




高职高专 **立体化教材** 计算机系列

JSP 编程技术

(第2版)

杨学全 主编

 赠送电子课件及
其他立体化资源



清华大学出版社

高职高专立体化教材 计算机系列

JSP 编程技术

(第 2 版)

杨学全 主 编

清华大学出版社
北京

内 容 简 介

JSP 是一种动态网页技术标准,利用这一技术可以快速构建跨平台的、先进安全的动态网站。本书全面、翔实地介绍了使用 JSP 进行 Web 应用开发所需的编程知识与技术,既有理论,又有编程实践,主要内容包括:JSP 技术概述、Web 开发基础、JSP 语法基础、JSP 内建对象、使用 JavaBean、文件访问、JSP 中使用数据库、Servlet 技术、基于 Servlet 的 MVC 模式、JSP 中使用 XML、网上报名系统开发案例等。

本书将 JSP 编程的基本知识与过程性知识、基本理论和开发实践有机整合,适合“项目驱动,案例教学,启发式学习”的教学方法;融“教、学、做”于一体,提供了所有例题及项目的源代码、电子课件和习题等资源。

本书不仅可以作为大学计算机及相关专业的教材,也可供各类培训、计算机从业人员和程序设计爱好者参考使用。

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 编程技术/杨学全主编. —2 版. —北京:清华大学出版社,2015

(高职高专立体化教材 计算机系列)

ISBN 978-7-302-38936-1

I. ①J… II. ①杨… III. ①JAVA 语言—网页制作工具—程序设计—高等职业教育—教材
IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 005671 号

责任编辑:桑任松

封面设计:刘孝琼

责任校对:周剑云

责任印制:何 芊

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

课 件 下 载: <http://www.tup.com.cn>, 010-62791865

印 装 者:三河市吉祥印务有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:26.5 字 数:641 千字

版 次:2009 年 4 月第 1 版 2015 年 3 月第 2 版 印 次:2015 年 3 月第 1 次印刷

印 数:1~3000

定 价:45.00 元

产品编号:057595-01

《高职高专立体化教材 计算机系列》

丛 书 序

一、编写目的

关于立体化教材,国内外有多种说法,有的叫“立体化教材”,有的叫“一体化教材”,有的叫“多元化教材”,其目的是一样的,就是要为学校提供一种教学资源整体解决方案,最大限度地满足教学需要,满足教育市场需求,促进教学改革。我们这里所讲的立体化教材,其内容、形式、服务都是建立在当前技术水平和条件基础上的。

立体化教材是“一揽子”式的(包括主教材、教师参考书、学习指导书、试题库)完整体系。主教材讲究的是“精品”意识,既要具备指导性和示范性,也要具有一定的适用性,喜新不厌旧。那种内容越编越多,本子越编越厚的低水平重复建设在“立体化”的世界中将被扫地出门。与以往不同,“立体化教材”中的教师参考书可不是千人一面的,教师参考书不只是提供答案和注释,而是含有与主教材配套的大量参考资料,使得老师在教学中能做到“个性化教学”。学习指导书更像一本明晰的地图册,难点、重点、学习方法一目了然。试题库或习题集则要完成对教学效果进行测试与评价的任务。这些组成部分采用不同的编写方式,把教材的精华从各个角度呈现给师生,既有重复、强调,又有交叉和补充,相互配合,形成一个教学资源有机的整体。

除了内容上的扩充外,立体化教材的最大突破还在于在表现形式上走出了“书本”这一平面媒介的局限,如果说音像制品让平面书本实现了第一次“突围”,那么电子和网络技术的大量运用,就让躺在书桌上的教材真正“活”了起来。用PowerPoint开发的电子教案不仅大大减少了教师案头备课的时间,而且也让学生的课后复习更加有的放矢。电子图书通过数字化使得教材的内容得以无限扩张,使平面教材更能发挥其提纲挈领的作用。

CAI(计算机辅助教学)课件把动画、仿真等技术引入了课堂,让课程的难点和重点一目了然,通过生动的表达方式达到深入浅出的目的。在科学指标体系控制之下的试题库,既可以轻而易举地制作标准化试卷,也能让学生进行模拟实践的在线测试,提高了教学质量评价的客观性和及时性。网络课程更厉害,它使教学突破了空间和时间的限制,彻底发挥了立体化教材本身的潜力,轻轻敲击几下键盘,你就能在任何时候得到有关课程的全部信息。

最后还有资料库,它把教学资料以知识点为单位,通过文字、图形、图像、音频、视频、动画等各种形式,按科学的存储策略组织起来,大大方便了教师在备课、开发电子教案和网络课程时的教学工作。如此一来,教材就“活”了。学生和书本之间的关系,不再像领导与被领导那样呆板,而是真正有了互动。教材不再只为老师们规定,什么重要什么不重要,而是成为教师实现其教学理念的最佳拍档。在建设观念上,从提供和出版单一纸质教材转向提供和出版较完整的教学解决方案;在建设目标上,以最大限度满足教学要求

为根本出发点;在建设方式上,不单纯以现有教材为核心,简单地配套电子音像出版物,而是以课程为核心,整合已有资源并聚拢新资源。

网络化、立体化教材的出版是我社下一阶段教材建设的重中之重,以计算机教材出版为龙头的清华大学出版社确立了“改变思想观念,调整工作模式,构建立体化教材体系,大幅度提高教材服务”的发展目标,并提出了首先以建设“高职高专计算机立体化教材”为重点的教材出版规划,希望通过邀请全国范围内的高职高专院校的优秀教师,共同策划、编写这一套高职高专立体化教材,利用网络等现代技术手段,实现课程立体化教材的资源共享,解决国内教材建设工作中存在的教材内容更新滞后于学科发展的状况。把各种相互作用、相互联系的媒体和资源有机地整合起来,形成立体化教材,把教学资料以知识点为单位,通过文字、图形、图像、音频、视频、动画等各种形式,按科学的存储策略组织起来,为高职高专教学提供一整套解决方案。

二、教材特点

在编写思想上,以适应高职高专教学改革的需要为目标,以企业需求为导向,充分吸收国外经典教材及国内优秀教材的优点,结合中国高校计算机教育的教学现状,打造立体化精品教材。

在内容安排上,充分体现先进性、科学性和实用性,尽可能选取最新、最实用的技术,并依照学生接受知识的一般规律,通过设计详细的可实施的项目化案例(而不仅仅是功能性的小例子),帮助学生掌握要求的知识点。

在教材形式上,利用网络等现代技术手段实现立体化的资源共享,为教材创建专门的网站,并提供题库、素材、录像、CAI 课件、案例分析,实现教师和学生更大范围内的教与学互动,及时解决教学过程中遇到的问题。

本系列教材采用案例式的教学方法,以实际应用为主,理论够用为度。教程中每一个知识点的结构模式为“案例(任务)提出→案例关键点分析→具体操作步骤→相关知识(技术)介绍(理论总结、功能介绍、方法和技巧等)”。

该系列教材将提供全方位、立体化的服务。网上提供电子教案、文字或图片素材、源代码、在线题库、模拟试卷、习题答案、案例动画演示、专题拓展和教学指导方案等。

在为教学服务方面,主要是通过教学服务专用网站在网络上为教师和学生提供交流的场所,每个学科、每门课程,甚至每本教材都建立网络上的交流环境。可以为广大教师信息交流、学术讨论、专家咨询提供服务,也可以让教师发表对教材建设的意见,甚至通过网络授课。对学生来说,则可以在教学支撑平台所提供的自主学习空间中进行学习、答疑、操作、讨论和测试,当然也可以对教材建设提出意见。这样,在编辑、作者、专家、教师、学生之间建立起一个以课本为依据、以网络为纽带、以数据库为基础、以网站为门户的立体化教材建设与实践的体系,用快捷的信息反馈机制和优质的教学服务促进教学改革。

前 言

近日, 国务院印发了《关于加快发展现代职业教育的决定》, 《决定》提出要牢固树立职业教育在国家人才培养体系中的重要位置, 以服务发展为宗旨, 以促进就业为导向, 适应技术进步和生产方式变革以及社会公共服务的需要, 培养数以亿计的高素质劳动者和技术技能人才。要深化产教融合、校企合作、工学结合, 推动专业设置与产业需求对接、课程内容与职业标准对接、教学过程与生产过程对接、毕业证书与职业资格证书对接、职业教育与终身学习对接, 提高人才培养质量, 强化职业教育的技术技能积累作用。

在加快职业教育发展, 提高人才培养质量的新形势下, 必须加强课程建设与改革, 推动课程内容与职业标准的对接, 教学过程与生产过程对接; 经过职教工作者们多年的探索和实践, 基于工作过程的课程开发理论得到了发展和应用, 课程开发取得了突出的成果。高职高专院校逐步构建了以技术应用能力培养为主线, 以就业为导向, 基于工作过程的计算机应用类专业课程体系。

高职高专教材是为教师、学生和课程服务的, 是知识的载体。它必须体现高职高专课程开发建设的新思想; 必须根据职业岗位(群)的任职要求, 参照国家职业资格标准开发和建设, 使其具有职业性; 必须将知识的学科性和工作的过程性有机地整合, 体现其综合性; 必须适用“教、学、做”一体化的课程教学模式, 使其具有实用性。一句话: 教材要教师用得好, 学生学得好, 学了用得上。

本书在第 1 版的基础上, 采纳了读者和同行的建议, 使用了 JDK、Tomcat 及开发工具的主流版本, 同时延续了前版的章节体系。本书是一本以职业技术能力培养为主线, 采用项目驱动模式的案例教材。教材融“教、学、做”于一体, 注重基本知识与基本技术讲解(教), 给出具有实用价值的案例供学生模仿(学), 通过课程设计强化学生能力的培养(做)。本书适用于计算机应用类专业或非计算机专业的 JSP 编程技术课程教学。全书共分为 11 章, 从基本概念和实际应用出发, 由浅入深、循序渐进地讲述 JSP 编程的基础知识、JavaBean 技术、Servlet 技术、MVC 模式和 Web 应用开发案例等内容; 通过对本书内容的学习, 读者可以快速、全面地掌握基于 MVC 模式的 JSP 编程技术; 建议教学时数为 72 学时, 也可根据教学的具体情况删减内容。

作为“项目驱动、案例教学”模式的教材, 本书具有以下特点。

(1) 内容选择合理、时序安排科学。本书以 Web 应用程序开发能力培养为主线, 根据岗位技术能力需要选择教材内容——JSP 基础、HTML 及页面布局、JavaScript 与正则表达式、JavaBean、Servlet 技术、MVC 模式、XML 以及基于 MVC 模式的 Web 应用开发等; 根据工作过程和认知规律安排内容时序为“JSP 基本知识→JSP+JavaBean 模式应用→JSP+JavaBean+Servlet 模式应用”, 将文件操作、数据库访问、XML 等编程技术合理地分配到模式 1 和模式 2 中, 强调知识的层次性和技能培养的渐进性, 最终为基于 MVC 模式框架开发打好 JSP 编程基础。

(2) 案例典型, 代码规范, 能力良构。本书以培养基于 MVC 模式的 Web 应用开发能

力为目的, 设置了具有代表性的例题、习题和案例, 比如设置了购物车、留言板、文件上传下载、分页显示、页面布局、文件操作和数据库应用等案例; 示例代码采用了 Sun 的模式 1 和模式 2, 代码规范、实用; 强调学生在例题、案例设置的工作情景中学习, 潜移默化地培养学科性知识与工作过程性知识有机整合、理论与实践相结合、具有良好结构的 JSP 编程能力。

本书由杨学全老师主编, 河北农业大学张悦、张春艳、苑萌萌等参加了部分章节的编写工作。刘海军教授审稿。

衷心感谢河北大学博士生导师徐建民教授、保定职业技术学院刘海军教授, 他们的辛勤工作使我们受益匪浅。

衷心感谢所有关心本书编写的师长和朋友。

编写一本优秀的教材是一件非常不容易的事情, 很多因素都会影响到教材的质量。尽管此书多次修改, 每次修改都考虑如何突出职业能力培养这条主线, 如何突出教材的高职特色等问题; 尽管本书的定稿经过了多人的努力, 但是我们还是感觉不太尽如人意, 唯恐对不起关心和支持我们编写这本教材的朋友们, 对不起孜孜求学的学子们。由于作者水平有限, 加之时间仓促, 书中难免有错漏之处, 敬请同行们批评指正, 我们将不胜感激。

编 者

目 录

第 1 章 JSP 技术概述 1	2.4 页面布局..... 34
1.1 Web 程序设计模式与运行原理 1	2.4.1 CSS 简介 34
1.1.1 Web 服务器与动态网页 1	2.4.2 DIV 层 41
1.1.2 浏览器/服务器结构及其优点..... 2	2.4.3 DIV+CSS 页面布局..... 49
1.1.3 JSP 与其他 Web 开发技术..... 3	2.5 上机实训..... 53
1.2 搭建 JSP 的运行环境..... 4	2.6 本章习题..... 54
1.2.1 安装和配置 JDK 4	第 3 章 JSP 语法基础 55
1.2.2 安装和配置 Tomcat 5	3.1 JSP 页面的基本结构..... 55
1.3 JSP 页面与 JSP 运行原理..... 7	3.2 JSP 脚本元素..... 56
1.3.1 第一个 JSP 页面..... 7	3.2.1 变量与方法的声明 57
1.3.2 设置 Web 服务目录 8	3.2.2 程序片 58
1.3.3 JSP 的运行原理 10	3.2.3 表达式 60
1.3.4 JSP、JavaBean 和 Java Servlet	3.3 注释..... 61
的关系..... 13	3.3.1 输出型注释 61
1.4 集成开发环境简介..... 13	3.3.2 隐藏型注释 61
1.4.1 MyEclipse 13	3.4 JSP 指令标记..... 63
1.4.2 开源的 Eclipse..... 14	3.4.1 page 指令标记..... 63
1.5 上机实训..... 16	3.4.2 include 指令..... 66
1.6 本章习题..... 17	3.5 JSP 动作标记..... 69
第 2 章 Web 开发基础 18	3.5.1 jsp:include 动作标记..... 69
2.1 HTML 简介 18	3.5.2 jsp:param 动作标记..... 70
2.1.1 什么是 HTML 18	3.5.3 jsp:forward 动作标记 71
2.1.2 什么是 URL..... 18	3.5.4 jsp:plugin 动作标记 73
2.1.3 HTML 文件结构 19	3.5.5 jsp:useBean 相关动作标记 74
2.2 常用的 HTML 标记 20	3.5.6 特殊字符 75
2.2.1 HTML 的文字标记 21	3.6 上机实训..... 76
2.2.2 特殊标记和图形标记..... 23	3.7 本章习题..... 77
2.2.3 超级链接标记..... 25	第 4 章 JSP 内建对象 78
2.3 表格..... 26	4.1 内建对象概述..... 78
2.3.1 定义表格的基本语法..... 26	4.1.1 什么是 HTTP 78
2.3.2 表格<table>标记的属性 27	4.1.2 内建对象 79
2.3.3 行<tr>标记的属性..... 30	4.2 out 对象 80
2.3.4 单元格<td>和<th>标记的	4.3 request 对象..... 82
属性..... 31	4.3.1 获取客户信息 82

4.3.2	处理汉字.....	86	6.2.1	File 类的重要属性与方法	149
4.3.3	处理表单子标记.....	88	6.2.2	查询文件属性	150
4.3.4	表单验证.....	96	6.2.3	目录管理	151
4.3.5	常用方法举例.....	103	6.3	字节流类.....	154
4.4	response 对象.....	105	6.3.1	字节流类概述	154
4.4.1	修改 ContentType 属性.....	105	6.3.2	以 File 存储类型为例介绍 字节流与缓冲流的使用	156
4.4.2	定时刷新页面.....	106	6.4	字符流类.....	161
4.4.3	重定向.....	107	6.4.1	字符流概述	161
4.4.4	改变状态码.....	108	6.4.2	以 File 存储类型为例介绍字符 流和字符缓冲流的使用	162
4.5	session 对象	108	6.5	随机读写文件.....	166
4.5.1	对象的 id 与生命周期	108	6.5.1	随机存取文件	166
4.5.2	对象存储数据.....	111	6.5.2	随机读写文件示例	168
4.5.3	对象与 URL 重写.....	112	6.6	文件操作案例.....	170
4.6	application 对象.....	114	6.6.1	上传文件	170
4.6.1	常用方法.....	114	6.6.2	下载文件	175
4.6.2	计数器.....	115	6.6.3	文件内容分页显示	178
4.7	上机实训.....	116	6.7	上机实训.....	181
4.8	本章习题.....	117	6.8	本章习题.....	182
第 5 章	使用 JavaBean.....	118	第 7 章	JSP 中使用数据库	183
5.1	JavaBean 的基本概念	118	7.1	JDBC 概述.....	183
5.1.1	什么是 JavaBean	118	7.1.1	什么是 JDBC.....	183
5.1.2	JavaBean 的规范	119	7.1.2	JDBC 的构成.....	184
5.2	创建与使用 JavaBean	120	7.2	JDBC 应用程序接口简介.....	185
5.2.1	创建 JavaBean	120	7.2.1	JDBC 的驱动程序管理器—— DriverManager 类.....	185
5.2.2	布置 JavaBean	121	7.2.2	JDBC 与数据库的连接—— Connection 接口.....	186
5.2.3	在 JSP 中使用 JavaBean	122	7.2.3	执行 SQL 语句——Statement 接口	187
5.3	JavaBean 的辅助类	129	7.2.4	数据结果集——ResultSet 接口	188
5.4	JSP 与 JavaBean 模式实例	133	7.2.5	数据库元数据—— DatabaseMetaData 和 ResultSetMetaData	189
5.4.1	计数器 Bean	133	7.3	利用 JDBC 访问数据库.....	190
5.4.2	购物车 Bean	136	7.3.1	通过 JDBC-ODBC 桥连接 来访问数据库	190
5.5	上机实训.....	145			
5.6	本章习题.....	146			
第 6 章	文件访问	147			
6.1	输入/输出流概述	147			
6.1.1	流的概念.....	147			
6.1.2	输入流与输出流.....	148			
6.1.3	字节流与字符流.....	148			
6.2	File 类	149			

7.3.2	利用本地协议纯 Java 驱动程序 连接数据库.....	194
7.3.3	配置和连接不同的数据库.....	198
7.4	数据库操作案例.....	200
7.4.1	查询数据.....	200
7.4.2	更新查询.....	209
7.4.3	分页查询.....	224
7.4.4	使用连接池.....	228
7.5	上机实训.....	231
7.6	本章习题.....	231
第 8 章	Servlet 技术.....	232
8.1	Servlet 介绍.....	232
8.1.1	什么是 Servlet.....	232
8.1.2	Servlet 的功能.....	232
8.1.3	Servlet 技术的特点.....	233
8.2	Servlet 技术原理.....	233
8.2.1	Servlet 的生命周期.....	233
8.2.2	Servlet 的结构.....	234
8.2.3	Servlet 常用类与接口的 层次关系.....	235
8.3	Servlet 的常用类、接口及其方法.....	236
8.3.1	javax.servlet 包.....	236
8.3.2	javax.servlet.http 包.....	239
8.4	编写、配置和调用 Servlet.....	242
8.4.1	编写第一个 Servlet.....	242
8.4.2	配置 Servlet.....	243
8.4.3	调用 Servlet.....	244
8.5	Servlet 的典型应用.....	246
8.5.1	读取表单数据.....	246
8.5.2	读取 cookie 数据.....	252
8.5.3	读取 session 数据.....	254
8.5.4	读取 HTTP 请求头数据.....	256
8.6	上机实训.....	258
8.7	本章习题.....	259
第 9 章	基于 Servlet 的 MVC 模式.....	260
9.1	MVC 模式介绍.....	260
9.1.1	MVC 设计模式.....	260
9.1.2	JSP 中的 MVC 模式.....	261
9.2	模型的生命周期与视图更新.....	262
9.2.1	request 周期的 JavaBean 与 视图更新.....	263
9.2.2	session 周期的 JavaBean 与 视图更新.....	263
9.2.3	application 周期.....	264
9.3	控制器的重定向与转发.....	265
9.3.1	重定向.....	265
9.3.2	转发.....	265
9.4	MVC 模式的分析.....	266
9.4.1	用户登录.....	266
9.4.2	留言板.....	272
9.4.3	访问数据库.....	277
9.5	上机实训.....	286
9.6	本章习题.....	287
第 10 章	JSP 中使用 XML.....	288
10.1	XML 简介.....	288
10.1.1	XML 文件的结构.....	288
10.1.2	XML 声明.....	290
10.1.3	XML 元素.....	290
10.1.4	XML 标记.....	291
10.2	DOM 解析器.....	293
10.2.1	什么是 DOM 解析器.....	293
10.2.2	JAXP 简介.....	294
10.2.3	使用 DOM 解析器读取 XML 文件示例.....	299
10.3	SAX 解析器.....	301
10.3.1	什么是 SAX 解析器.....	301
10.3.2	SAX 的常用接口.....	302
10.3.3	使用 SAX 解析器读取文档 内容.....	304
10.4	上机实训.....	307
10.5	本章习题.....	308
第 11 章	网上报名系统开发案例.....	309
11.1	网上报名系统设计.....	309
11.1.1	需求分析.....	309

第 1 章 JSP 技术概述

学习目的与要求:

JSP 技术是目前 Web 应用程序开发的主流技术之一。本章主要学习 Web 程序设计模式与原理,搭建 JSP 运行环境, JSP 页面与 JSP 运行原理和 JSP 集成开发环境等内容。通过本章的学习,学生要了解 B/S 模式 Web 应用程序架构, JSP 运行原理, JSP 与 JavaBean、Servlet 的关系,理解 Web 服务目录的概念,掌握搭建 JSP 开发环境的方法以及集成开发环境 MyEclipse 的安装和配置,能够编写一个简单的 JSP 页面,并能通过客户端浏览器访问页面。学生必须认真学习本章,体验学习 JSP 的成功。万事开头难,良好的开端就是成功的一半。

1.1 Web 程序设计模式与运行原理

用户在学习 JSP 编程技术之前,需要对 Web 程序设计模式有所了解。Web 程序或网站的运行方式不同于单机或 C/S 模式的 Windows 应用程序,本节主要从 Web 服务、浏览器/服务器模式与动态网页技术三个方面对其简述。

1.1.1 Web 服务器与动态网页

互联网中有数以亿计的网站,用户可以通过浏览这些网站获得所需要的信息。例如,用户在浏览器的地址栏中输入 <http://www.sina.com.cn>,浏览器就会显示新浪网的首页,查看新闻等信息。那么新浪网首页的内容是存放在哪里的呢?新浪网首页的内容是存放在新浪网服务器上的。所谓服务器就是网络中的一台主机,由于它提供 Web、FTP 等网络服务,因此称其为服务器。用户的计算机又是如何将存在网络服务器上的网页显示在浏览器中的呢?当用户在地址栏中输入新浪网地址(URL,即“统一资源定位符”)的时候,浏览器会向新浪网的服务器发送 HTTP 请求,这个请求使用 HTTP,其中包括请求的主机名、HTTP 版本号等信息。服务器在收到请求信息后,将回复的信息(一般是文字、图片等网页信息,也就是 HTML 页面)准备好,再通过网络发送到客户端浏览器。客户端的浏览器在接收到服务器传回的信息后,将其解释并显示在浏览器的窗口中,这样用户就可以进行浏览了。整个过程如图 1.1 所示。

在这个“请求—响应”过程中,如果在服务器上存放的网页为静态 HTML 网页文件,服务器就会原封不动地返回网页的内容。如果存放的是动态网页,如 JSP、ASP、ASP.NET 等文件,则服务器会执行动态网页,执行的结果是生成一个 HTML 文件,然后再将这个 HTML 文件发送给客户端浏览器,客户端浏览器将其解释为用户见到的页面。



图 1.1 Web 服务过程

因此,静态网页和动态网页的根本区别在于服务器端返回的 HTML 文件是事先存储好的还是由动态网页程序生成的。静态网页文件里只有 HTML 标记,没有程序代码,网页的内容都是事先写好,存放在服务器上的;动态网页文件不仅含有 HTML 标记,并且还含有程序代码,当用户发出请求时,服务器由动态网页程序即时生成 HTML 文件。动态网页能够根据不同的时间、不同的用户生成不同的 HTML 文件,显示不同的内容。

1.1.2 浏览器/服务器结构及其优点

随着网络技术的不断发展,单机的软件程序已经难以满足网络计算的需求,因此,基于网络的软件架构应运而生。早期常用的网络架构为“客户/服务器”(即 Client/Server,简称为 C/S)模式。使用这种架构编写的软件分为客户端和服务端两部分,需要分别在客户机和服务器上安装。这种模式在用户数据录入等方面很有优势,也降低了系统的通信开销,但是也有一定的缺点,如开发和维护成本较高,可移植性较差等。

互联网的普及使得用于上网浏览的浏览器已经成为操作系统中不可缺少的一项,浏览器的功能也越来越强大,甚至可以取代“客户/服务器”架构的客户端软件,成为统一的客户端。这样,程序员就可以只编写运行在服务器上的软件,浏览器代替 C/S 模式中的客户端软件,客户通过浏览器与服务器端软件进行交互并得到运算结果,这种软件架构就是“浏览器/服务器”(即 Browser/Server,简称为 B/S)模式。B/S 模式主要是利用了不断成熟的 WWW 浏览器技术,结合动态网站制作技术,通过通用浏览器实现了原来需要复杂的专用软件才能实现的强大功能,节约了开发成本,是一种全新的软件系统构造技术。随着互联网络的不断发展,B/S 架构已经成为当今应用软件的首选体系结构。

B/S 模式的应用程序相对于传统的 C/S 模式的应用程序来讲无疑是一个巨大的进步。它的主要优点如下。

1. 开发、维护成本较低

C/S 模式软件,当客户端的软件需要升级的时候,所有客户端都必须进行升级安装或者重新安装,而 B/S 模式的软件只需要在服务器端发布,客户端浏览器无须维护,因而极大地降低了开发和维护成本。

2. 可移植性高

C/S 模式软件,不同开发工具开发的程序,一般情况下互不兼容,主要运行在局域网

中,移植困难。而 B/S 模式的软件运行在互联网上,提供了异种网、异种机、异种应用服务的联机、联网服务基础,客户端安装的是通用浏览器,不存在移植的问题。

3. 用户界面统一

C/S 模式软件的客户端界面由所安装的客户端软件所决定,因此不同的软件客户端界面不同,而 B/S 模式的软件都是通过浏览器来使用的,操作界面基本统一。

1.1.3 JSP 与其他 Web 开发技术

在简单介绍了 Web 服务器、动态网页和 B/S 模式 Web 应用程序结构的优点之后,读者要问哪些语言和技术可用于 B/S 模式的 Web 应用程序开发?目前使用较多的技术有 JSP、ASP、ASP.NET、PHP 等。本节对它们进行简单的介绍和比较。

1. JSP 技术及其优点

JSP 全称为 Java Server Pages,是 Sun 公司倡导、多家公司参与,于 1999 年提出的一种 Web 服务技术标准。它的主要的编程脚本为 Java 语言,同时还支持 JavaBeans/Servlet 等技术,利用这些技术可以建立安全、跨平台的 Web 应用程序。JSP 技术具有以下优点。

1) 跨平台性

由于 JSP 的脚本语言是 Java 语言,因此它具有 Java 语言的一切特性。同时, JSP 也支持现在大部分平台,拥有“一次编写,到处运行”的特点。

2) 执行效率高

当 JSP 第一次被请求时, JSP 页面转换成 Servlet,然后被编译成*.class 文件,以后(除非页面有改动或 Web 服务器被重新启动)再有客户请求该 JSP 页面时, JSP 页面不被重新编译,而是直接执行已编译好的*.class 文件,因此执行效率高。

3) 可重用性

可重用的、跨平台的 JavaBeans 和 EJB(Enterprise JavaBeans)组件,为 JSP 程序的开发提供方便。例如,用户可以将复杂的处理程序(如对数据库的操作)封装到组件中,在开发中可以多次使用这些组件,提高了组件的可重用性。

4) 将内容的生成和显示进行分离

使用 JSP 技术, Web 页面开发人员可以使用 HTML 或者 XML 标记来设计和格式化最终页面。生成动态内容的程序代码封装在 JavaBean 组件、EJB 组件或 JSP 脚本段中。在最终页面中使用 JSP 标记或脚本将 JavaBean 组件中的动态内容引入。这样,可以有效地将内容生成和页面显示进行分离,使页面的设计人员和编程人员可以同步进行,也可以保护程序的关键代码。

2. 其他 Web 开发技术

1) ASP 技术

ASP 是 Active Server Pages 的缩写,是微软在早期推出的动态网页制作技术,包含在 IIS(Internet 信息服务)中,是一种服务器端的脚本编写环境,使用它可以创建和运行动态、交互的 Web 服务器应用程序。在动态网页技术发展的早期, ASP 是绝对的主流技术,但是它也存在着许多缺陷:由于 ASP 的核心是脚本语言,决定了它的先天不足,它

无法进行像传统编程语言那样的底层操作；由于 ASP 通过解释执行代码，因此运行效率较低；同时由于脚本代码与 HTML 代码混在一起，不便于开发人员进行管理与维护。随着技术的发展，ASP 的辉煌已经成为过去，微软也已经不再对 ASP 提供技术支持和更新，ASP 技术目前处于被淘汰的边缘。

2) PHP 技术

PHP 从语法和编写方式上来看与 ASP 类似，是完全免费的，最早是一个小开放源码的软件，随着越来越多的人意识到它的实用性而逐渐发展起来。Rasmus Lerdorf 在 1994 年发布了 PHP 的第一个版本。从那时起它就飞速发展，在原始发行版上经过无数的改进和完善，现在已经发展到 5.5 版。PHP+MySQL+Linux 的组合是最常见的，因为它们都可以免费获得。但是 PHP 的弱点也是很明显的，例如 PHP 不支持真正意义上的面向对象编程，接口支持不统一，缺乏正规支持，不支持多层结构和分布式计算等。

3) ASP.NET

ASP.NET 是微软继 ASP 后推出的全新动态网页制作技术，目前最新版本为 .NET 4.0。在性能上，ASP.NET 比 ASP 强很多，与 PHP 相比，也存在明显的优势。ASP.NET 可以使用 C#(读音为 C Sharp)，VB.NET，Visual J# 等语言来开发，程序开发人员可以选择自己习惯或熟悉的语言进行开发。ASP.NET 依托 .NET 平台先进而强大的功能，极大地简化了编程人员的工作量，使得 Web 应用程序的开发更加方便、快捷，同时也使得程序的功能更加强大，是 JSP 技术的有力竞争对手。

1.2 搭建 JSP 的运行环境

用户学习 JSP 或者使用 JSP 开发 Web 应用程序，必须搭建一个 JSP 运行环境。JSP 运行环境至少要具备三个基本条件：一是要在用户的计算机上安装 Java 的 JDK，并进行环境变量的设置；二要在计算机上安装 JSP 引擎，例如 Tomcat、J2EE、WebLogic、WebSphere 服务器；三要在计算机上有浏览器。搭建 JSP 运行环境是 JSP 的初学者必须要学习的基础知识，本节将引导读者一步步完成。

1.2.1 安装和配置 JDK

Sun 公司提供了一个免费的 Java 软件开发工具包 JDK(Java Development Kit)，该工具包包含了编译、运行及调试 Java 程序所需要的工具，此外还提供了大量的基础类库，供编写程序使用，它是开发 Java 程序的基础。2009 年 4 月 20 日，Oracle 宣布正式以 74 亿美元的价格收购 Sun 公司，Java 商标从此正式归 Oracle 所有。

1. JDK 的安装

Sun 公司为不同的操作系统平台，如 Windows、Unix/Linux 等提供了相应的 Java 开发包。用户可到 Oracle 公司网站 <http://www.oracle.com> 下载最新的适用于相应操作系统的开发包。本书中使用 Windows 32 位操作系统环境下的 Java 开发包 `jdk-8u5-windows-i586.exe`。书中的实例程序均在此版本下运行通过，所使用的操作系统为 Windows 7 旗舰版。

下载完成后，运行 jdk-8u5-windows-i586.exe 安装文件，本书安装目录为：C:\Program Files\Java。安装完成后在 C:\Program Files\Java 目录中会有 jdk1.8.0_05 和 jre8 两个子目录，jdk1.8.0_05 为 Java 开发工具目录，jre8 为 Java 运行环境目录。

2. JDK 的配置

安装完 JDK 后，需要在 Windows 操作系统中为 JDK 设置几个环境变量，以便系统能够自动查找 JDK 的命令和类库。对于 Windows 7 系统，从“控制面板”→“系统安全”→“系统”→“高级系统设置”可调出“系统属性”对话框，单击“系统属性”对话框的“高级”选项卡，然后单击“环境变量”按钮，弹出“环境变量”设置对话框，如图 1.2、图 1.3 和图 1.4 所示。

分别添加如下的环境变量。

变量名：JAVA_HOME，变量值：C:\Program Files\Java\jdk1.8.0_05。

变量名：CLASSPATH，变量值：

.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar;

变量名：PATH，变量值：%JAVA_HOME%\bin;

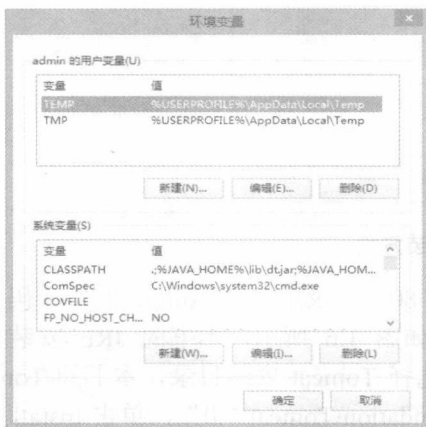


图 1.2 “环境变量”对话框



图 1.3 设置 JAVA_HOME

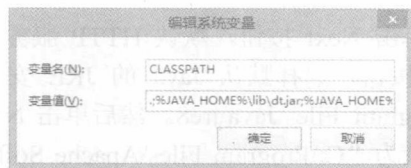


图 1.4 设置 CLASSPATH

注意：如果用户已经有需要设置的环境变量，如“PATH”变量，可选中该变量进行编辑操作，将需要的变量值追加在后面即可，值与值之间用“;”分隔，切记不要把原来的值覆盖。CLASSPATH 变量中的“.”不能少，其含义是在当前目录寻找类库。为了验证用户环境变量设置是否正确，可用记事本编写一个简单的 Java 程序，对其进行编译、执行，用来验证 JDK 的安装。

1.2.2 安装和配置 Tomcat

自 1999 年 JSP 发布以来，到目前为止出现了各种各样的 JSP 引擎。如 Tomcat、J2EE、WebLogic、WebSphere 等引擎。一般将安装了 JSP 引擎的计算机称为一个支持 JSP 的 Web 服务器，它负责运行 JSP 程序，并将执行结果返回给浏览器。Tomcat 是一个免费的开源 JSP 引擎，也称为 Jakarta Tomcat Web 服务器。目前 Tomcat 能和大多数主流 Web

服务器一起高效的工作。

1. 下载和安装 Tomcat

用户可以到 <http://tomcat.apache.org/> 站点免费下载 Tomcat 7.0。在主页中的 Download 里选择 Tomcat 7.0, 然后在 Binary Distributions 里的 Core 中选择 zip(pgp, md5)、tar.gz(pgp, md5)或 32-bit/64-bit Windows Service Installer (pgp, md5)。本书下载的是 32-bit/64-bit Windows Service Installer(pgp, md5), 文件名为: apache-tomcat-7.0.53.exe。apache-tomcat-7.0.53.exe 是专门为 Windows 开发的 Tomcat 服务器。

双击 apache-tomcat-7.0.53.exe 文件, 出现安装向导, 单击 Next 按钮, 出现“授权”界面, 接受授权协议后, 用户可以选择 Normal、Minimun、Custom 和 Full 安装形式, 本书选择“Full”安装模式, 如图 1.5 所示。

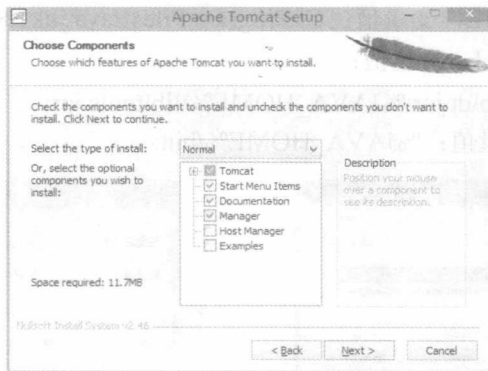


图 1.5 选择安装方式

单击 Next 按钮, 默认 HTTP 服务端口号 8080、登录用户名 Admin 和空密码。单击 Next 按钮, 选择默认 Java 的 JRE 安装目录, 如图 1.6 所示, 本书的 JRE 安装目录为 C:\Program Files\Java\jre8。然后单击 Next 按钮选择 Tomcat 安装目录, 本书的 Tomcat 安装目录为“C:\Program Files\Apache Software Foundation\Tomcat 7.0”, 单击 Install 按钮开始 Tomcat 的安装。安装完成后, 在 C 盘中会有安装程序创建的 Apache Tomcat 7.0 菜单组, 产生的目录结构如图 1.7 所示。

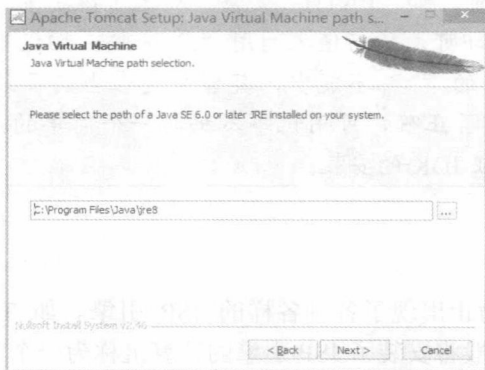


图 1.6 选择 JRE 安装目录

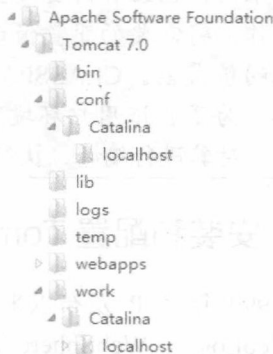


图 1.7 Tomcat 7.0 目录结构