



免费提供实例源代码

信息科学与技术丛书

程序设计系列

JSP 2.0

动态网站开发

实例指南

殷庆轩 左昉 王佩楷 等编著

- ◎ 访问数据库
- ◎ Java Beans 组件
- ◎ Java Servlet
- ◎ 网上商店
- ◎ 博客网站
- ◎ 酒店住宿预定系统



机械工业出版社
CHINA MACHINE PRESS

信息科学与技术丛书

程序设计系列

JSP 2.0 动态网站开发实例指南

殷庆轩 左昉 王佩楷 等编著



机械工业出版社

JSP 是一种服务器端脚本语言, 是由 Sun 公司在其强大的 Java 语言的基础上开发出来的。与其他脚本语言 (如 CGI、ASP) 相比, JSP 依托于 Java 语言, 继承并发展了 Java 语言的很多优点, 例如优秀的跨平台性、易于掌握、运行稳定、安全性好等。

本书面向广大计算机爱好者, 特别是对网页制作、编程开发感兴趣的中、高级读者, 由浅入深, 遵照 JSP 2.0 的规范, 介绍如何利用 JSP 语言进行交互式动态网站的开发。

本书分为 13 章, 主要内容有: JSP 简介、HTML 表单、Java 基础、JSP 基本语法、JSP 内置对象、表达式语言、JSP 的文件操作、访问数据库、JavaBeans 组件、Java Servlet、网上商店、博客网站和酒店住宿预定系统。

本书文字浅显易懂, 内容充实。书中所列举的大量例子既可以帮助读者学习掌握知识, 又是制作动态网页和网站的优秀模板——读者只要修改少量的代码, 就可以制作出具有专业水准的动态网页和网站。

图书在版编目 (CIP) 数据

JSP 2.0 动态网站开发实例指南 / 殷庆轩等编著. —北京: 机械工业出版社, 2006.7

(信息科学与技术丛书. 程序设计系列)

ISBN 7-111-19598-1

I. J... II. 殷... III. JAVA 语言—主页制作—程序设计—指南
IV. TP393.092-62

中国版本图书馆 CIP 数据核字 (2006) 第 080557 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策 划: 胡毓坚

责任编辑: 罗子超

责任印制: 李 妍

煤炭工业出版社印刷厂印刷

2006 年 8 月第 1 版·第 1 次印刷

184mm×260mm·21.25 印张·521 千字

0001—5000 册

定价: 32.00 元

凡购本图书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话 (010) 68326294

编辑热线电话 (010) 88379739

封面无防伪标均为盗版

出版说明

随着信息科学与技术的迅速发展，人类每时每刻都会面对层出不穷的新技术、新概念。毫无疑问，在节奏越来越快的工作和生活中，人们需要通过阅读和学习大量信息丰富、具备实践指导意义的图书，来获取新知识和新技能，从而不断提高自身素质，紧跟信息化时代发展的步伐。

众所周知，在计算机硬件方面，高性价比的解决方案和新型技术的应用一直备受青睐；在软件技术方面，随着计算机软件的规模和复杂性与日俱增，软件技术受到不断挑战，人们一直在为寻求更先进的软件技术而奋斗不止。目前，计算机在社会生活中日益普及，随着因特网延伸到人类世界的层层面面，掌握计算机网络技术和理论已成为大众的文化需求。由于信息科学与技术 在电工、电子、通信、工业控制、智能建筑、工业产品设计与制造等专业领域中已经得到充分、广泛的应用，所以这些专业领域中的研究人员和工程技术人员越来越迫切需要汲取自身领域信息化所带来的新理念和新方法。

针对人们对了解和掌握新知识、新技能的热切期待，以及由此促成的人们对语言简洁、内容充实、融合实践经验的图书迫切需要的现状，机械工业出版社适时推出了“信息科学与技术丛书”。这套丛书涉及计算机软件、硬件、网络、工程应用等内容，注重理论与实践相结合，内容实用，层次分明，语言流畅，是信息科学与技术领域专业人员不可或缺的图书。

现今，信息科学与技术的发展可谓一日千里，机械工业出版社欢迎从事信息技术方面工作的科研人员、工程技术人员积极参与我们的工作，为推进我国的信息化建设作出贡献。

机械工业出版社

前 言

要把信息发布到网上，就必须使用能够被大众接受且易懂的语言，也就是要使用一种能够被大多数计算机识别的语言。HTML 语言和用于动态网页设计的 JSP 语言就是这样的语言，使用它们可以设计出真正实现客户和服务端交互的动态网站。JSP 是一种服务器端脚本语言，是由 Sun 公司在其强大的 Java 语言的基础上开发出来的。与其他脚本语言（如 CGI、ASP）相比较，JSP 依托于 Java 语言，继承并发展了 Java 语言的很多优点，例如优秀的跨平台性、易于掌握、运行稳定、安全性好等。

本书面向广大计算机爱好者，特别是对网页制作、编程开发感兴趣的中、高级读者，由浅入深，遵照 JSP 2.0 规范，介绍如何利用 JSP 语言进行交互式动态网站的开发。书中所列举的大量例子既可以帮助读者学习掌握知识，又是制作动态网页和网站的优秀模板——读者只要修改少量的代码，就可以制作出具有专业水准的动态网页和网站。

阅读此书时，可以不必掌握太多的计算机专业知识，也无须丰富的编程经验。无论是熟悉网页制作、Java 编程高手，还是没有任何经验的新手，都能很快地理解并掌握建设一个动态网站所需的前、后台技术，并能将这些技术应用到动态网站的实际开发中。

本书主要由殷庆轩、左昉和王佩楷编写，参加编写的还有谯谊、胡仁喜、王兵学、周冰、董伟、李瑞、王渊峰、张俊生等。书中主要内容来自于作者长期使用 JSP 进行网页开发的经验总结，也有部分内容取自于国内外有关文献资料。由于作者水平有限，书中错误之处在所难免，希望广大读者批评指正。联系邮箱：win760520@126.com。

为了便于读者自学及教师授课，本书配备了书中实例的源程序文件，可以在我社网站（<http://www.cmpbook.com>）免费下载。

编 者

目 录

出版说明	
前言	
第 1 章 JSP 简介	1
1.1 Web 发展史	1
1.2 JSP 的运行原理	2
1.3 JSP 的特点	3
1.4 建立 JSP 运行开发环境	4
1.5 一个简单的 JSP 程序	6
1.6 小结	8
第 2 章 HTML 表单	9
2.1 声明 HTML 表单	9
2.2 BUTTON 控件	10
2.3 INPUT 控件	11
2.4 TEXTAREA 控件	18
2.5 SELECT 控件	19
2.6 LABEL 控件	21
2.7 FIELDSET 控件	22
2.8 小结	23
第 3 章 Java 基础	25
3.1 Java 概述	25
3.1.1 Java 语言的特点	25
3.1.2 面向对象的程序设计	25
3.1.3 简单的 Java 程序	26
3.2 数据类型	27
3.2.1 常量和变量	27
3.2.2 简单数据类型	28
3.3 数组	30
3.3.1 一维数组	30
3.3.2 二维数组	31
3.4 运算符	32
3.4.1 位操作运算符	32
3.4.2 关系运算符	33
3.4.3 逻辑运算符	33
3.4.4 算术运算符	33
3.5 流程控制	34
3.5.1 分支	34
3.5.2 循环	38
3.6 对象、类	41
3.7 Java 程序举例	45
3.8 小结	47
第 4 章 JSP 基本语法	48
4.1 语法规则	48
4.1.1 声明	48
4.1.2 表达式	50
4.1.3 Scriptlet	50
4.1.4 注释	52
4.2 指令元素	54
4.2.1 include 指令	54
4.2.2 page 指令	55
4.2.3 taglib 指令	57
4.3 动作元素	58
4.3.1 <jsp:include>	58
4.3.2 <jsp:forward>	59
4.3.3 <jsp:plugin>	60
4.3.4 <jsp:useBean>	62
4.3.5 <jsp:setProperty>	63
4.3.6 <jsp:getProperty>	64
4.4 小结	65
第 5 章 JSP 内置对象	67
5.1 request 对象	68
5.2 response 对象	75
5.3 application 对象	85
5.4 session 对象	88
5.5 pageContext 对象	102
5.6 out 对象	102
5.7 exception 对象	103
5.8 实例	103
5.8.1 注册	103
5.8.2 计数器	111
5.9 小结	114

第 6 章 表达式语言	116	9.2 JavaBeans 的编写和使用	181
6.1 表达式语言简介	116	9.2.1 JavaBeans 的编写规范	181
6.2 语法规则	117	9.2.2 在 JSP 中使用 JavaBeans	182
6.3 内置对象	118	9.3 实例	184
6.4 运算符	123	9.3.1 数据库连接池	184
6.5 小结	126	9.3.2 注册与登录	191
第 7 章 JSP 的文件操作	127	9.3.3 购物车	208
7.1 File 类	127	9.3.4 BlackJack 游戏	216
7.2 字节输入/输出流	130	9.4 小结	225
7.2.1 字节输入流	130	第 10 章 Java Servlet	226
7.2.2 字节输出流	132	10.1 Servlet 的基本概念	226
7.3 字符输入/输出流	135	10.1.1 什么是 Servlet	226
7.3.1 字符输入流	135	10.1.2 Servlet 的工作原理	228
7.3.2 字符输出流	136	10.1.3 一个简单的 Servlet 示例	228
7.4 RandomAccessFile 类	138	10.2 Servlet 的生命周期	230
7.5 实例	140	10.3 Servlet API	233
7.5.1 文件上传	140	10.4 创建和调用 Servlet	235
7.5.2 文件下载	143	10.4.1 创建 Servlet	235
7.5.3 网上测验	146	10.4.2 调用 Servlet	238
7.6 小结	150	10.5 Servlet 过滤器	240
第 8 章 访问数据库	151	10.6 实例	241
8.1 SQL 语言	151	10.7 小结	244
8.1.1 查询记录	152	第 11 章 网上商店	245
8.1.2 更新记录	153	11.1 系统功能设计	245
8.1.3 添加记录	153	11.2 创建数据源	247
8.1.4 删除记录	154	11.3 页面设计及代码分析	248
8.2 JDBC	154	11.3.1 主页	249
8.3 Web 数据库	155	11.3.2 用户注册	251
8.3.1 建立数据库	155	11.3.3 用户登录	253
8.3.2 创建数据源	156	11.3.4 用户注销	254
8.3.3 连接数据库	156	11.3.5 修改注册信息	254
8.3.4 查询数据库	158	11.3.6 浏览商品	255
8.4 其他数据库的连接	164	11.3.7 搜索商品	259
8.5 实例	165	11.3.8 公共留言板	263
8.5.1 留言板	165	11.3.9 私人留言板	265
8.5.2 网上投票	173	11.3.10 客户购买商品	267
8.6 小结	179	11.3.11 管理员增加商品	268
第 9 章 JavaBeans 组件	180	11.3.12 查看销售记录	273
9.1 JavaBeans 简介	180	11.4 小结	276

第 12 章 博客网站	277	12.5 小结	297
12.1 系统功能设计	277	第 13 章 酒店住宿预定系统	298
12.2 创建数据源	278	13.1 系统设计	298
12.3 Bean 设计及代码分析	279	13.1.1 系统功能设计	298
12.3.1 User 和 UserManager	279	13.1.2 系统结构设计	298
12.3.2 Admin	280	13.2 数据表结构设计	299
12.3.3 Board 和 BoardManager	280	13.3 逻辑层设计及代码分析	301
12.3.4 Msg	281	13.3.1 总体框架	301
12.3.5 Reply	281	13.3.2 DataCenter 包	303
12.4 页面设计及代码分析	282	13.3.3 Accounts 包	304
12.4.1 注册、登录和注销	282	13.3.4 Room 包	306
12.4.2 基本页面设计	284	13.3.5 Order 包	306
12.4.3 导航栏	286	13.4 表现层设计及代码分析	307
12.4.4 个人信息栏	287	13.4.1 用户登录和权限控制	307
12.4.5 浏览主题列表	287	13.4.2 客房、订单、用户信息	
12.4.6 浏览主题文章	289	管理	316
12.4.7 发表主题	291	13.5 小结	326
12.4.8 搜索主题	292	附录	327
12.4.9 日历	292	附录 A 常用技术站点和资源	
12.4.10 论坛统计	295	连接	327
12.4.11 管理员管理主题分类		附录 B 常用的 HTML 标记	
信息	295	元素	328
12.4.12 管理员管理用户信息	297		

第 1 章 JSP 简介

网络科技的迅速发展，网络新闻、电子报纸、聊天室……各种网络化的媒体、资源纷纷出炉。互联网到处充满了生机，也充满了挑战，Web 技术也是日新月异。由于商机涌现，各种需求相应而生，静态网页根本不能满足当今网络实务的需要，于是出现了 CGI、PHP、ASP、ASP.NET、JSP 等为网页加入动态内容的新技术。JSP 是由 Sun Microsystems 公司倡导、多家公司参与建立的一种动态网页技术标准，它在动态网页的建设中有着强大而特别的功能。

1.1 Web 发展史

Web 是全球信息网 (World Wide Web) 的简称，也称为 3W、WWW、万维网等。1982 年，欧洲高能物理研究所的 Tim Berners-Lee 首先提出了 Web 的网络方案。这个方案提出的目的是为本地的工作人员共享科研成果提供便捷的手段，因此，早期的 Web 内容过于专业，那时并没有受到大众的推崇和支持。在 20 世纪 90 年代初，美国 NEXT 公司首先推出了第一款 Web 浏览器，人们通过浏览器可以在网络上使用多媒体技术，从此 Web 开始在世界范围内的 Internet 上迅速发展。

Web 是基于超文本方式、具有良好的用户查询接口的信息查询工具。Web 由遍布世界各地的接入 Internet 网的 Web 服务器组成。Web 将位于不同的空间、存储于不同平台下的计算机上的信息资源有机地组织到了同一个网络上，使得接入此网络的用户可以访问这些信息资源。

Web 是以 Client/Server 方式工作的，由 3 个部分协调完成工作，即客户机、服务器和协议。客户机就是用来接入 Internet 的计算机；服务器是在 Internet 上提供服务、可供客户机访问的计算机；客户机和服务器根据协议来传输文本信息，这个协议叫做 HTTP 协议。客户机通过 Web 浏览器向服务器发出一个查询请求，并从服务器上得到一个回答。根据这个回答，客户可以继续或停止查询。服务器负责对来自客户机的请求进行回答，并负责管理信息和传递信息。服务器不仅自身提供信息服务，而且还可以帮助客户找到其他服务器和其他服务器上的信息资源。

使用超文本 (Hypertext) 是 Web 技术的最大特点。“超文本”，顾名思义就是特殊的、超出一般的文本，它既可以是 Web 页面上文本的一部分，又可以是描述多媒体数据的信息，也可以是指向 Internet 资源的超链接。Web 技术可以方便地处理文本、图像、音频等多媒体数据。设计 Web 页面，最基本的方法是使用 HTML 语言制作。HTML 语言是专门用来描述 Web 页面的，其英文全称就是 Hypertext Markup Language (超文本链接标示语言)。它是在原来文本的基础上，加入一系列的标记符号指明 Web 页面的样式以形成网络文件的。

但是，若只用静态的 HTML 页面，用户看到的都是静态的文字和图像，即使在其中加入了一些动画或者其他多媒体效果，也只是页面上的各种活动效果，无法实现服务器端和用户的真正动态交互。随着人们对网络需求的增加和网络技术日新月异的发展，真正引入了“动

态内容”思想的技术出现了。最早的解决方案是使用了 CGI（公共网关接口），它可以根据需要随时生成 Web 页面的内容。CGI 脚本可以通过简单的鼠标点击来实现，而且可以实现 CGI，并不需要更改客户机/服务器协议。

为了在 Internet 这个具有巨大潜力的市场上占有一席之地，各大公司也纷纷推出自己的方案。如现在比较流行的 Microsoft 公司推出的 ASP 技术、ASP.NET 技术，服务器端嵌入式脚本语言 PHP 以及本书将要介绍的由 Sun 公司推出的 JSP 技术。

1.2 JSP 的运行原理

JSP（Java Server Pages）是一种可以内嵌于 HTML 或 XML 语言中的服务器端嵌入式脚本语言。在传统的 HTML 网页或 XML 网页中加入 Java 程序片段和 JSP 标记，就构成了 JSP 网页。Web 服务器在遇到访问 JSP 网页的请求时，首先执行其中的 Java 程序片段，然后将执行结果以 HTML 的格式返回给客户。程序片段可以操作数据库、重新定向网页以及发送 Email 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很低。

JSP 代码转译成 Servlet 原始码，如图 1-1 所示。

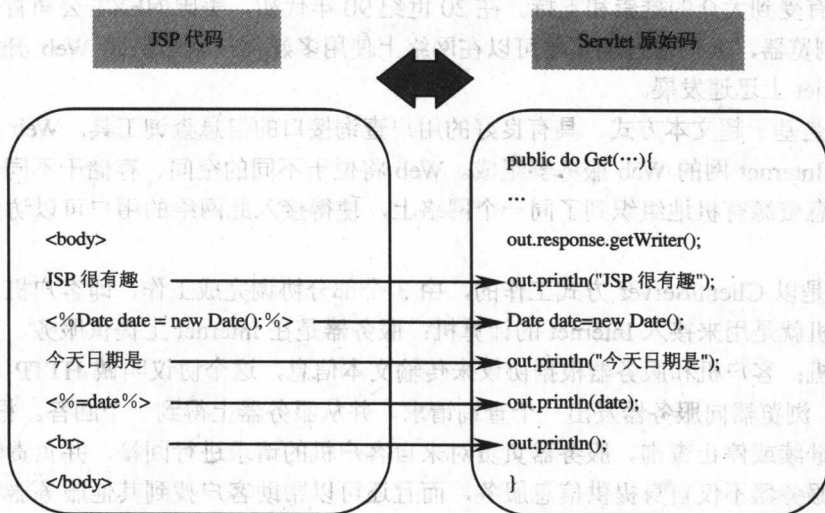


图 1-1 JSP 代码转译成 Servlet 原始码

Web 服务器上的某个 JSP 页面在最近一次修改之后，第一次接收访问请求时，JSP 文件将被 JSP Engine（JSP 引擎）转译成为 Servlet 原始码（.Java 文件）。然后 Servlet 原始码被转交给 Servlet Engine（Servlet 引擎）运行，再利用 Java 编译器将其编译成为 Java class 文件，即 Java Servlet，Servlet 对客户端的消息进行处理，以 HTML 或 XML 的格式把响应返回给客户端，完成服务端与客户端的交互。由于第一次执行 JSP 页面需要进行如上过程，因此时间会稍慢，在此之后运行 JSP 页面的速度就会完全会 Servlet 相同了。由于 Servlet 始终驻于内存，所以响应是非常快的。

如果服务器上的某个 JSP 文件被修改了，则服务器将自动对文件进行重新编译，将编译

结果取代内存中的 Servlet。JSP 的运行原理，如图 1-2 所示。

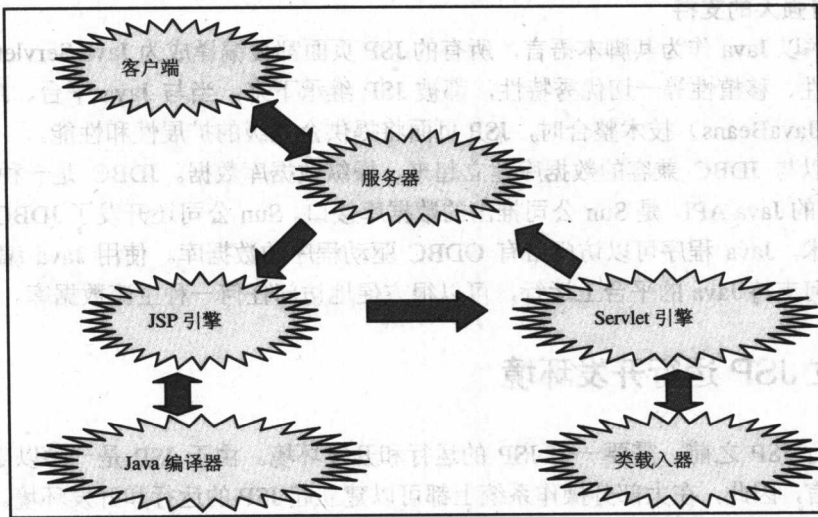


图 1-2 JSP 运行原理

1.3 JSP 的特点

JSP 作为一种动态交互式网页设计语言，有很多优良的特性。

1. 平台无关性

JSP 技术是完全与协议和平台无关的。JSP 支持在任何工作平台上设计动态网页，支持 Web 服务器端工作（需要 Web 服务器本身支持 JSP 语言），而且其返回的结果为 HTML 格式，可以在任何浏览器中显示。几乎所有平台都支持 Java、JSP+JavaBean 可以在所有平台下通行无阻。

2. 高效性

JSP 代码被编译成为 Servlet，并有 Java 虚拟机（Java Virtual Machine）解释执行，编译过程只在程序第一次执行时进行，不需要每次执行程序时都编译。另外，服务器上还有字节码的 Cache 机制，能提高字节码的访问效率。

3. 安全性

使用 JSP 技术时，Web 开发人员利用 HTML 或 XML 来设计和格式化最终页面，而使用 JSP 标记或者脚本来生成动态内容。生成内容的逻辑被封装在标记或 JavaBeans 组件中，对于客户端是不可见的。在服务器端，JSP 引擎解释 JSP 标记和脚本，所有的 JSP 页面都被编译为 Java Servlet，结果以 HTML 或 XML 的形式送回客户浏览器。内容的生成和显示的分离有助于作者保护自己的代码。

4. 可重用性

组件技术的思想是把一个庞大的应用程序分成多个模块，每个模块保持一定的功能独立性。在协同工作时，通过相互之间的接口完成实际的任务。JSP 技术以可重用的 Java 组件模型 JavaBeans 来加强 JSP 的组件使用能力。开发人员能够共享和互换执行普通操作的组件，

或者使得这些组件为更多的使用者或者客户团体所使用，加速了开发过程。

5. 拥有强大的支持

JSP 技术以 Java 作为其脚本语言，所有的 JSP 页面都被编译成为 Java Servlet，Java 的安全性、鲁棒性、移植性等一切优秀特性，都被 JSP 继承下来。当与 Java 平台、J2EE 和 EJB（Enterprise JavaBeans）技术整合时，JSP 页面将提供企业级的扩展性和性能。

JSP 可以与 JDBC 兼容的数据库建立起来，操纵数据库数据。JDBC 是一种可以用于执行 SQL 语句的 Java API，是 Sun 公司推出的数据库接口。Sun 公司还开发了 JDBC-ODBC 桥，利用此项技术，Java 程序可以访问带有 ODBC 驱动程序数据库。使用 Java 编写的应用程序可以在任何支持 Java 的平台上运行，可以很方便地访问任何一种主流数据库。

1.4 建立 JSP 运行开发环境

开始运行 JSP 之前，需要一个 JSP 的运行和开发环境。由于 JSP 是一种以 Java 为脚本的跨平台语言，因此，在大部分操作系统上都可以建立起 JSP 的运行和开发环境。一般来说，读者在学习 JSP 程序设计时，在 Windows 下编写程序代码和调试相对是比较方便的，所以本章将介绍如何在 Windows 操作系统平台下建立 JSP 开发和运行环境。

运行 JSP 需要有 Java 运行环境、JSP Engine 和支持 JSP 的 Web 服务器。Java 运行环境只需要安装 JDK（Java 开发工具包）即可。目前可以提供 JSP Engine 功能的软件有 Resin、Tomcat、WebLogic Application Server、WebSphere Application Server 等。支持 JSP 的 Web 服务器有 Apache、JSWDK（Java Server Web Development，既可以作为 Web 服务器，同时又是 JSP Engine）等。

Tomcat 不仅提供了 JSP Engine，还附带一个 Web 服务器。对于开发者来说，作为开发阶段的 JSP 运行环境十分合适。Tomcat 是 Sun 公司官方推荐的 Servlet 和 JSP 引擎及容器（具体可以参见网页 <http://Java.sun.com/products/jsp/tomcat/>），受到众多软件公司和开发人员的喜爱。Servlet 和 JSP 的最新规范都可以在 Tomcat 的新版本中得到实现。本书所有实例代码都是在 Windows XP+Tomcat+JDK 5.0 环境下调试的。因此，本节将介绍如何建立这样的开发环境。

建立 Windows XP+Tomcat+JDK 5.0 开发环境，需要的软件有：

- Java 开发工具包，下载地址：<http://Java.sun.com/>。
- Tomcat 软件，下载地址：<http://tomcat.apache.org/>。

为支持 JSP 2.0 规范，Java 开发工具包必须在 1.4 版本或 1.4 版本以上，而 Tomcat 服务器必须在 5.0 版本或者 5.0 版本以上。目前，它们的最新版本分别是 `jdk-1_5_0_05-windows-i586-p.exe` 和 `apache-tomcat-5.5.4.exe`，可以分别到上面给出的网址中下载最新的软件并安装。下面介绍安装过程。

双击安装程序 `jdk-1_5_0_05-windows-i586-p.exe`，开始安装。安装过程中使用默认配置即可。JDK 的默认安装目录为“`C:\Program Files\Java`”，也可以更改安装目录。JDK 1.5 版本以前，在安装完 JDK 后，需要修改环境变量 `ClassPath` 和 `Path`。JDK 1.5 版本后，不再需要在环境变量中指定 `ClassPath` 和 `Path`。安装完毕后，进入 MS-DOS 窗口，输入 Java 命令并按〈Enter〉键后，如果出现其命令参数提示信息（如图 1-3 所示），则说明安装正确。

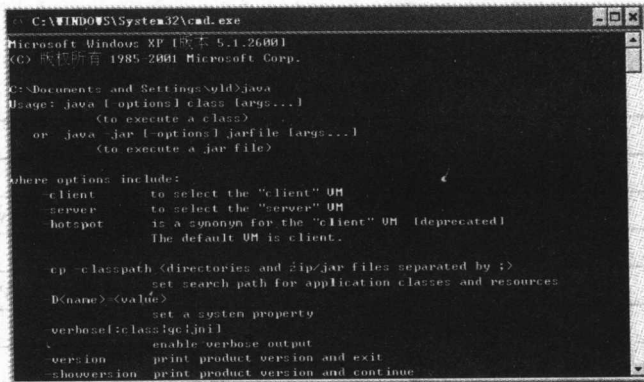


图 1-3 运行 Java 命令

运行 Tomcat 安装程序，按照默认模式安装。当安装进行到如图 1-4 所示的步骤时，选择 JSP 服务器端口，这里选择“8080”作为 Tomcat 服务器端口。安装完毕后，在 IE 浏览器的地址栏中输入：<http://127.0.0.1:8080>，出现 Tomcat 的默认网页，表明配置成功，如图 1-5 所示。

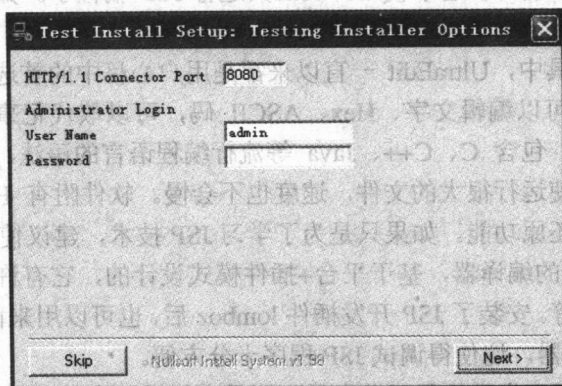


图 1-4 安装 Tomcat 选择端口

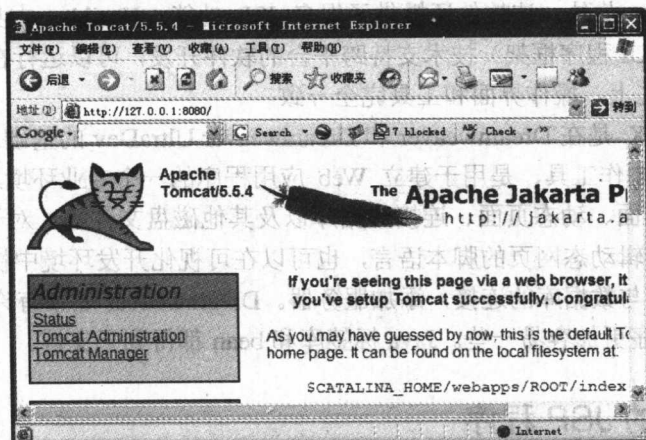


图 1-5 安装 Tomcat 成功界面

Tomcat 的目录结构见表 1-1。

表 1-1 Tomcat 的目录结构

目录名	说明
bin	存放启动或关闭 Tomcat 脚本
conf	包含不同的配置文件, server.xml (Tomcat 的主要配置文件) 和 web.xml
work	存放 JSP 编译后产生的 class 文件
webapp	存放应用程序示例, 以后用户要部署的应用程序也要放到此目录中
logs	存放日志文件
lib/jasper/common	这 3 个目录主要存放 Tomcat 所需的 jar 文件

Tomcat 既可以作为一个小型的 JSP 服务器, 也可以与 Apache 服务器集成在一起提供 JSP 服务。虽然 Tomcat 可以作 Web 服务器, 但其处理静态 HTML 的速度比不上 Apache, 且其作为 Web 服务器的功能远不如 Apache, 因此如果希望对外提供 Web 服务, 把 Apache 和 Tomcat 集成起来是个很好的选择。但作为学习之用的 JSP 运行环境, Tomcat 已经足够。

JSP 引擎搭建起来后就可以着手使用开发工具进行 JSP 编程了。如今下流行的 JSP 开发工具主要有 UltraEdit、Eclipse、Jbuilder 和 Dreamweaver 等。

在众多文本编辑工具中, UltraEdit 一直以来都是用户心目中的首选。UltraEdit 是一套功能强大的文本编辑器, 可以编辑文字、Hex、ASCII 码, 可以取代记事本、内建英文单字检查、具有词法分析功能, 包含 C、C++、Java 等流行编程语言的语法, 编辑速度快, 可同时编辑多个文件, 而且即使运行很大的文件, 速度也不会慢。软件附有 HTML Tag 颜色显示、搜寻替换以及无限制的还原功能。如果只是为了学习 JSP 技术, 建议使用 UltraEdit 编辑器。

Eclipse 是一个免费的编译器, 基于平台+插件模式设计的, 它有许多插件的支持, 可以用来开发 Java、C++、C 等。安装了 JSP 开发插件 lombok 后, 也可以用来协助 JSP 开发。Eclipse 还可以连接 Tomcat 服务器, 这使得调试 JSP 程序十分方便。

JBuilder 软件的目标定位是代码开发人员而不是高级设计人员, 所以 JBuilder 中包含了大量的向导程序和其他针对中间层的快速开发工具。JBuilder 性能稳定、使用方便, 特别适用于创建 Java 组件。此外, 该软件还提供了很多 JSP 功能。JBuilder 本身通过内置的 Java 和 Swing (Java 的 GUI 程序框架) 技术支持跨平台的软件开发, 可以运行在 Windows、Linux 和 Solaris 操作系统之上, 操作界面和工具完全一致。

Dreamweaver MX 是在 Dreamweaver 和 Dreamweaver UltraDev 的基础之上能够对动态网页进行编辑的 Web 制作工具, 是用于建立 Web 应用程序的一个专业环境, 可以辅助网站全程设计, 包括静态页面、动态页面、连接数据库以及其他磁盘文件等。对于初学者来说, 即使他完全不懂各种编辑动态网页的脚本语言, 也可以在可视化开发环境中快速创建动态 Web 站点, 轻松实现网页与数据库的连接, 添加服务器。Dreamweaver 还带有许多 Java 对象, 这让编写 J2EE 应用程序更加容易一些。Java 标签库和 bean 都得到了支持。

1.5 一个简单的 JSP 程序

建立了 JSP 开发和调试环境, 选择了程序开发工具之后, 下面编写一个简单的 JSP 程序。

几乎任何一门计算机语言都从无处不在的“HelloWorld”开始，同样，下面也给出这个著名示例的 JSP 程序代码。

【例】 HelloWorld.jsp

```
<!--HelloWorld.jsp-->
<%@ page language="Java" contentType="text/html;charset=gb2312"%>
<html>
<head>
<title>HelloWorld</title>
</head>
<body bgcolor="#ffffff">
<font face="Arial,HelVetica"><font size=" +2">
<br><br><br>
<center>
<b>HelloWorld!<br>
<%
out.println("我爱 jsp!");
%>
I'm ready to be a JSPer!!!</b>
</center></font></body>
</html>
```

把以上代码保存为文件 HelloWorld.jsp，在服务器根目录下建立一个新的文件夹 chapter01，并把 HelloWorld.jsp 文件放到里面（以后每章出现的例子都会放到相应的目录中去）。打开 IE 浏览器，在地址栏中输入“http://127.0.0.1:8080/chapter01/HelloWorld.jsp”，按〈Enter〉键，则会出现如下的页面，如图 1-6 所示。

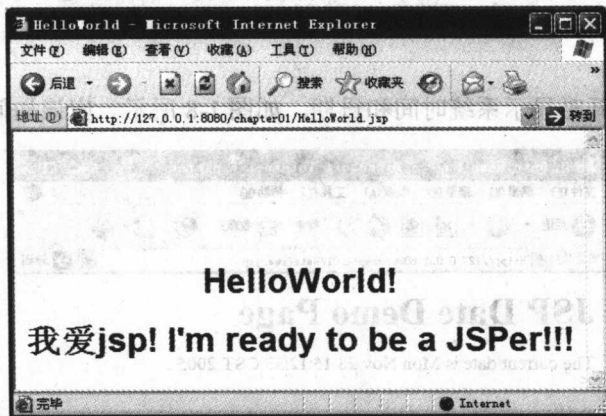


图 1-6 客户端打开 JSP 页面

而如果将 HelloWorld.jsp 文件直接用 IE 打开，则会看到如下页面，如图 1-7 所示。

从上面两个页面中可以发现，没有通过 JSP 引擎而使用 IE 浏览器直接打开时，“我爱 jsp!”这句话没有输出来。观察上面代码可以看出，JSP 页面除了比普通 HTML 页面多一些 Java 代码外，两者具有基本相同的结构。Java 代码是通过“<%”和“%>”符号加入到 HTML

代码中间的。在例子中，它的主要功能是生成并显示字符串“我爱 jsp!”。当直接用 IE 浏览器打开 JSP 文件的时候，没有经过 JSP 引擎的处理，这些 Java 代码就被当成 HTML 语言中的注释，被忽略掉了，因此没有显示。

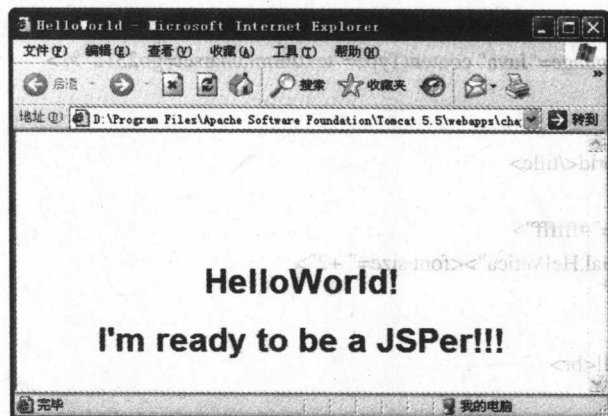


图 1-7 本地打开 JSP 页面

1.6 小结

交互式动态 Web 网站开发技术有很多种，从早期的 CGI 到现在的 ASP、PHP 和 JSP 等。JSP 技术定位于一种规范，基于 Java 语言，而且与平台无关，具有很高的运行效率，开发周期又短，扩展性能也特别强，尤其是在电子商务开发方面显现出了它优秀的的能力。本章还介绍了如何在 Windows 操作系统上创建 JSP 运行开发环境，最后通过一个小程序“HelloWorld”带领大家进入奇妙的 JSP 世界。

练一练：

若希望网页在打开时显示系统时间和日期，如图 1-8 所示。应该如何编写 JSP 代码？

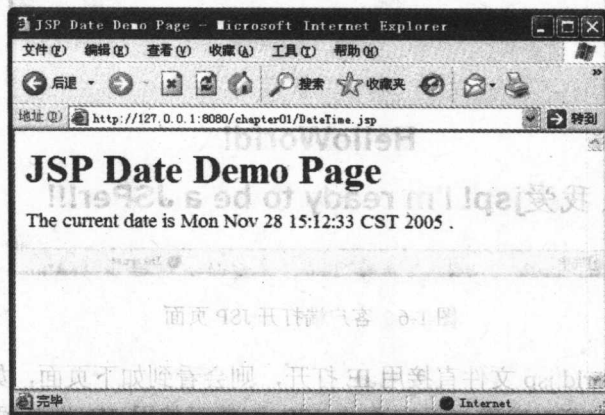


图 1-8 显示时间和日期的 JSP 页面

第 2 章 HTML 表单

动态网页相对于传统的静态网页，其优点是客户端和服务端可以互动。用户可以在网页上输入一个数据，而远在千里之外的服务端就可以通过网络马上作出相应的响应。表单（FORM）就是提供给用户输入信息的一个工具。JSP 网页通过 HTML 表单实现了大部分用户端和服务端之间的互动。

2.1 声明 HTML 表单

在 HTML 语言中，建立一个表单需要包括下面几个部分。

声明表单的语法格式为：

```
<FORM>
... ..
</FORM>
```

常见的表单属性说明如下：

- ACTION=URI（表单处理程序）
- METHOD=[get|post]（发送表单的 HTTP 方式，默认为 get）
- ACCEPT-CHARSET=Charsets（可支持的字符列表）
- TARGET=FrameTarget（显示表单的框架）

当用户填写完表单后，通过“TYPE=submit”的<INPUT>或<BUTTON>元素发送给表单，表单的值将被发送到表单的“ACTION”属性所规定的 URI。“ACTION”属性在<FORM>中是不可省略的，它指向服务端处理表单数据的程序或组件。

URI 的英文是 Uniform Resource Identifier，是惟一的能将自定义的一系列标签和标签前缀联系起来。一个 URI 可以是下面的各种形式：

- 1) 使用 Uniform Resource Locator (URL)，在 RFC 2396 中有相关的定义和说明，可以到“<http://www.hut.fi/u/jkorpela/rfc/2396/full.html>”中查找详细的信息。
- 2) 使用 Uniform Resource Name (URN)，在 RFC 2396 中有定义。
- 3) 使用一个绝对或相对的路径。

表单输入的方式取决于“METHOD”属性。当“METHOD”的值是“get”时，表单的输入就会作为“HTTP GET”请求，将表单数据发送给“ACTION”属性所规定的 URI 上。如果“METHOD”的值是“post”时，表单的输入就作为“HTTP POST”请求。输入中可能出现包含非 ASCII 码的字符或表单内容超过 100 个字符，这时不能使用“get”方式发送表单内容，必须使用“post”方式。

“ACCEPT-CHARSET”属性规定了可以被表单处理程序支持的字符编码列表。列表的值用逗号或者空格分开，默认值为“UNKNOWN”。