



全国高等职业教育规划教材

JSP动态网站开发 案例教程

主 编 刘素芳

副主编 刘 芳



电子教案下载网址 www.cmpedu.com



机械工业出版社
CHINA MACHINE PRESS

全国高等职业教育规划教材

JSP 动态网站开发案例教程

主 编 刘素芳
副主编 刘 芳
参 编 刘张榕 郑志娴



机械工业出版社

本书系统地介绍了使用 JSP 技术开发动态网站的相关知识、方法与实现过程。主要内容包括 JSP 概述、JSP 开发环境、JSP 语法、JSP 内置对象、JSP 数据库应用、JavaBean 技术、Servlet 技术、MVC 模式和 Web 人事管理系统设计。

本书根据 Java Web 程序员的岗位能力要求和学生的认知规律精心组织教材内容。全书采用任务驱动方式展开,以程序案例为主导,通过 43 个完整的案例和 1 个 Web 人事管理系统,将知识点融入案例,以案例带动知识点的学习。同时将知识介绍和技能训练有机结合,融“教、学、练”三者于一体,适合“项目驱动、案例教学、理论实践一体化”的教学模式。同时,本书提供了完整的配套教学资源。

本书可作为高职高专院校计算机及相关专业的教材,也可作为 JSP 初学者及网站开发人员的参考用书。

本书免费提供电子课件和案例代码,需要的教师可登录 www.cmpedu.com 免费注册、审核通过后下载,或联系编辑索取(QQ: 1239258369, 电话: 010-88379739)。

图书在版编目(CIP)数据

JSP 动态网站开发案例教程/刘素芳主编. —北京:机械工业出版社,2012. 4

全国高等职业教育规划教材

ISBN 978-7-111-37271-4

I. ①J… II. ①刘… III. ①JAVA 语言-网页制作工具-高等职业教育-教材 IV. ①TP312②TP393.092

中国版本图书馆 CIP 数据核字(2012)第 013891 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:鹿 征

责任印制:李 妍

中国农业出版社印刷厂印刷

2012 年 4 月第 1 版·第 1 次印刷

184mm×260mm·15.75 印张·385 千字

0001-3000 册

标准书号:ISBN 978-7-111-37271-4

定价:31.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心:(010)88361066

门户网:<http://www.cmpbook.com>

销售一部:(010)68326294

教材网:<http://www.cmpedu.com>

销售二部:(010)88379649

读者购书热线:(010)88379203

封面无防伪标均为盗版

前 言

JSP (Java Server Pages) 是基于 Java Servlet 以及整个 Java 体系的 Web 开发技术, 利用这一技术可以建立先进的、安全的和跨平台的动态网站。JSP 以 Java 技术为基础, 又在许多方面做了改进, 具有动态页面与静态页面分离、能够脱离硬件平台的束缚及编译后运行等优点。JSP 已经成为开发动态网站的主流技术之一。

本书主要有以下特点:

1. 合理的知识结构

本书的定位是读者在具备 Java 语言的基础上使用 JSP 技术进行 Web 开发, 重点介绍 JSP 开发的主要技能。本书以应用为主, 融合大量实例介绍理论知识, 以具体应用带动知识点的学习, 由浅入深、循序渐进地讲解了如何利用 JSP 技术创建灵活、安全和健壮的 Web 站点。

2. 完整的案例教学

本书围绕实用项目, 经过精心设计将项目分解为多个既独立又具有一定联系的教学案例 (43 个), 每个案例的讲解都按照“案例学习目标”→“案例知识要点”→“案例完成步骤”这些环节详细展开, 体现了“在做中学、学以致用”的教学理念。

读者学习案例的过程, 就是学习 JSP 知识和技能的过程, 达到“知识案例的一体化”; 读者在完成案例的同时, 也就完成了项目的某一功能, 达到“案例项目的一体化”。

3. 习题和实训

本书各章都有习题和实训, 习题用于复习本章理论知识, 实训用于培养读者的实践技能。

本书共 9 章, 内容由浅入深, 涵盖了 JSP 的各个主要知识点。具体安排如下所示。

第 1 章为 JSP 概述, 介绍了 JSP 基本概念、常用 Web 开发技术和 JSP 开发模式。

第 2 章介绍了 JSP 开发环境, 具体包括 JDK 的安裝配置、Tomcat 的安裝配置、JSP 开发工具、创建第 1 个 JSP 程序和 JSP 页面执行流程。

第 3 章介绍了 JSP 语法, 详细介绍了 JSP 页面结构、JSP 页面中的注释、JSP 脚本元素、JSP 指令标记和 JSP 动作标记。

第 4 章介绍了 JSP 内置对象, 具体包括 request 对象、out 对象、response 对象、session 对象、application 对象、pageContext 对象、config 对象、page 对象和 exception 对象。

第 5 章为 JSP 数据库应用, 介绍了 JDBC 技术、JDBC 连接方式、JDBC 常用应用程序接口和应用 JDBC 实现对数据库记录的增加、删除、修改和查询操作。

第 6 章介绍了 JavaBean 技术, 具体包括 JavaBean 基础、编写 JavaBean、在 JSP 中应用 JavaBean、JavaBean 与 HTML 表单的交互以及 JavaBean 的典型应用。

第 7 章介绍了 Servlet 技术, 具体包括 Servlet 的基本概念、编写和配置 Servlet、调用 Servlet、Servlet 的生命周期和 Servlet 的典型应用。

第 8 章介绍了 MVC 模式, 具体包括 MVC 模式概念、JSP 中的 MVC 模式、模型的生命周期与视图更新和 MVC 模式的典型应用。

第9章通过开发Web人事管理系统,介绍了Web人事管理的系统功能分析、系统功能模块设计、数据库需求分析、数据库概念结构设计、数据库逻辑结构设计和系统功能模块开发。

本书纳入“福建省高等职业教育教材建设计划”,在编写过程中得到了福建省教育厅的大力支持,在此表示忠心感谢!

本书由刘素芳任主编,刘芳任副主编,刘张榕、郑志娴参与了部分章节的编写工作。

由于编者水平有限,书中难免存在错误和疏漏之处,欢迎广大读者和同仁提出宝贵意见和建议。本书配有完整课件和案例代码,如果需要,请登录机械工业出版社教材服务网 <http://www.cmpedu.com> 下载。

编 者

目 录

出版说明

前言

第1章 JSP 概述 1

1.1 JSP 简介 1

1.1.1 JSP 的基本概念 1

1.1.2 JSP 的优势 1

1.2 JSP 与其他 Web 开发技术的
比较 2

1.2.1 ASP/ASP.NET 技术 2

1.2.2 PHP 技术 2

1.2.3 各种 Web 技术性能比较 3

1.3 JSP 开发模式 3

1.4 习题 5

第2章 JSP 开发环境 6

2.1 JSP 运行环境简介 6

2.2 JDK 的安装与配置 6

2.3 Tomcat 的安装与配置 9

2.4 JSP 开发工具 12

2.4.1 Eclipse 的安装与配置 12

2.4.2 案例1 在 Eclipse 中创建 JSP
程序 15

2.4.3 案例2 独立创建 JSP 程序 19

2.5 JSP 页面执行流程 21

2.6 习题 21

2.7 实训 22

第3章 JSP 语法 23

3.1 JSP 页面的基本结构 23

3.1.1 JSP 页面的组成元素 23

3.1.2 案例1 包含5种元素的 JSP
页面 24

3.2 JSP 页面中的注释 25

3.2.1 HTML 注释 25

3.2.2 JSP 注释 25

3.2.3 案例2 使用注释的 JSP 页面 25

3.3 脚本元素 26

3.3.1 声明 27

3.3.2 表达式 27

3.3.3 脚本小程序 27

3.3.4 案例3 使用声明、表达式和脚本
小程序的 JSP 页面 28

3.4 JSP 指令标记 29

3.4.1 page 指令 29

3.4.2 include 指令 30

3.4.3 taglib 指令 30

3.4.4 案例4 使用 page、include 指令的
JSP 页面 30

3.5 JSP 动作标记 31

3.5.1 include 动作 31

3.5.2 param 动作 32

3.5.3 forward 动作 32

3.5.4 plugin 动作 32

3.5.5 useBean 动作 33

3.5.6 案例5 使用 include、param
动作的 JSP 页面 33

3.5.7 案例6 使用 forward、param
动作的 JSP 页面 34

3.6 习题 35

3.7 实训 36

第4章 JSP 内置对象 42

4.1 JSP 内置对象概述 42

4.2 request 对象 43

4.2.1 request 对象的常用方法 43

4.2.2 案例1 使用 request 对象获取客
户端请求的 HTTP 头信息并
显示 43

4.2.3 案例2 使用 request 对象获取
表单信息、处理汉字 45

4.3 out 对象 47

4.3.1	out 对象的常用方法	47	5.3	JDBC 数据库操作实例	79
4.3.2	案例 3 使用 out 对象输出各种 类型数据到客户端	48	5.3.1	案例 1 使用 JDBC - ODBC 桥驱动 连接数据库	79
4.4	response 对象	49	5.3.2	案例 2 使用本地协议纯 Java 驱动 程序连接数据库	84
4.4.1	response 对象的常用方法	50	5.3.3	案例 3 查询员工档案信息	88
4.4.2	案例 4 使用 response 对象自动 刷新页面	50	5.3.4	案例 4 删除员工档案信息	90
4.4.3	案例 5 使用 response 对象重定向 页面	51	5.3.5	案例 5 修改员工档案信息	94
4.4.4	案例 6 使用 response 对象动态响应 contentType 属性	52	5.3.6	案例 6 新增员工档案信息 ...	100
4.5	session 对象	54	5.3.7	案例 7 分页显示员工档案 信息	105
4.5.1	session 对象的常用方法	54	5.3.8	案例 8 连接 Access 数据库 ...	108
4.5.2	案例 7 使用 session 制作站点 计数器	55	5.3.9	案例 9 连接 MySQL 数据库 ...	110
4.5.3	案例 8 制作登录模块	56	5.4	习题	112
4.6	application 对象	59	5.5	实训	113
4.6.1	application 对象的常用方法	59	第 6 章	JavaBean 技术	116
4.6.2	案例 9 使用 application 制作站点 计数器	60	6.1	JavaBean 简介	116
4.6.3	案例 10 制作简单的聊天室	61	6.1.1	JavaBean 的概念	116
4.7	其他对象	65	6.1.2	JavaBean 的优势	116
4.7.1	pageContext 对象	65	6.1.3	JavaBean 分类	117
4.7.2	config 对象	65	6.1.4	JavaBean 规范	117
4.7.3	page 对象	65	6.2	JavaBean 开发	117
4.7.4	exception 对象	66	6.2.1	创建 JavaBean	117
4.7.5	案例 11 使用 exception 对象获取异常 信息	66	6.2.2	案例 1 编写一个简单的 JavaBean	117
4.8	习题	67	6.3	JavaBean 使用	119
4.9	实训	68	6.3.1	<jsp: useBean > 动作标记	119
第 5 章	JSP 数据库应用	73	6.3.2	<jsp: setProperty > 动作 标记	119
5.1	JDBC 技术概述	73	6.3.3	<jsp: getProperty > 动作 标记	121
5.1.1	JDBC 的技术简介	73	6.3.4	案例 2 使用 JavaBean	121
5.1.2	JDBC 的连接方式	74	6.3.5	案例 3 使用 JavaBean 与 HTML 表单 交互	123
5.2	JDBC 应用程序接口	74	6.4	JavaBean 在 JSP 中的典型 应用	126
5.2.1	驱动程序管理器 Driver- Manager 类	74	6.4.1	案例 4 使用 JavaBean 封装数据库 访问操作	126
5.2.2	数据库连接接口 Connection	75	6.4.2	案例 5 使用 JavaBean 封装编码 转换操作	130
5.2.3	SQL 声明类 Statement	76			
5.2.4	数据结果集接口 ResultSet	78			

6.5 习题	131	第8章 MVC 模式	168
6.6 实训	132	8.1 MVC 模式简介	168
第7章 Servlet 技术	135	8.2 JSP 中的 MVC 模式	168
7.1 Servlet 简介	135	8.3 模型的声明周期与视图更新 ...	169
7.1.1 Servlet 的概念	135	8.3.1 request 周期的 JavaBean	169
7.1.2 Servlet 的特点	136	8.3.2 session 周期的 JavaBean	170
7.1.3 Servlet 的优势	136	8.3.3 application 周期的 JavaBean	171
7.1.4 Servlet 与 JSP 的区别	137	8.4 MVC 模式应用实例	172
7.1.5 案例1 编写一个简单的 Servlet	137	8.4.1 案例1 MVC 模式计算三角形 面积	172
7.2 Servlet 技术原理	138	8.4.2 案例2 MVC 模式操作数据库完成 会员注册	175
7.2.1 Servlet 的常用类与接口	138	8.5 习题	179
7.2.2 Servlet 的程序结构	142	8.6 实训	180
7.2.3 Servlet 的生命周期	142	第9章 Web 人事管理系统	181
7.3 Servlet 开发	143	9.1 系统功能分析	181
7.3.1 创建 Servlet	143	9.1.1 系统功能分析	181
7.3.2 案例2 配置、调用 Servlet ...	144	9.1.2 系统功能模块设计	182
7.3.3 案例3 应用 Servlet 获取指定 HTML 表单数据	145	9.2 数据库模型	182
7.4 Servlet 在 JSP 中的典型应用 ...	147	9.2.1 数据库需求分析	182
7.4.1 案例4 编写生成4位随机验证码 的 Servlet	147	9.2.2 数据库概念结构设计	183
7.4.2 案例5 在 Servlet 中实现页面 转发	152	9.2.3 数据库逻辑结构设计	184
7.4.3 案例6 应用 Servlet 获取所有 HTML 表单数据	157	9.3 系统功能模块实现	186
7.4.4 案例7 应用 Servlet 获取 Cookie 数据	159	9.3.1 公用组件及管理首页开发	186
7.4.5 案例8 应用 Servlet 获取 Session 数据	162	9.3.2 用户管理模块开发	189
7.5 习题	164	9.3.3 档案管理模块开发	193
7.6 实训	165	9.3.4 培训管理模块开发	214
		9.3.5 职称管理模块	227
		9.3.6 奖惩管理及调动管理模块	241
		参考文献	242

第 1 章 JSP 概述

学习目标

通过本章节的学习，读者可以达到以下学习目标：

- 了解并掌握 JSP 基本概念、常用 Web 技术的优缺点。
- 掌握 JSP 的开发模式。

教学要求

知识要点	能力要求	相关知识
JSP 简介	理解并掌握	JSP 基本概念及优势
JSP 与其他 Web 开发技术的比较	掌握	常用 Web 开发技术：PHP、ASP/ASP.NET、JSP 等
JSP 的开发模式	掌握	JSP 开发模式介绍

1.1 JSP 简介

1.1.1 JSP 的基本概念

JSP (Java Server Pages) 是基于 Java Servlet 以及整个 Java 体系的 Web 开发技术。利用这一技术可以建立安全的、跨平台的 Web 应用程序。JSP 的安全性和跨平台性得益于 Java 语言，这是因为 Java 语言具有不依赖于平台、面向对象和安全等优良特性，已经成为网络程序设计的佼佼者。许多和 Java 有关的技术都得到了广泛的应用和认可，JSP 技术就是其中之一。JSP 以 Java 技术为基础，又在许多方面做了改进，具有动态页面与静态页面分离、能够脱离硬件平台的束缚及编译后运行等优点。JSP 已经成为开发动态网站的主流技术之一。

1.1.2 JSP 的优势

JSP 基于强大的 Java 语言，具有良好的伸缩性，且在跨平台、执行速度等特性上具有很大的技术优势，主要体现在以下几个方面。

(1) 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 标识来设计和格式化显示页面，同时使用 JSP 标记或 JSP 脚本小程序来生成页面上的动态内容。生成动态内容的程序代码被封装在 JavaBean 组件或 JSP 标记中，且所有的 JavaBean 组件或脚本程序都运行于服务器端。由于核心逻辑被封装在 JSP 标记或 JavaBean 组件中，Web 管理员或页面设计者，能够方便地编辑和使用 JSP 页面，而不影响内容的生成。

(2) 可重用性

JSP 中可重用、跨平台的组件 (JavaBean 或 Enterprise JavaBean 组件) 可用于执行应用程序所要求的多次使用的或是更为复杂的处理。例如开发人员可以将多次使用到的对数据库的操作封装到组件中, 在开发过程中可以多次使用这些组件来提高代码的可重用性。

(3) 跨平台

几乎所有的平台都支持 Java、JSP + JavaBean。从一个平台移植到另一个平台, JSP 和 JavaBean 甚至不需要重新编译, 因为 Java 字节码都是标准的字节码, 与平台无关。JSP 完全继承了 Java 编程语言“一次编写、随处运行”的特性。

(4) 执行效率高

当 JSP 页面被第一次请求执行时, JSP 引擎首先将 JSP 页面文件转译成一个 Java 文件, 再将这个 Java 文件编译生成字节码文件, 然后通过执行字节码文件响应用户的请求, 而该 JSP 页面再次被请求执行时, JSP 引擎将直接执行这个字节码文件来响应用户, 因此执行效率高。

1.2 JSP 与其他 Web 开发技术的比较

目前流行的动态网页开发技术主要有以下几种: ASP、ASP.NET、PHP 和 JSP 等, 它们各有优缺点。

1.2.1 ASP/ASP.NET 技术

ASP (Active Server Pages) 是 Microsoft 公司早期推出的动态网页开发技术。它通过在页面代码中嵌入 VBScript、JavaScript 或 JScript 脚本语言, 生成动态、交互且高效率的 Web 应用程序。

动态网页开发初期, ASP 技术由于其简单易学、安装方便、功能实用及效率高等优点迅速占领了广大市场, 应用非常普遍。

但是随着技术的发展, ASP 逐步显示出其局限性和缺陷, Microsoft 公司也已经不再对 ASP 提供技术支持和版本更新。

ASP.NET 是 Microsoft 公司继 ASP 之后推出的动态网页开发技术, 它是 .NET 框架的一部分。ASP.NET 可以使用任何与 .NET 兼容的语言, 如 Visual Basic.NET、C# 和 JScript.NET 等来编写 ASP.NET 应用程序。ASP.NET 页面编译后其性能比脚本语言更为出色, 并且任何 ASP.NET 应用程序都可以使用整个 .NET 框架先进而强大的功能, 极大地简化了开发人员的工作量, 使得 Web 应用程序开发变得更加方便、快捷, 而且其程序的功能也更加强大。

1.2.2 PHP 技术

PHP 的全称最早为 Personal Home Page, 是 Rasmus Lerdorf 于 1994 年为了维护个人网站, 而用 C 语言开发的一些 CGI 工具程序集, 用来取代其网站原先使用的 Perl 程序。之后发展为一个开放源码的动态网页语言。英文全称也变为 Hypertext Preprocessor, 即超级文本预处理语言。

PHP 是跨平台的、服务器端的嵌入式脚本语言，其语法类似于 C，并且混合了 C++、Java 语言和 Perl 语言的一些特性。借助 PHP，开发人员能够快速编写网页动态代码。PHP 中提供了许多已经定义好的函数，开发人员也可以把一些复杂的操作封装到函数或类中。PHP 可以被多个平台支持，主要应用于 UNIX 和 Linux 平台，而且 PHP 是完全免费的，可以从 PHP 官方网站（<http://www.php.net>）自由下载，不受限制地获得源码。由于 PHP 的对外开放性，其代码经过许多开发人员的检测，因此具有良好的安全性。

但是 PHP 也具有较明显的缺点，例如，PHP 不支持面向对象编程、接口不统一、缺乏正规的商业支持，不支持多层分布结构和分布式计算等。

1.2.3 各种 Web 技术性能比较

上述的 ASP.NET 技术、PHP 技术、JSP 技术是目前市场上主流的 3 种动态网站开发技术。

总体来讲，这 3 种技术的核心工作原理基本上都是一致的，符合 Web 程序的工作过程，即用户在客户端发出 Web 请求，服务器接收用户请求，并根据用户的请求做出响应，之后再将服务器的运行结果反馈给客户端。

3 种技术各有特点和优势。ASP.NET 简单易学，但是不可以跨平台；PHP 技术以及 JSP 技术可以自由跨平台，而且页面执行效率高，但是语法整体较为复杂。下面从 Web 服务器支持、运行平台支持、开发难易程度、运行速度、可扩展性和安全性几个方面来进行比较，见表 1-1。

表 1-1 各种 Web 技术性能比较

比较参数	JSP	ASP.NET	PHP
Web 服务器支持	Tomcat、Resin 等大多数通用的 Web 服务器	Microsoft IIS	Apache
运行平台支持	绝大部分平台	Windows	Windows/Linux/UNIX
开发难易程度	容易	较容易	简单
运行速度	快	较快	较快
可扩展性	好	较好	较差
安全性	好	好	好

1.3 JSP 开发模式

使用 JSP 技术开发动态网站可以使用多种开发模式，如完全使用 JSP 进行开发、使用 JSP + JavaBean 进行开发、使用 MVC 模式进行开发、使用 Struts 等框架进行开发、使用 J2EE 进行开发等。下面简要介绍完全使用 JSP、JSP + JavaBean 及 MVC 开发模式。

1. 完全使用 JSP

完全使用 JSP 开发动态网站时，所有的程序代码都在 JSP 页面中。即 JSP 页面包含所有

代码，包括 HTML 标记、CSS 标记、JavaScript 标记、JSP 标签、JSP 脚本程序段等。用户通过 JSP 页面提交客户端请求，JSP 页面接受请求，经过处理后将处理结果仍然发送到 JSP 页面中。其执行过程如图 1-1 所示。

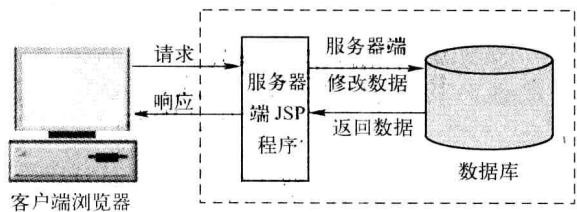


图 1-1 完全使用 JSP 的执行过程

这种开发模式简单易学，适合初学者。但是由于所有代码都在 JSP 页面中，降低了程序的可读性，如果出现错误，也不容易查找和调试，增加了维护难度。不建议采用完全使用 JSP 模式进行动态网站开发。

2. JSP + JavaBean

使用 JSP + JavaBean 开发程序时，将部分可以重复利用的程序代码抽取出来封装到 JavaBean 中，当用户提交请求时，JSP 页面响应请求，同时调用 JavaBean 负责相关数据存取、逻辑运算等处理，最后将结果回传到 JSP 页面返回给用户。JSP + JavaBean 开发模式实现了页面的显示和页面商业逻辑相分离。其执行过程如图 1-2 所示。

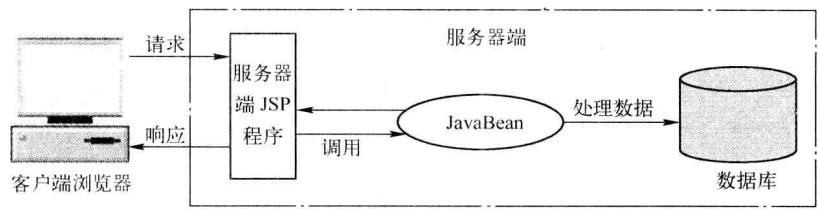


图 1-2 JSP + JavaBean 开发模式的执行过程

这种开发模式会导致页面里嵌入大量的 Java 代码，当需要处理的商业逻辑很复杂时，该模式会使页面内包含大量的 Java 代码，不利于系统后期维护。因此该模式不适合大型系统的开发，但是可以较好地满足中小型 Web 应用的需要。

3. MVC

MVC (Model-View-Controller, 模型 - 视图 - 控制器) 是 Xerox PARC 在 20 世纪 80 年代为编程语言 Smalltalk - 80 发明的一种软件设计模式，至今被广泛使用。最近几年被推荐为 J2EE 平台的设计模式。

MVC 应用程序被分为 3 个核心部件：模型、视图、控制器。其中模型 (Model) 用于存储数据对象；视图 (View) 用于显示数据；控制器 (Controller) 用于处理业务逻辑。

而在 JSP 技术中，JSP 页面用于数据的显示，即适合作为用户的视图；Servlet 用于数据的处理，即适合作为控制器；JavaBean 用于数据的存储，即适合作为模型。JSP + JavaBean + Servlet 技术组合很好地实现了 MVC 模式。其执行过程如图 1-3 所示。

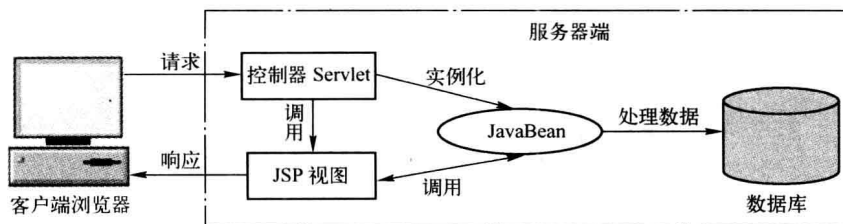


图 1-3 MVC 开发模式的执行过程

1.4 习题

- (1) 目前的动态网页技术主要包括哪些？
- (2) JSP 的全称是什么？JSP 技术有什么优点？

第 2 章 JSP 开发环境

学习目标

通过本章节的学习，读者可以达到以下学习目标：

- 了解 JSP 开发环境。
- 掌握 JDK 及 Tomcat 的安装与配置。
- 掌握 Eclipse 开发工具的使用。
- 掌握通过 Eclipse 开发 JSP 动态网站的基本过程。
- 掌握在 Tomcat 服务器环境下编写、调试和运行 JSP 程序。

教学要求

知识要点	能力要求	相关知识
JSP 运行环境简介	了解	JSP 运行环境
JDK 的安装和配置	掌握	JDK 的安装和配置
Tomcat 的安装与配置	掌握	Tomcat 的安装与配置
JSP 开发工具	掌握	Eclipse 的安装与配置、在 Eclipse 中创建 JSP、独立创建 JSP
JSP 页面执行流程	理解并掌握	JSP 页面执行流程

2.1 JSP 运行环境简介

编写 JSP 程序前，需要先搭建 JSP 的运行环境。即在计算机上安装 JDK，并设置相关的环境变量；在计算机上安装 JSP 引擎（或理解为 Web 服务器），如 Tomcat、Resin 和 J2EE 服务器等。

基于 Java 的 Web 应用系统的开发和运行环境包括客户端环境、服务器端环境。客户端运行环境只需要浏览器，如 IE 浏览器等。服务器端运行环境的搭建根据 Web 服务器的不同应该选择不同的方案。本书案例的开发环境选择 JDK + Tomcat。Tomcat 既可作为 JSP 引擎，又可作为 Web 服务器，配置比较简单，适合初学者。

2.2 JDK 的安装与配置

1. 下载 JDK

JDK 可以到 Oracle 公司网站上提供的下载地址免费下载最新版本，本书案例下载页面如图 2-1 所示，下载 JavaSE 6 版本。

2. 安装 JDK

下载完毕后，直接运行所下载的 jdk - 6 - windows - i586. exe，并按照提示进行安装，安

装过程中可以更改默认的安装路径（如 D:\java\jdk1.6.0）。

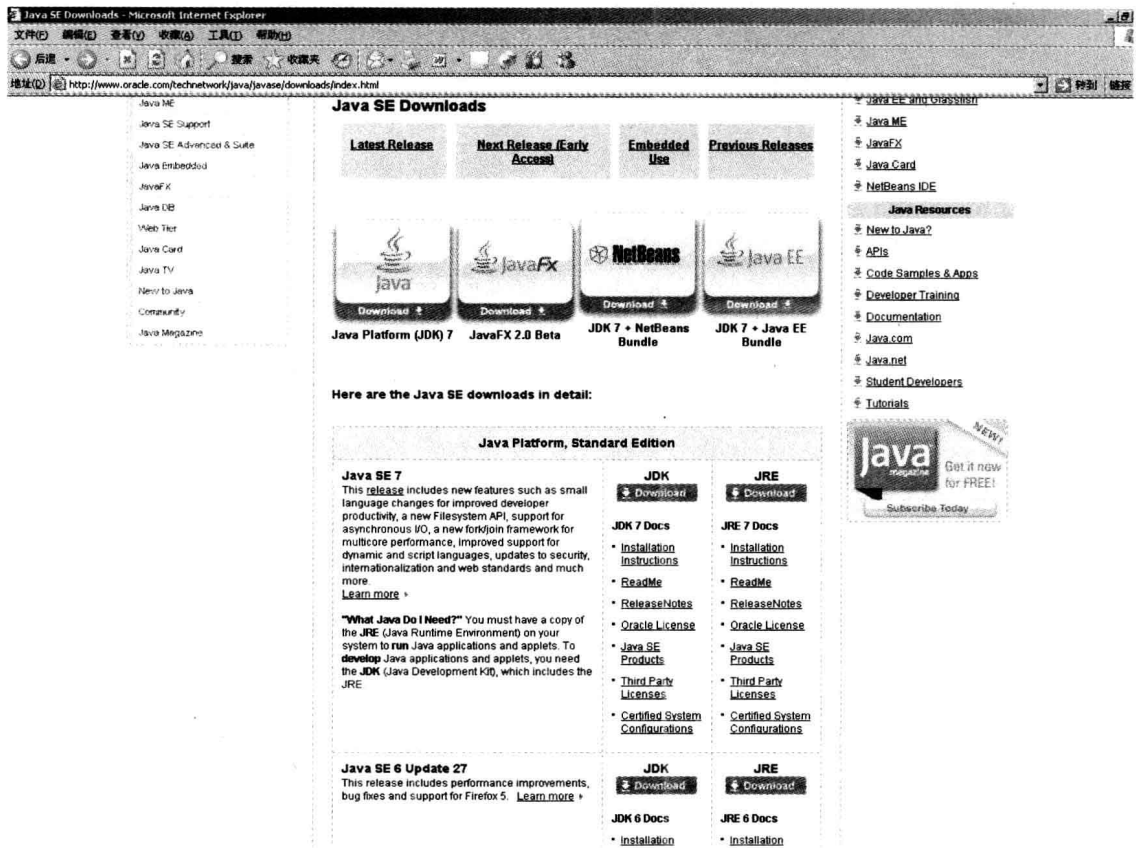


图 2-1 Java SE 6 下载界面

3. 配置环境变量

安装完 JDK 后，需要设置环境变量，具体步骤如下所示。

1) 选中操作系统中“我的电脑”选项后，单击鼠标右键，在弹出的快捷菜单中选择“属性”命令，之后在弹出的对话框中选择“高级”选项卡，如图 2-2 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 2-3 所示。

3) 单击“系统变量”区域中的“新建”按钮，打开“新建系统变量”对话框，在该对话框的“变量名”文本框中输入“JAVA_HOME”，在“变量值”文本框中输入 JDK 的安装路径“D:\java\jdk1.6.0”，如图 2-4 所示。之后再单击“确定”按钮，完成环境变量“JAVA_HOME”的配置。

4) 单击“系统变量”区域中的“新建”按钮，打开“新建系统变量”对话框，在该对话框的“变量名”文本框中输入“PATH”，在“变量值”文本框中输入“D:\java\jdk1.6.0\bin”或“%JAVA_HOME%\bin”，如图 2-5 所示。之后再单击“确定”按钮，完成环境变量“PATH”的配置。

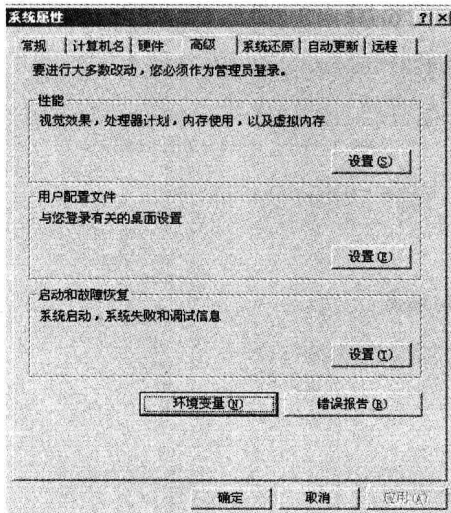


图 2-2 “系统属性”对话框

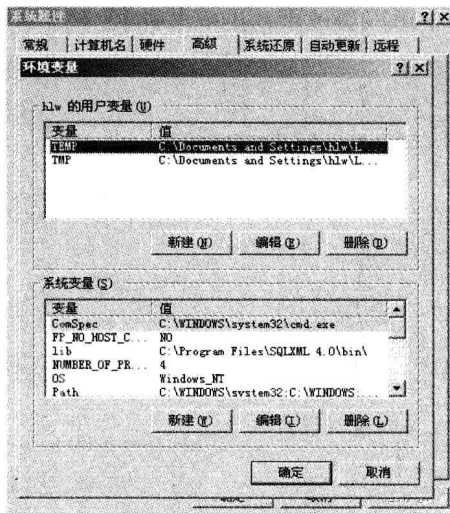


图 2-3 “环境变量”对话框

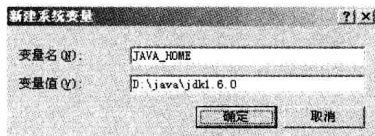


图 2-4 设置 JAVA_HOME

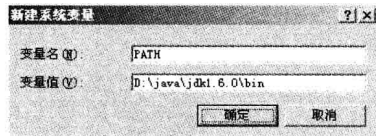


图 2-5 设置 PATH

5) 单击“系统变量”区域中的“新建”按钮，打开“新建系统变量”对话框，在该对话框的“变量名”文本框中输入“CLASSPATH”，在“变量值”文本框中输入“.; D:\java\jdk1.6.0\lib\dt.jar; D:\java\jdk1.6.0\lib\tools.jar; D:\java\jdk1.6.0\jre\lib\rt.jar;”如图 2-6 所示。之后单击“确定”按钮，完成环境变量“CLASSPATH”的配置。

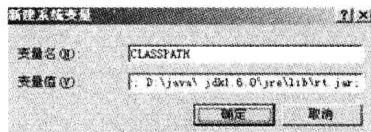


图 2-6 设置 CLASSPATH

说明：如果系统曾经设置过环境变量 JAVA_HOME、PATH、CLASSPATH，就可单击该变量进行编辑操作，将需要的值加入即可，也可以不设置 JAVA_HOME 环境变量。

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 HelloWorld 程序，对其进行编译、执行，以确定 JDK 环境基本正确。

打开记事本程序，输入如下代码：

```
import java.io.*;

public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

将程序保存为 HelloWorld.java。打开命令提示符窗口，进入到 HelloWorld.java 所在目

录，之后输入如下命令：

```
javac HelloWorld.java
java HelloWorld
```

如果该窗口显示“Hello World”字符串，表示 JDK 配置成功。

2.3 Tomcat 的安装与配置

1. 下载 Tomcat

Tomcat 服务器是当今使用最广泛的 Servlet/JSP 服务器，它运行稳定、性能可靠，是学习 JSP 技术和中小型企业应用的最佳选择。

1) 访问 Tomcat 的官方网站地址，可看到各个版本的下载信息，如图 2-7 所示。

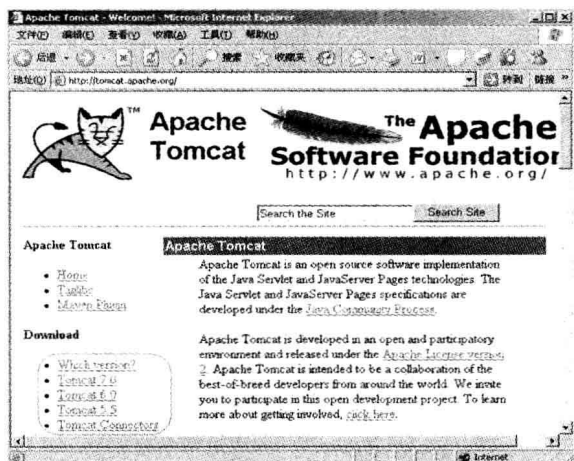


图 2-7 Tomcat 主页

2) 选择要下载的版本，本书案例下载版本 Tomcat 6.0，进入如图 2-8 所示的下载页面。下载 Windows 环境下的程序包 apache - tomcat - 6.0.18.exe。

2. 安装 Tomcat

apache - tomcat - 6.0.18.exe 程序包下载完成后，双击运行 apache - tomcat - 6.0.18.exe 即可进行安装，具体步骤如下。

1) 双击 apache - tomcat - 6.0.18.exe 文件，打开安装向导对话框，如图 2-9 所示，单击“Next”按钮。

2) 在弹出的“License Agreement”（协议许可）对话框中，如图 2-10 所示，单击“I Agree”按钮，接受许可协议，继续安装。

3) 在弹出的“Choose Components”（选择安装组件）对话框中，选择需要安装的组件，通常保留其默认选项，如图 2-11 所示。单击“Next”按钮，继续安装。

4) 在弹出的“Choose Install Location”（选择安装位置）对话框中，在“Destination Folder”处直接输入安装路径，或单击“Browse”按钮来选择安装路径，如图 2-12 所示。之后单击“Next”按钮，继续安装。