体系和 Java 编程体系。

开发基于 B/S 结构项目,目前主要采用三种服务器端语言: JSP(Java Server Pages)、PHP(Hypertext Preprocessor)和 ASP.NET。这三种语言构成三种常用应用开发组合: JSP+Oracle 体系、PHP+MySQL 体系以及 ASP.NET+SQL Server 体系。

软件开发涉及的语言很多,学习起来也是有规律可循的。图 1-1 最下面的方框将目前常用的开发语言分成两大语系: Basic 语系和 C 语系,语系中的语言所有的流程控制语句都是一样的,常用的函数也大同小异。所以只要精通其中任何一门语言,该语系中的其他语言也就比较容易掌握了。

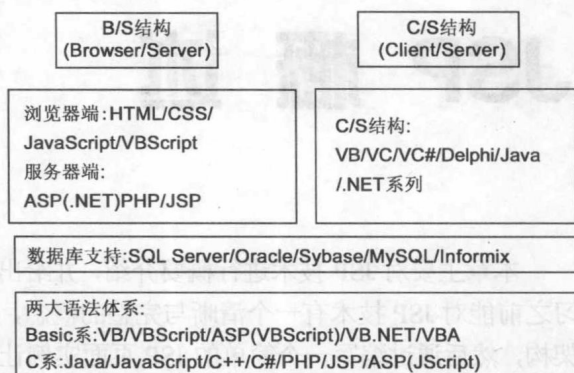


图 1-1 应用程序开发体系

1.1.1 C/S 结构编程体系

2000 年以前, C/S 结构占据开发领域的主流, 随着 B/S 结构的发展, C/S 结构主流地位已经逐步被 B/S 结构取代, 目前在整个开发领域中, C/S 结构的应用大概能占到 40% 的份额。C/S 结构应用程序最大特点是在每个用户端需要安装程序, 所有用户端程序和中心服务器进行信息交互; 这种结构优点是用户端程序一致, 比较方便控制, 服务器端和用户本地的数据很容易进行交互, 通信速度比较快; 缺点是每个用户都需要安装客户端, 比较繁琐, 而且不能很好地跨操作系统平台。

C/S 结构通常适用于具有固定的用户端或者少量的用户端, 并且是对安全性要求比较高的应用。比如银行信息管理系统、邮局信息管理系统和飞机票火车票售票系统等。

传统的 C/S 结构通常使用 PowerBuilder、Delphi、Visual Basic、Visual C++、JBuilder 作为开发环境, 使用 SQL Server、Oracle 或者 DB2 作为数据库支持。随着时间的发展、版本的更新, 主流的 C/S 开发环境开始向 .NET 和 Java 两大主流体系转变, 目前大部分 C/S 结构应用都使用 VB.NET、VC#.NET 以及 Java 开发, 其中, VB.NET 和 VC#.NET 只是描述的语言不一样, 设计思想和开发环境全部一样, 因此只要掌握其中一个, 就可以满足开发要求了。

VB.NET 是从 Visual Basic 发展而来的, Visual Basic 曾经具有开发领域世界第一的程序员数量, 因此非常多的 C/S 应用采用 VB.NET 开发环境。

1.1.2 B/S 结构编程体系

B/S 结构编程语言分成浏览器端编程语言和服务端编程语言。

浏览器端包括 HTML (HyperText Markup Language, 超文本标记语言)、CSS(Cascading Style Sheets, 层叠样式表单)、JavaScript 语言和 VBScript 语言。所谓浏览器端编程语言就是这些语言都是被浏览器解释执行的。HTML 和 CSS 是由浏览器解释的, JavaScript 语言和 VBScript 语言也是在浏览器上执行的。

为了实现一些复杂的操作, 比如连接数据库、操作文件等, 需要使用服务器端编程语言。目前主要是 3P(ASP.NET、JSP 和 PHP)技术。ASP.NET 是微软公司推出的, 在这三种语言中是用得最为广泛的一种。JSP 是 SUN 公司(已被甲骨文公司收购)推出的 J2EE(Java 2 Enterprise Edition, Java2 企业版)核心技术中重要的一种。PHP 在 1999 年的下半年和 2000 年应用得非常广泛, 因为 Linux+PHP+MySQL(一种中小型数据库管理系统)构成全免费的而且非常稳定的应用平台, 这三种语言是目前应用开发体系的主流。

数据库支持是必须的, 目前应用领域的数据库系统全部采用关系型数据库系统(Relation Database Management System, RDBMS)。在企业级的开发领域中, 主要采用三大厂商的关系数据库系统: 微软公司的 SQL Server、Oracle 公司的 Oracle 和 IBM 公司的 DB2。

1.2 企业应用开发架构

在构建企业级应用时, 通常需要大量的代码, 而且这些代码一般分布在不同的计算机上, 划分代码运行在不同计算机上的理论就是多层设计理论。

企业级应用系统通常分为两层、三层和 N 层架构。

1.2.1 两层架构

传统的两层应用包括用户接口和后台程序, 后台程序通常是一个数据库, 用户接口直接同数据库进行对话。实现上, 通常使用 JSP、ASP 或者 VB 等技术编写这类软件, 结构如图 1-2 所示。

两层应用架构显示逻辑层一般由 HTML、JSP、ASP 实现, 通过 JSP 和 ASP 直接和数据库相连。

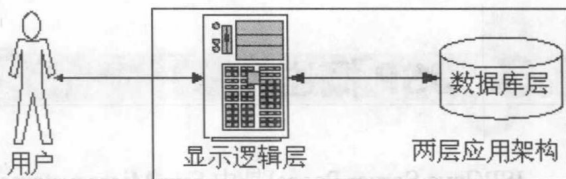


图 1-2 两层应用架构

1.2.2 三层架构

在两层应用中, 应用程序直接同数据库进行对话。三层结构在用户接口代码和数据库中间加入了一个附加的逻辑层, 通常这个层称为“商务逻辑层”, 如图 1-3 所示。

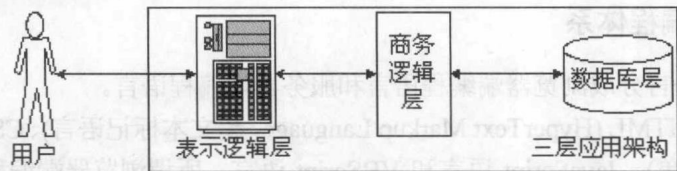


图 1-3 三层应用架构

1.2.3 N 层架构

如果某个应用超过 3 个独立的代码层,就不再称四层或者五层等名称,而是统称为 N 层,那么这个应用称为 N 层应用,如图 1-4 所示。

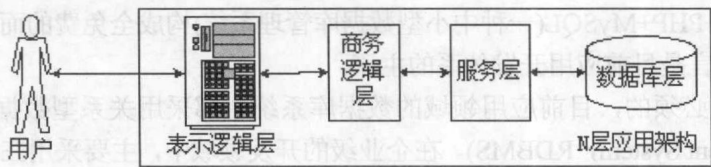


图 1-4 N 层应用架构

1.2.4 开发架构比较

两层架构的优点是开发过程比较简单,利用服务器端的程序直接访问数据库,部署起来比较方便;缺点是程序代码维护起来比较困难,程序执行的效率比较低,用户容量比较少。

三层架构基本解决了两层架构的缺点,将显示部分和逻辑流程控制分开,利用服务器应用程序实现显示部分,利用商务逻辑层实现程序的流程控制,分层使维护变得方便一些,而且执行效率也会有所提高,但是相对部署起来就比较困难一些。

根据实际需要,会进一步细化每一层,或者添加一些层,就形成了 N 层架构。和三层架构一样,组件化的设计使维护相对容易,但是部署相对困难。

1.3 JSP 概述

JSP(Java Server Pages)是由 Sun Microsystems 公司(现已被甲骨文公司收购)倡导的、许多公司参与一起建立的一种动态网页技术标准。JSP 技术是用 Java 语言作为脚本语言, JSP 网页为整个服务器端的 Java 库单元提供了一个接口来服务于 HTTP 的应用程序。

在传统的网页 HTML 文件(*.htm, *.html)中加入 Java 程序片段(Scriptlet)和 JSP 标记(Tag),就构成了 JSP 网页(*.jsp)。Web 服务器在遇到访问 JSP 网页的请求时,首先执行其中的程序片段,然后将执行结果以 HTML 格式返回给客户。程序片段可以操作数据库、重新定向网页以及发送 E-mail 等,这就是建立动态网站所需要的功能。所有程序操作都在服务器端执行,网络上传送给客户端的仅是得到的结果,对客户浏览器的要求最低,可以实现无 Plugin、无 ActiveX、无 JavaApplet,甚至无 Frame。

1.3.1 什么是 JSP

JSP 是基于 Java 的技术,用于创建可支持跨平台及 Web 服务器的动态网页。从构成情况上来看,JSP 页面代码一般由普通的 HTML 语句和特殊的基于 Java 语言的嵌入标记组成,所以它具有了 Web 和 Java 功能的双重特性。

JSP 1.0 规范是 1999 年 9 月推出的,12 月又推出了 1.1 规范。此后 JSP 又经历了几个版本,最新版本是 2003 年发布的 JSP 2.0。本书介绍的技术都基于 JSP 2.0 规范。

为了让读者对 JSP 技术有一个直观的认识,先来看一个非常简单的 JSP 页面及其运行效果。以下是 helloWorld.jsp 的源代码。

```
<%@ page language="java" contentType="text/html; charset=gbk"%>
<html>
<head>
<title>Hello World!</title>
</head>
<body bgcolor="#FFFFFF">
<h3>
<%
    out.println("JSP Hello World!");
%>
</h3>
</body>
</html>
```

JSP 是一种动态网页技术标准。可以将网页中的动态部分和静态的 HTML 相分离。用户可以使用平常得心应手的工具并按照平常的方式来书写 HTML 语句。然后,将动态部分用特殊的标记嵌入即可,这些标记常常以“<%”开始并以“%>”结束。

程序运行效果如图 1-5 所示。

同 HTML 以及 ASP 等语言相比,JSP 虽然在表现形式上同它们的差别并不大,但是它却提供了一种更为简便、有效的动态网页编写手段,而且,由于 JSP 程序同 Java 语言有着天然的联系,所以在众多基于 Web 的架构中,都可以看到 JSP 程序。

由于 JSP 程序增强了 Web 页面程序的独立性、兼容性和可重用性,所以,与传统的 ASP、PHP 网络编程语言相比,它具有以下特点:

- JSP 的执行效率比较高。由于每个基于 JSP 的页面都被 Java 虚拟机事先解析成一个 Servlet,服务器通过网络接收到来自客户端 HTTP 的请求后,Java 虚拟机解析产生的 Servlet 将开启一个“线程(Thread)”来提供服务,并在服务处理结束后自动销毁这个线程,如图 1-6 所示,这样的处理方式将大大提高系统的利用率,并能有效地降低系统的负载。
- 编写简单。由于 JSP 是基于 Java 语言和 HTML 元素的一项技术,所以只要熟悉 Java 和 HTML 的程序员都可以开发 JSP。
- 跨平台。由于 JSP 运行在 Java 虚拟机之上,所以它可以借助于 Java 本身的跨平台能力,在任何支持 Java 的平台和操作系统上运行。