

21世纪高等职业教育信息技术类规划教材

21 Shiji Gaodeng Zhiye Jiaoyu Xinxì Jìshùlèi Guìhuà Jiàocái

# JSP程序设计 基础教程

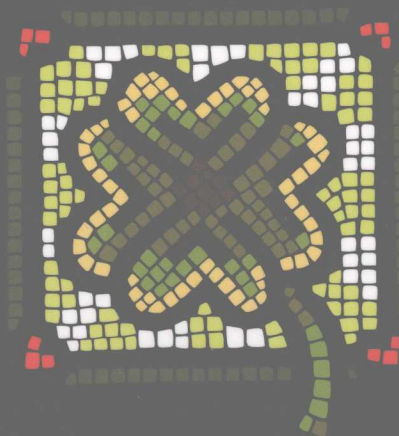
JSP CHENGXU SHEJI JICHU JIAOCHENG

苗连强 主编 王国辉 宋来 李振军 副主编 刘志成 审

■ 由浅入深的理论介绍

■ 结合实例的技术讲解

■ 巩固知识的综合实训



人民邮电出版社  
POSTS & TELECOM PRESS

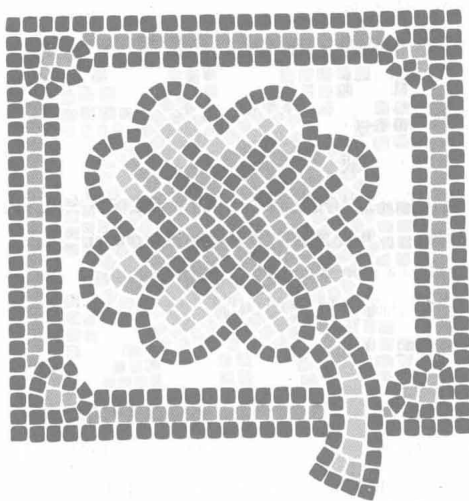
21世纪高等职业教育信息技术类规划教材

21 Shiji Gaodeng Zhiye Jiaoyu Xinxi Jishulei Guihua Jiaocai

# JSP程序设计 基础教程

JSP CHENGXU SHEJI JICHU JIAOCHENG

苗连强 主编 王国辉 宋来 李振军 副主编 刘志成 审



人民邮电出版社

北京

## 图书在版编目 (CIP) 数据

JSP程序设计基础教程 / 苗连强主编. —北京: 人民邮电出版社, 2009.6  
21世纪高等职业教育信息技术类规划教材  
ISBN 978-7-115-20638-1

I. J… II. 苗… III. JAVA语言—主页制作—程序设计—高等学校: 技术学校—教材 IV. TP393.092

中国版本图书馆CIP数据核字 (2009) 第048136号

## 内 容 提 要

本书系统地介绍了 JSP 程序设计的相关知识。全书共分 9 章, 内容包括 JSP 概述、JSP 开发基础、JSP 语法、JSP 内置对象、JavaBean 技术、Servlet 技术、JSP 实用组件、JSP 数据库应用开发和综合实训。本书所有知识都结合具体实例进行讲解, 力求详略得当, 使读者快速掌握 JSP 程序设计的相关技术。

本书可作为高职高专计算机及相关专业 JSP 课程的教材, 也适合 JSP 初学者及程序开发人员参考使用。

21 世纪高等职业教育信息技术类规划教材

### JSP 程序设计基础教程

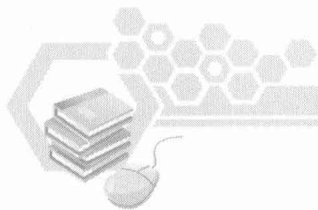
- 
- ◆ 主 编 苗连强
  - 副 主 编 王国辉 宋 来 李振军
  - 审 刘志成
  - 责任编辑 潘春燕
  - 执行编辑 王 威
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京鑫正太印刷有限公司印刷
  - ◆ 开本: 787×1092 1/16  
印张: 16  
字数: 408 千字 2009 年 6 月第 1 版  
印数: 1—3 000 册 2009 年 6 月北京第 1 次印刷

---

ISBN 978-7-115-20638-1/TP

定价: 26.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223  
反盗版热线: (010)67171154



JSP (Java Server Page) 是由 Sun 公司在 Java 语言基础上开发出来的一种动态网页制作技术。它是 Java 开发阵营中最具代表性的解决方案。JSP 是比较热门的一种动态网页技术, 它依靠 Java 语言的稳定、安全、可移植性好等优点, 成为大、中型网站开发的首选。

高职高专院校希望培养的学生能在未来就业中熟练使用所学知识与技能, 因此也将 JSP 作为一门重要的程序设计课程。本书在编写过程中力求突出高职教育特点, 坚持基础知识管用、理论知识够用、专业知识实用、专业技能会用的编写原则, 以培养专业技术应用能力和职业操作技能为教材编写重点, 将教材中的教学内容与职业培养目标相结合, 注重实践教学, 力求教学内容简洁、实用。

本书用通俗易懂的语言和实用生动的例子, 系统地介绍了 JSP 开发技术、并且在每一章的后面还提供了习题, 方便读者及时验证自己的学习效果。

为方便教师教学, 本书配备了内容丰富的教学资源, 包括所有程序源代码、PPT 电子教案、教学大纲。任课老师可登录人民邮电出版社教学服务与资源网 ([www.ptpedu.com.cn](http://www.ptpedu.com.cn)) 下载使用。

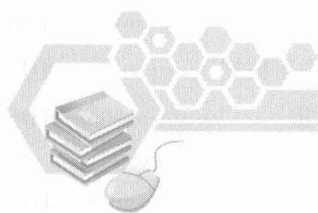
全书内容共分 9 章。第 1 章主要介绍 JSP 技术的相关概念以及 JSP 开发环境和开发工具, 包括 JSP 技术概述、JSP 的技术特征、JSP 的处理过程、JSP 与其他服务器端脚本语言的比较、JSP 开发环境搭建、JSP 开发工具及 JSP 程序开发模式等内容; 第 2 章主要介绍 JSP 开发基础, 包括 Java 语言基础和客户端脚本语言 JavaScript; 第 3 章主要介绍 JSP 语法, 主要包括 JSP 的基本构成、JSP 的指令标识、JSP 的脚本标识、JSP 的注释和动作标识; 第 4 章主要介绍 JSP 内置对象, 包括 request 对象、response 对象、session 对象、application 对象、out 对象和其他内置对象。第 5 章介绍 JavaBean 技术, 主要包括 JavaBean 概述、创建 JavaBean、JavaBean 的应用和 JavaBean 的应用实例; 第 6 章主要介绍 Servlet 技术, 主要包括 Servlet 技术简介、Servlet 开发和 Servlet 的应用实例; 第 7 章介绍 JSP 实用组件, 主要包括 JSP 文件操作、发送 E-mail、JSP 动态图表和 JSP 报表; 第 8 章介绍 JSP 数据库应用开发, 包括 SQL Server 2000 数据库管理系统、JDBC 简介、JDBC 中的常用接口、连接数据库、典型 JSP 数据库连接和数据库操作技术; 第 9 章实训, 是为巩固各章所学内容提供的实训练习。

本书由苗连强任主编, 王国辉、宋来、李振军任副主编, 其中第 1~4 章由宋来编写, 第 5~8 章由李振军编写, 第 9~12 章由苗连强编写, 程序代码由王国辉编写并测试, 全书由苗连强统稿, 湖南铁道职院的刘志成老师审阅全稿。

虽然在编写本书的过程中, 我们本着科学、严谨的态度, 力求精益求精, 但错误、疏漏之处在所难免, 敬请广大读者批评指正。

编 者

2009 年 2 月



# 目 录

|                                     |    |                                  |    |
|-------------------------------------|----|----------------------------------|----|
| 第 1 章 JSP 概述 .....                  | 1  | 2.1.2 基本数据类型及基本数据类型间的转换 .....    | 35 |
| 1.1 JSP 技术概述 .....                  | 2  | 2.1.3 常量与变量 .....                | 36 |
| 1.1.1 Java 语言 .....                 | 2  | 2.1.4 运算符的应用 .....               | 37 |
| 1.1.2 Servlet 技术 .....              | 3  | 2.1.5 流程控制语句 .....               | 40 |
| 1.1.3 JavaBean 技术 .....             | 3  | 2.1.6 字符串处理 .....                | 45 |
| 1.1.4 JSP 技术 .....                  | 4  | 2.1.7 数组的创建与应用 .....             | 48 |
| 1.2 JSP 的技术特征 .....                 | 4  | 2.1.8 集合类的应用 .....               | 49 |
| 1.3 JSP 的处理过程 .....                 | 5  | 2.1.9 异常处理语句 .....               | 51 |
| 1.4 JSP 与其他服务器端脚本语言的比较 .....        | 6  | 2.2 客户端脚本语言 JavaScript .....     | 52 |
| 1.4.1 CGI .....                     | 6  | 2.2.1 JavaScript 脚本语言概述 .....    | 53 |
| 1.4.2 ASP .....                     | 6  | 2.2.2 在 JSP 中引入 JavaScript ..... | 53 |
| 1.4.3 PHP .....                     | 6  | 2.2.3 JavaScript 的数据类型与运算符 ..... | 53 |
| 1.4.4 ASP.NET .....                 | 7  | 2.2.4 JavaScript 的流程控制语句 .....   | 56 |
| 1.5 JSP 开发环境搭建 .....                | 7  | 2.2.5 函数的定义和调用 .....             | 57 |
| 1.5.1 JSP 的运行环境 .....               | 7  | 2.2.6 事件 .....                   | 59 |
| 1.5.2 JDK 的安装与配置 .....              | 8  | 2.2.7 JavaScript 常用对象的应用 .....   | 60 |
| 1.5.3 Tomcat 的安装与启动 .....           | 11 | 习 题 .....                        | 63 |
| 1.6 JSP 开发工具 .....                  | 13 | 第 3 章 JSP 语法 .....               | 64 |
| 1.6.1 Eclipse 的安装 .....             | 13 | 3.1 了解 JSP 的基本构成 .....           | 65 |
| 1.6.2 Eclipse 的使用 .....             | 19 | 3.1.1 JSP 中的指令标识 .....           | 66 |
| 1.7 JSP 程序开发模式 .....                | 23 | 3.1.2 HTML 简介 .....              | 66 |
| 1.7.1 单纯的 JSP 页面编程 .....            | 23 | 3.1.3 嵌入的 Java 代码片段 .....        | 66 |
| 1.7.2 JSP+JavaBean 编程 .....         | 23 | 3.1.4 JSP 表达式 .....              | 66 |
| 1.7.3 JSP+Servlet+JavaBean 编程 ..... | 24 | 3.2 JSP 的指令标识 .....              | 66 |
| 1.7.4 MVC 模式 .....                  | 25 | 3.2.1 使用 page 指令 .....           | 67 |
| 1.8 第一个 JSP 应用 .....                | 25 | 3.2.2 使用 include 指令 .....        | 69 |
| 习 题 .....                           | 26 | 3.2.3 使用 taglib 指令 .....         | 70 |
| 第 2 章 JSP 开发基础 .....                | 27 | 3.3 JSP 的脚本标识 .....              | 70 |
| 2.1 Java 语言基础 .....                 | 28 | 3.3.1 JSP 表达式 (Expression) ..... | 70 |
| 2.1.1 面向对象程序设计 .....                | 29 | 3.3.2 声明标识 (Declaration) .....   | 71 |



|                                       |     |                                           |     |
|---------------------------------------|-----|-------------------------------------------|-----|
| 3.3.3 脚本程序 (Scriptlet) .....          | 72  | 4.6.1 管理响应缓冲 .....                        | 105 |
| 3.4 JSP 的注释 .....                     | 73  | 4.6.2 向客户端输出数据 .....                      | 105 |
| 3.4.1 HTML 中的注释 .....                 | 73  | 4.7 其他内置对象 .....                          | 106 |
| 3.4.2 带有 JSP 表达式的注释 .....             | 73  | 4.7.1 获取会话范围的 pageContext<br>对象 .....     | 106 |
| 3.4.3 隐藏注释 .....                      | 74  | 4.7.2 读取 web.xml 配置信息的 config<br>对象 ..... | 106 |
| 3.4.4 脚本程序 (Scriptlet) 中的<br>注释 ..... | 74  | 4.7.3 应答或请求的 page 对象 .....                | 107 |
| 3.5 动作标识 .....                        | 76  | 4.7.4 获取异常信息的 exception<br>对象 .....       | 107 |
| 3.5.1 <jsp:include> .....             | 77  | 习题 .....                                  | 108 |
| 3.5.2 <jsp:forward> .....             | 79  | 第 5 章 JavaBean 技术 .....                   | 109 |
| 3.5.3 <jsp:useBean> .....             | 79  | 5.1 JavaBean 概述 .....                     | 110 |
| 3.5.4 <jsp:setProperty> .....         | 82  | 5.1.1 JavaBean 技术介绍 .....                 | 111 |
| 3.5.5 <jsp:getProperty> .....         | 85  | 5.1.2 JavaBean 的种类 .....                  | 111 |
| 3.5.6 <jsp:fallback> .....            | 86  | 5.1.3 JavaBean 规范 .....                   | 112 |
| 3.5.7 <jsp:plugin> .....              | 86  | 5.2 创建 JavaBean .....                     | 113 |
| 习题 .....                              | 89  | 5.3 JavaBean 的应用 .....                    | 115 |
| 第 4 章 JSP 内置对象 .....                  | 90  | 5.3.1 获取 JavaBean 信息 .....                | 115 |
| 4.1 JSP 内置对象概述 .....                  | 91  | 5.3.2 向 JavaBean 中发送信息 .....              | 117 |
| 4.2 request 对象 .....                  | 92  | 5.3.3 在 JSP 页面中应用工具<br>JavaBean .....     | 121 |
| 4.2.1 访问请求参数 .....                    | 92  | 5.4 JavaBean 的应用实例 .....                  | 124 |
| 4.2.2 在作用域中管理属性 .....                 | 93  | 5.4.1 应用 JavaBean 解决中文乱码 .....            | 124 |
| 4.2.3 获取 Cookie .....                 | 94  | 5.4.2 应用 JavaBean 实现购物车<br>程序 .....       | 127 |
| 4.2.4 获取客户信息 .....                    | 95  | 习题 .....                                  | 134 |
| 4.2.5 访问安全信息 .....                    | 96  | 第 6 章 Servlet 技术 .....                    | 135 |
| 4.2.6 访问国际化信息 .....                   | 97  | 6.1 Servlet 技术简介 .....                    | 136 |
| 4.3 response 对象 .....                 | 97  | 6.1.1 什么是 Servlet .....                   | 136 |
| 4.3.1 重定向网页 .....                     | 97  | 6.1.2 Servlet 技术特点 .....                  | 137 |
| 4.3.2 设置 HTTP 响应报头 .....              | 98  | 6.1.3 Servlet 的生命周期 .....                 | 137 |
| 4.3.3 缓冲区配置 .....                     | 100 | 6.1.4 Servlet 与 JSP 的区别 .....             | 138 |
| 4.4 session 对象 .....                  | 100 | 6.1.5 开发简单的 Servlet 程序 .....              | 139 |
| 4.4.1 创建及获取客户的会话 .....                | 101 | 6.2 Servlet 开发 .....                      | 140 |
| 4.4.2 从会话中移除指定的对象 .....               | 101 | 6.2.1 Servlet 的创建 .....                   | 140 |
| 4.4.3 销毁 session .....                | 102 | 6.2.2 Servlet 的配置 .....                   | 141 |
| 4.4.4 会话超时的管理 .....                   | 102 |                                           |     |
| 4.5 application 对象 .....              | 103 |                                           |     |
| 4.5.1 访问应用程序初始化参数 .....               | 103 |                                           |     |
| 4.5.2 管理应用程序环境属性 .....                | 104 |                                           |     |
| 4.6 out 对象 .....                      | 105 |                                           |     |





|                                                 |            |                                                |            |
|-------------------------------------------------|------------|------------------------------------------------|------------|
| 6.2.3 编写生成验证码的 Servlet .....                    | 142        | 8.1.4 创建数据表 .....                              | 187        |
| 6.2.4 在 Servlet 中实现页面转发 .....                   | 145        | 8.2 JDBC 简介 .....                              | 188        |
| 6.3 Servlet 的应用实例 .....                         | 146        | 8.2.1 JDBC 技术介绍 .....                          | 188        |
| 6.3.1 应用 Servlet 获取表单数据 .....                   | 146        | 8.2.2 JDBC 驱动程序 .....                          | 188        |
| 6.3.2 应用 Servlet 读取文件 .....                     | 148        | 8.3 JDBC 中的常用接口 .....                          | 189        |
| 6.3.3 应用 Servlet 写入文件 .....                     | 150        | 8.3.1 驱动程序接口 Driver .....                      | 189        |
| 习题 .....                                        | 152        | 8.3.2 驱动程序管理器<br>DriverManager .....           | 190        |
| <b>第 7 章 JSP 实用组件</b> .....                     | <b>153</b> | 8.3.3 数据库连接接口 Connection .....                 | 191        |
| 7.1 JSP 文件操作 .....                              | 154        | 8.3.4 执行 SQL 语句接口 Statement .....              | 191        |
| 7.1.1 jspSmartUpload 组件的安装与<br>配置 .....         | 155        | 8.3.5 执行动态 SQL 语句接口<br>PreparedStatement ..... | 192        |
| 7.1.2 jspSmartUpload 组件中的常<br>用类 .....          | 155        | 8.3.6 执行存储过程接口 Callable<br>Statement .....     | 193        |
| 7.1.3 采用 jspSmartUpload 组件进行<br>文件操作 .....      | 160        | 8.3.7 访问结果集接口 ResultSet .....                  | 193        |
| 7.2 发送 E-mail .....                             | 162        | 8.4 连接数据库 .....                                | 194        |
| 7.2.1 Java Mail 组件简介 .....                      | 163        | 8.4.1 加载 JDBC 驱动程序 .....                       | 194        |
| 7.2.2 Java Mail 核心类简介 .....                     | 163        | 8.4.2 创建数据库连接 .....                            | 195        |
| 7.2.3 搭建 Java Mail 的开发环境 .....                  | 167        | 8.4.3 执行 SQL 语句 .....                          | 195        |
| 7.2.4 在 JSP 中应用 Java Mail 组件<br>发送 E-mail ..... | 167        | 8.4.4 获得查询结果 .....                             | 196        |
| 7.3 JSP 动态图表 .....                              | 170        | 8.4.5 关闭连接 .....                               | 197        |
| 7.3.1 JFreeChart 的下载与使用 .....                   | 170        | 8.5 典型 JSP 数据库连接 .....                         | 197        |
| 7.3.2 JFreeChart 的核心类 .....                     | 170        | 8.5.1 SQL Server 2000 数据库的<br>连接 .....         | 197        |
| 7.3.3 利用 JFreeChart 生成动态<br>图表 .....            | 171        | 8.5.2 Access 数据库的连接 .....                      | 198        |
| 7.4 JSP 报表 .....                                | 173        | 8.5.3 MySQL 数据库的连接 .....                       | 199        |
| 7.4.1 iText 组件简介 .....                          | 173        | 8.6 数据库操作技术 .....                              | 200        |
| 7.4.2 iText 组件的下载与配置 .....                      | 173        | 8.6.1 查询操作 .....                               | 200        |
| 7.4.3 应用 iText 组件生成 JSP 报表 .....                | 174        | 8.6.2 添加操作 .....                               | 202        |
| 习题 .....                                        | 182        | 8.6.3 修改操作 .....                               | 203        |
| <b>第 8 章 JSP 数据库应用开发</b> .....                  | <b>183</b> | 8.6.4 删除操作 .....                               | 205        |
| 8.1 SQL Server 2000 数据库管理<br>系统 .....           | 184        | 习题 .....                                       | 206        |
| 8.1.1 安装 SQL Server 2000 .....                  | 185        | <b>第 9 章 实训</b> .....                          | <b>207</b> |
| 8.1.2 安装 SQL Server 2000 SP4 补丁 .....           | 186        | 9.1 实训 1 开发环境的搭建与<br>开发工具的使用 .....             | 207        |
| 8.1.3 创建数据库 .....                               | 187        | 9.1.1 实训目的 .....                               | 207        |
|                                                 |            | 9.1.2 实训内容 .....                               | 207        |
|                                                 |            | 9.1.3 实训步骤 .....                               | 208        |



|                               |     |                              |     |
|-------------------------------|-----|------------------------------|-----|
| 9.2 实训 2 JSP 开发基础 .....       | 215 | 9.5.2 实训内容 .....             | 225 |
| 9.2.1 实训目的 .....              | 215 | 9.5.3 实训步骤 .....             | 225 |
| 9.2.2 实训内容 .....              | 215 | 9.6 实训 6 Servlet 技术的应用 ..... | 229 |
| 9.2.3 实训步骤 .....              | 215 | 9.6.1 实训目的 .....             | 229 |
| 9.3 实训 3 JSP 语法 .....         | 220 | 9.6.2 实训内容 .....             | 229 |
| 9.3.1 实训目的 .....              | 220 | 9.6.3 实训步骤 .....             | 229 |
| 9.3.2 实训内容 .....              | 220 | 9.7 实训 7 JSP 实用组件的应用 .....   | 233 |
| 9.3.3 实训步骤 .....              | 220 | 9.7.1 实训目的 .....             | 233 |
| 9.4 实训 4 JSP 内置对象 .....       | 223 | 9.7.2 实训内容 .....             | 233 |
| 9.4.1 实训目的 .....              | 223 | 9.7.3 实训步骤 .....             | 233 |
| 9.4.2 实训内容 .....              | 223 | 9.8 实训 8 数据库应用开发 .....       | 240 |
| 9.4.3 实训步骤 .....              | 223 | 9.8.1 实训目的 .....             | 240 |
| 9.5 实训 5 JavaBean 技术的应用 ..... | 225 | 9.8.2 实训内容 .....             | 240 |
| 9.5.1 实训目的 .....              | 225 | 9.8.3 实训步骤 .....             | 241 |



# 第1章

## JSP 概述

### 【学习目标】

随着 Internet 和电子商务的普遍应用,陆续诞生了各种动态网页技术。其中, JSP 自从发布以来,在一些主流的动态网页技术中一直受到了密切的关注。JSP 是由 Sun 公司倡导,与多个公司共同建立的一种技术标准,它建立在 Servlet 之上。应用 JSP,程序员和非程序员都可以高效率地创建 Web 应用,并使得开发的 Web 应用具有安全性高、跨平台等优点。

本章首先对 JSP 技术进行简要说明,然后介绍如何搭建开发 JSP 网站所必须的开发环境及开发工具 Eclipse 的使用。通过学习本章,读者可以达到以下学习目的:

- 了解什么是 JSP 及 JSP 相关的概念;
- 了解 JSP 的技术特征及处理过程;
- 掌握 JDK 及 Tomcat 的安装与配置;
- 掌握 Eclipse 开发工具的使用;
- 理解 JSP 程序开发模式;
- 掌握通过 Eclipse 开发 JSP 动态网站的基本过程。

### 【学习导航】

本章主要介绍学习 JSP 需要掌握和了解的一些技术与概念。例如,读者需要了解 JSP 技术的概述、技术特征和处理过程等;而在 JSP 开发环境的搭建中,读者需要掌握 JDK 及 Tomcat 的安装与配置,以及掌握如何安装和使用 Eclipse 开发工具,并且能够应用它开发一个简单的 JSP 程序;另外,在开发 JSP 程序的两种经典设计模式中,读者需要了解 JSP+Servlet+JavaBean 模式的架构。本章内容在全书中的位置如图 1-1 所示。

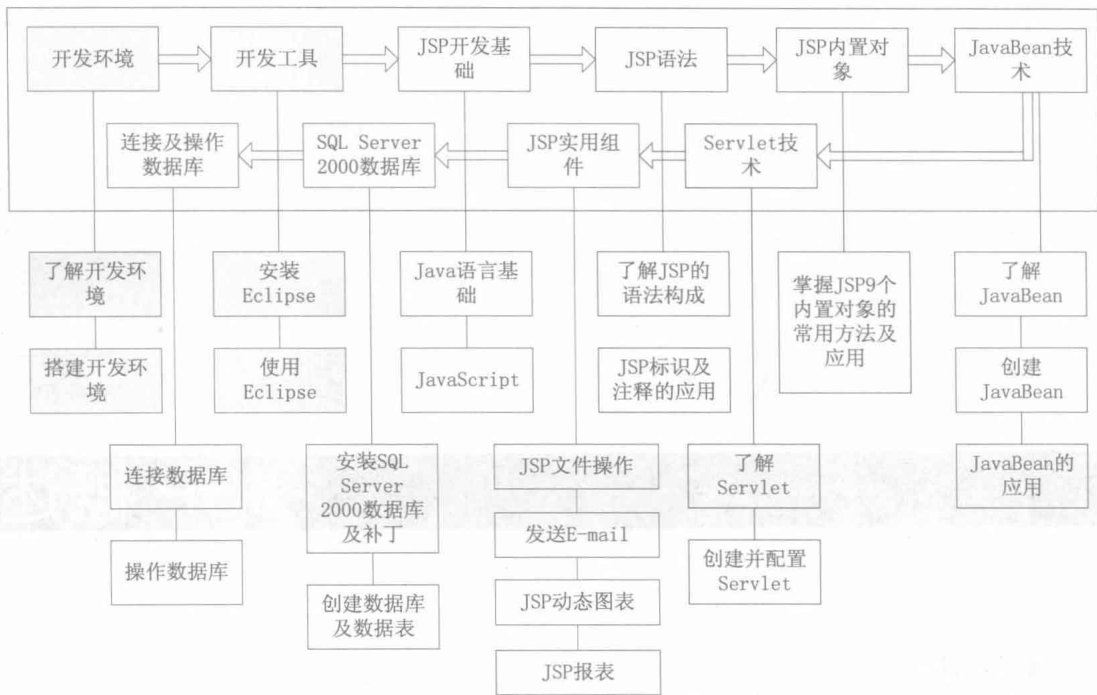


图 1-1 本章内容在全书中的位置

### 【知识框架】

本章内容的知识框架如图 1-2 所示。

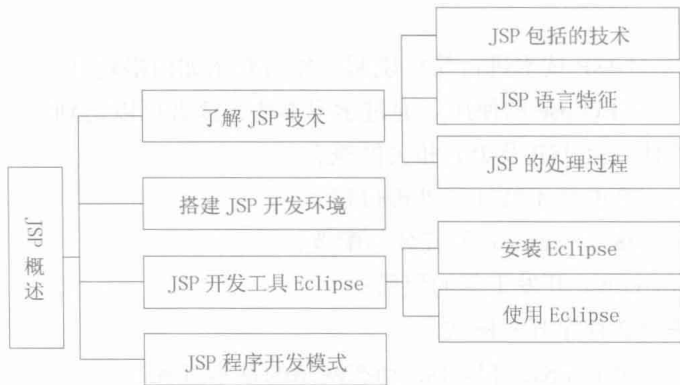


图 1-2 本章内容的知识框架

## 1.1 JSP 技术概述

在了解 JSP 技术之前，先向读者介绍与 JSP 技术相关的一些概念，这样有助于读者阅读后面的内容。

### 1.1.1 Java 语言

Java 语言是由 Sun 公司于 1995 年推出的编程语言，一经推出，就赢得了业界的一致好评，



并受到了广泛关注。Java 语言适用于 Internet 环境, 目前已成为开发 Internet 应用的主要语言之一。它具有简单、面向对象、可移植性、分布性、解释器通用性、稳健、多线程、安全和高性能等优点, 其中最重要的就是实现了跨平台运行, 这使得应用 Java 开发的程序可以方便地移植到不同的操作系统中运行。

Java 语言是完全面向对象的编程语言, 它的语法规则和 C++ 语言的语法规则类似, 但 Java 语言对 C++ 语言进行了简化和提高。例如, C 语言中的指针和多重继承通常会使程序变得复杂, 而 Java 语言通过接口取代了多重继承, 并取消了指针、内存的申请和释放等影响系统安全的部分。

在 Java 语言中, 最小的单位是类, 不允许在类外面定义变量和方法, 所以就不存在所谓的“全局变量”这一概念。在 Java 类中定义的变量和方法分别称为成员变量和成员方法, 其中成员变量也叫做类的属性。在定义这些类的成员时, 需要通过权限修饰符来声明它们的使用范围。

Java 语言编写的程序应被保存为后缀名为 .java 的文件, 然后编译成后缀名为 .class 的字节码文件, 最终通过执行该字节码文件执行 Java 程序。

### 1.1.2 Servlet 技术

Servlet 是在 JSP 之前就存在的运行在服务器端的一种 Java 技术, 它是用 Java 语言编写的服务器端程序, Java 语言能够实现的功能, Servlet 基本上都可以实现 (除图形界面外)。Servlet 主要用于处理 Http 请求, 并将处理的结果传递给浏览器生成动态 Web 页面。Servlet 具有可移植 (可在多种系统平台和服务器平台下运行)、功能强大、安全、可扩展和灵活等优点。

在 JSP 中用到的 Servlet 通常都继承自 javax.servlet.http.HttpServlet 类, 在该类中实现了用来处理 Http 请求的大部分功能。

JSP 是在 Servlet 的基础上开发的一种新的技术, 所以 JSP 与 Servlet 有着密不可分的关系。JSP 页面在执行过程中会被转换为 Servlet, 然后由服务器执行该 Servlet。

关于 Servlet 技术的相关介绍请参看本书的第 6 章。

### 1.1.3 JavaBean 技术

JavaBean 是根据特殊规范编写的普通的 Java 类, 可称它们为“独立的组件”。每一个 JavaBean 实现一个特定的功能, 通过合理地组织具有不同功能的 JavaBean, 可以快速地生成一个全新的应用程序。如果将这个应用程序比做一辆汽车, 那么程序中的 JavaBean 就好比组成这辆汽车的不同零件。对于程序开发人员来说, JavaBean 的最大优点就是充分提高了代码的可重用性, 并且对程序的后期维护和扩展起到了积极的作用。

JavaBean 可按功能划分为可视化和不可可视化两种。可视化 JavaBean 主要应用在图形界面编程的领域中, 在 JSP 中通常应用的是不可可视化 JavaBean, 应用该种 JavaBean 可用来封装各种业务逻辑, 例如连接数据库、获取当前时间等。这样, 当在开发程序的过程中需要连接数据库或实现其他功能时, 就可直接在 JSP 页面或 Servlet 中调用实现该功能的 JavaBean 来实现。

通过应用 JavaBean, 可以很好地将业务逻辑和前台显示代码分离, 这大大提高了代码的可读性和易维护性。

关于 JavaBean 技术的相关介绍请参看本书的第 5 章。



### 1.1.4 JSP 技术

Java Server Pages 缩写为 JSP，是由 Sun 公司倡导，与多个公司共同建立的一种技术标准，它建立在 Servlet 之上。应用 JSP，程序员和非程序员可以高效率地创建 Web 应用程序，并使得开发的 Web 应用程序具有安全性高、跨平台等优点。

JSP 是运行在服务器端的脚本语言之一，与其他的服务器端脚本语言一样，是用来开发动态网页的一种技术。

JSP 页面由传统的 HTML 代码和嵌入到其中的 Java 代码组成。当用户请求一个 JSP 页面时，服务器会执行这些 Java 代码，然后将结果与页面中的静态部分相结合返回给客户端浏览器。JSP 页面中还包含了各种特殊的 JSP 元素，通过这些元素可以访问其他的动态内容并将它们嵌入到页面中，例如访问 JavaBean 组件的<jsp:useBean>动作元素。程序员还可以通过编写自己的元素来实现特定的功能，开发出更为强大的 Web 应用程序。

JSP 是在 Servlet 的基础上开发的技术，它继承了 Java Servlet 的各项优秀功能。而 Java Servlet 是 Java 的一种解决方案，在制作网页的过程中，它继承了 Java 的所有特性。因此，JSP 同样继承了 Java 技术的简单、便利、面向对象、跨平台和安全可靠等优点，比起其他服务器脚本语言，JSP 更加简单、迅速和有力。在 JSP 中利用 JavaBean 和 JSP 元素，可以有效地将静态的 HTML 代码和动态数据区分开来，给程序的修改和扩展带来了很大方便。

## 1.2 JSP 的技术特征

本节将介绍 JSP 在开发 Web 应用程序时的一些特点，如跨平台、分离静态内容和动态内容、可重复使用的组件、沿用了 Java Servlet 的所有功能及预编译等。

### 1. 跨平台

JSP 是以 Java 为基础开发的，所以它不仅可以沿用 Java 强大的 API 功能，而且不管是在何种平台下，只要服务器支持 JSP，就可以运行使用 JSP 开发的 Web 应用程序，体现了它的跨平台、跨服务器的特点。

在数据库操作中，因为 JDBC 同样是独立于平台的，所以在 JSP 中使用 Java API 提供的 JDBC 来连接数据库时，就不用担心平台变更时的代码移植问题。正是因为 Java 的这种特征，使得应用 JSP 开发的 Web 应用程序能够很简单地运用到不同的平台上。

### 2. 分离静态内容和动态内容

在前面提到的 Java Servlet，对于开发 Web 应用程序而言是一种很好的技术。但同时面临着一个问题：所有的内容必须在 Java 代码中来完成，包括 HTML 代码，都要嵌入到程序代码中来生成静态的内容。这使得即使因 HTML 代码出现的小问题，也需要有熟悉 Java Servlet 的程序员来解决。

JSP 弥补了 Java Servlet 在工作中的不足。使用 JSP，程序员可以使用 HTML 或 XML 标记来设计和格式化静态的内容部分，使用 JSP 标记及 JavaBean 组件或者小脚本程序来制作动态内容部分。服务器将执行 JSP 标记和小脚本程序，并将结果与页面中的静态部分结合后以 HTML 页面的形式发



送给客户端浏览器。程序员可以将一些业务逻辑封装到 JavaBean 组件中，Web 页面的设计人员可以利用程序员开发的 JavaBean 组件和 JSP 标记来制作出动态页面，而且不会影响到内容的生成。

### 3. 可重复使用的组件

JavaBean 组件是 JSP 中不可缺少的重要组成部分之一，程序通过 JavaBean 组件来执行所要求的更为复杂的运算。JavaBean 组件不仅可以应用于 JSP 中，同样适用于其他的 Java 应用程序中。这种特性使得开发人员之间可以共享 JavaBean 组件，加快了应用程序的总体开发进程。

同样，JSP 的标准标签和自定义标签与 JavaBean 组件一样可以一次生成重复使用。这些标签都是通过编写的程序代码来实现特定功能的，在使用它们时与通常在页面中用到的 HTML 标记用法相同。这样可以将一个复杂而且需要出现多次的操作简单化，大大提高了工作效率。

### 4. 沿用了 Java Servlet 的所有功能

相对于 Java Servlet 来说，使用从 Java Servlet 发展而来的 JSP 技术开发 Web 应用更加简单易学，并且 JSP 同样提供了 Java Servlet 所有的特性。实际上，服务器在执行 JSP 文件时先将其转换为 Servlet 代码，然后对其进行编译，可以说 JSP 就是 Servlet，创建一个 JSP 文件其实就是创建一个 Servlet 文件的简化操作。理所当然，Servlet 中的所有特性在 JSP 中同样可以使用。

### 5. 预编译

预编译是 JSP 的另一个重要的特性。JSP 页面在被服务器执行前，都是已经被编译好的，并且通常只进行一次编译，即在 JSP 页面被第一次请求时进行编译，在后续的请求中如果 JSP 页面没有被修改过，服务器只需要直接调用这些已经被编译好的代码，这大大提高了访问速度。

## 1.3 JSP 的处理过程

当客户端浏览器向服务器发出请求访问一个 JSP 页面时，服务器根据该请求加载相应的 JSP 页面，并对该页面进行编译，然后执行。JSP 的处理过程如图 1-3 所示。

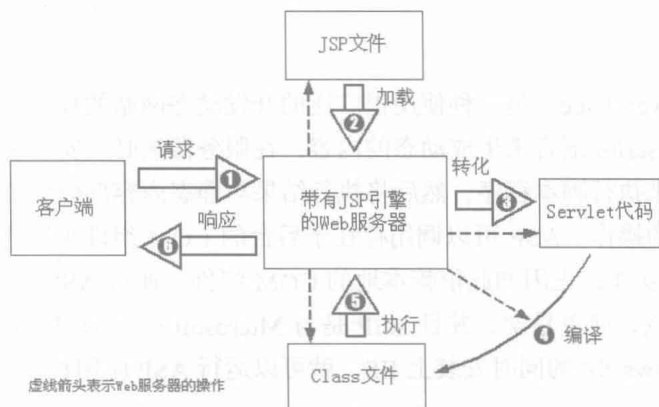


图 1-3 JSP 的处理过程

(1) 客户端通过浏览器向服务器发出请求，在该请求中包含了请求的资源的路径，这样当服务器接收到该请求后就可以知道被请求的资源。



- (2) 服务器根据接收到的客户端的请求来加载被请求的 JSP 文件。
- (3) Web 服务器中的 JSP 引擎会将加载的 JSP 文件转化为 Servlet。
- (4) JSP 引擎将生成的 Servlet 代码编译成 Class 文件。
- (5) 服务器执行这个 Class 文件。

(6) 服务器将执行结果发送给浏览器进行显示。

从上面的介绍中可以看到, JSP 文件被 JSP 引擎进行转换后, 又被编译成了 Class 文件, 最终由服务器通过执行这个 Class 文件来对客户端的请求进行响应。其中第 3 步与第 4 步构成了 JSP 处理过程中的翻译阶段, 而第 5 步为请求处理阶段。

但并不是每次请求都需要重复进行这样的处理。当服务器第一次接收到对某个页面的请求时, JSP 引擎就开始进行上述的处理过程, 将被请求的 JSP 文件编译成 Class 文件。在后续对该页面再次进行请求时, 若页面没有进行任何改动, 服务器直接调用 Class 文件执行即可。所以当某个 JSP 页面第一次被请求时, 会有一些延迟, 而再次访问时会感觉快了很多。如果被请求的页面经过了修改, 那么服务器将会重新编译这个文件, 然后执行。

## 1.4 JSP 与其他服务器端脚本语言的比较

### 1.4.1 CGI

CGI 是最早用来创建动态网页的一种技术, 它可以使浏览器与服务器之间产生互动关系。CGI 的全称为 Common Gateway Interface, 即通用网关接口。它允许使用不同的语言来编写适合的 CGI 程序, 该程序被放在 Web 服务器上运行。当客户端发出请求给服务器时, 服务器根据客户请求建立一个新的进程来执行指定的 CGI 程序, 并将执行结果以网页的类型传输到客户端的浏览器上进行显示。CGI 可以说是当前应用程序的基础技术, 但这种技术编制方式比较困难且效率低下, 因为每次页面被请求时, 都要求服务器重新将 CGI 程序编译成可执行的代码。在 CGI 中使用最为常见的语言为 C/C++、Java 和 Perl (Practical Extraction and Report Language, 实用萃取与报告语言)。

### 1.4.2 ASP

ASP (Active Server Page) 是一种使用很广泛的开发动态网站的技术。它通过在页面代码中嵌入 VBScript 或 JavaScript 语言来生成动态的内容, 在服务器端必须安装适当的解释器后, 可以通过调用此解释器来执行脚本程序, 然后将执行结果与静态内容部分结合并传送到客户端浏览器上。对于一些复杂的操作, ASP 可以调用存在于后台的 COM 组件来完成, 所以说 COM 组件无限地扩充了 ASP 的功能; 正因如此依赖本地的 COM 组件, 使得 ASP 主要用于 Windows 平台中。ASP 具有很多优点, 简单易学, 并且 ASP 是与 Microsoft 公司的 IIS 捆绑在一起的, 在安装 Windows 2000、Windows XP 的同时安装上 IIS, 就可以运行 ASP 应用程序了。

### 1.4.3 PHP

PHP 来自于 Personal Home Page 一词, 但现在的 PHP 已经不再表示名词的缩写, 而是一个开



发动态网页技术的名称。PHP 语法类似于 C，并且混合了 Perl、C++ 和 Java 的一些特性。它是一种开源的 Web 服务器脚本语言，与 ASP 和 JSP 一样可以在页面中加入脚本代码来生成动态内容。对于一些复杂的操作可以封装到函数或类中。在 PHP 中提供了许多已经定义好的函数，例如提供的标准的数据库接口，使得数据库连接方便，扩展性强。PHP 可以被多个平台支持，主要被广泛应用于 UNIX 和 Linux 平台。由于 PHP 本身的代码对外开放，经过许多软件工程师的检测，因此到目前为止该技术具有公认的安全性能。

#### 1.4.4 ASP.NET

ASP.NET 也是一种建立动态 Web 应用程序的技术，它是 .NET 框架的一部分，可以使用任何 .NET 兼容的语言，如 Visual Basic.NET、C# 和 J# 等来编写 ASP.NET 应用程序。这种 ASP.NET 页面（Web Forms）编译后可以提供比脚本语言更出色的性能表现。Web Forms 允许在网页基础上建立强大的窗体。当建立页面时，可以使用 ASP.NET 服务端控件来建立常用的 UI 元素，并对它们编程来完成一般的任务。这些控件允许开发者使用内建可重用的组件和自定义组件来快速建立 Web Forms，使代码简单化。

### 1.5 JSP 开发环境搭建

#### 1.5.1 JSP 的运行环境

使用 JSP 进行开发，需要具备以下对应的运行环境：Web 浏览器、Web 服务器、JDK 开发工具包和数据库。下面分别介绍这些环境。

##### 1. Web 浏览器

浏览器是客户端用户访问 Web 应用的工具，与开发 JSP 应用不存在很大的关系，所以开发 JSP 对浏览器的要求并不是很高，任何支持 HTML 的浏览器都可以。

##### 2. Web 服务器

Web 服务器是运行及发布 Web 应用的大容器，只有将开发的 Web 项目放置到该容器中，才能使网络中的所有用户通过浏览器进行访问。开发 JSP 应用所采用的服务器主要是 Servlet 兼容的 Web 服务器，比较常用的有 BEA WebLogic、IBM WebSphere 和 Apache Tomcat 等。

Weblogic 是 BEA 公司的产品，它又分为 WebLogic Server、WebLogic Enterprise 和 WebLogic Portal 系列，其中 WebLogic Server 的功能特别强大，它支持企业级的、多层次的和完全分布式的 Web 应用，并且服务器的配置简单、界面友好，对于那些正在寻求能够提供 Java 平台所拥有的一切应用服务器的用户来说，WebLogic Server 是一个十分理想的选择，在后面的章节中将对该服务器的安装与配置进行讲解。

Tomcat 服务器最为流行，它是 Apache-Jakarta 开源项目中的一个子项目，是一个小型的、轻量级的、支持 JSP 和 Servlet 技术的 Web 服务器，它已经成为学习开发 JSP 应用的首选，本书中的所有例子都使用了 Tomcat 作为 Web 服务器，所以对该服务器的安装与配置在后面的章节中也





进行了讲解。目前 Tomcat 的最新版本为 apache-tomcat-6.0.18。

### 3. JDK

JDK (Java Develop Kit, Java 开发工具包) 包括运行 Java 程序所必须的 JRE 环境及开发过程中常用的库文件。在使用 JSP 开发网站之前, 首先要安装 JDK, 目前 JDK 的最新版本为 JDK1.6。

### 4. 数据库

任何项目的开发几乎都需要使用数据库, 数据库用来存储项目中需要的信息。根据项目的规模, 应采用合适的数据库。如大型项目可采用 Oracle 数据库, 中型项目可采用 Microsoft SQL Server 或 MySQL 数据库, 小型项目可采用 Microsoft Access 数据库。Microsoft Access 数据库的功能远比不上 Microsoft SQL Server 和 MySQL 强大, 但它具有方便、灵活的特点, 对于一些小型项目来说是比较理想的选择。

## 1.5.2 JDK 的安装与配置

JDK 由 SUN 公司提供, 其中包括运行 Java 程序所必须的 JRE (Java Runtime Environment, Java 运行环境) 及开发过程中常用的库文件。在使用 JSP 开发网站之前, 首先要安装 JDK 组件。

### 1. JDK 的安装

本书中采用的是 JDK1.6 版本, 读者可到 SUN 公司的官方网站进行下载, 网址是: <http://java.sun.com/javase/downloads/index.jsp>。下载后的文件名称为 jdk-6u10-rc2-bin-b32-windows-i586-p-12\_sep\_2008.exe, 双击该文件即可开始安装。具体安装步骤如下。

(1) 双击 jdk-6u10-rc2-bin-b32-windows-i586-p-12\_sep\_2008.exe 文件, 在弹出的对话框中, 单击“接受”按钮, 接受许可证协议。

(2) 此时将弹出“自定义安装”对话框, 如图 1-4 所示, 在该对话框中, 可以设置 JDK 的安装路径并选择安装组件。

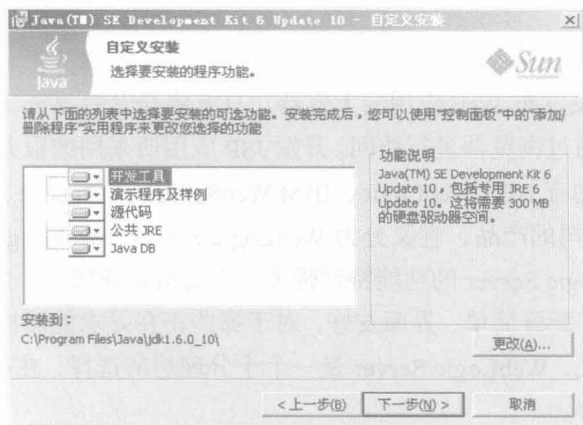


图 1-4 “自定义安装”对话框

(3) 在图 1-4 所示的对话框中单击“更改”按钮, 在打开的“更改当前目标文件夹”对话框中, 更改安装路径为 C:\Java\jdk1.6.0\_10\, 其他采用默认设置, 单击“下一步”按钮, 将打开安



装进度对话框安装 JDK。在安装过程中将打开如图 1-5 所示的设置 JRE 安装路径的对话框。

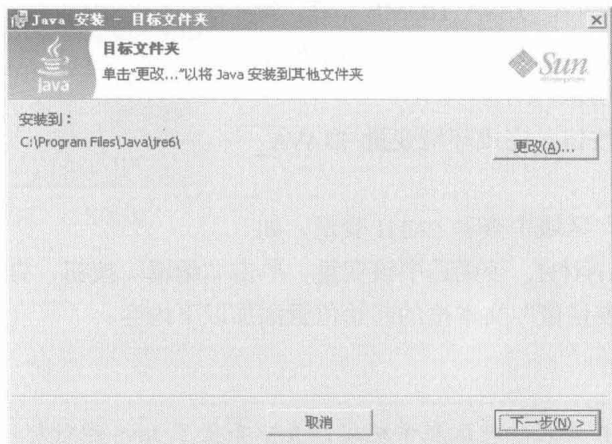


图 1-5 设置 JRE 安装路径的对话框



由于 JDK 只是 Java 程序的开发环境,所以在 JDK 的安装文件中还包含了一个 JRE (J2SE Runtime Environment), 默认情况下同 JDK 一起安装。

(4) 在设置 JRE 安装路径的对话框中,单击“更改”按钮,在打开的对话框中将 JRE 的安装路径修改为 C:\Java\jre6\,单击“下一步”按钮继续安装 JRE。当安装完成时,将打开安装成功的提示对话框,单击“完成”按钮,即可完成 JDK 的安装。

## 2. JDK 的配置与测试

安装完 JDK 后,需要设置环境变量及测试 JDK 配置是否成功,具体步骤如下。

(1) 在“我的电脑”上单击鼠标右键,在弹出的快捷菜单中选择“属性”菜单项,在打开的“系统属性”对话框中选择“高级”选项卡,如图 1-6 所示。

(2) 在“高级”选项卡中,单击“环境变量”按钮,打开“环境变量”对话框,如图 1-7 所示。在其中可以添加针对单个用户的“用户变量”和针对所有用户的“系统变量”。

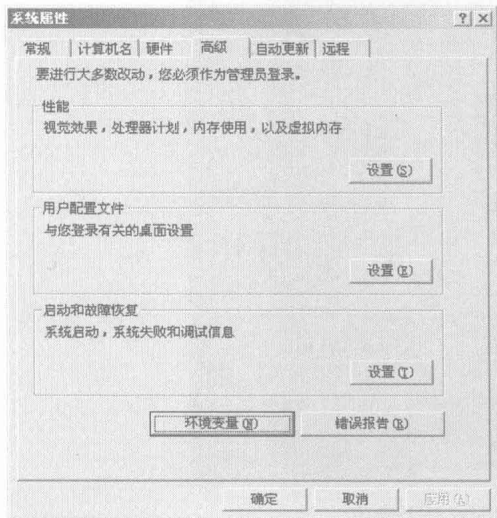


图 1-6 “系统属性”对话框

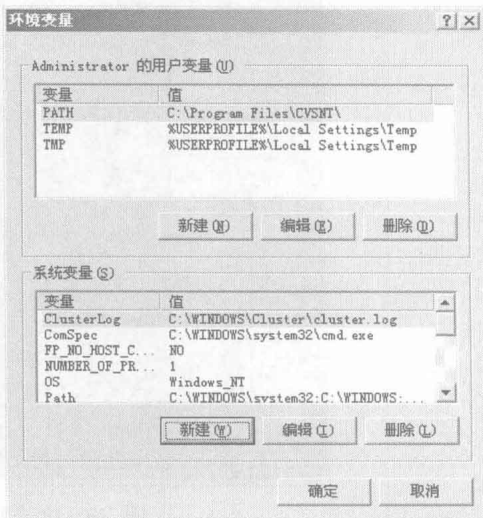


图 1-7 “环境变量”对话框