



高等学校精品规划教材

JSP程序设计实用教程 (第二版)

主 编 梁建武

副主编 何英姿 陈语林 赵 晶



中国水利水电出版社
www.waterpub.com.cn

21 世纪高等学校精品规划教材

JSP 程序设计实用教程

（第二版）

主 编 梁建武

副主编 何英姿 陈语林 赵 晶



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书按易学、易懂、易掌握的原则,结合 JSP 内部知识体系,由浅入深、循序渐进地介绍了如何用 JSP 进行 Web 动态网站的开发和基于 B/S 的网上教学系统的规划、设计、代码编制、调试。全书共 11 章,从 JSP 基础开始,到 Web 页面制作基础、语法、JSP 中的内置对象、Web 数据库开发、JavaBean 的应用、文件操作、综合实例等,循序渐进地对 JSP 进行了全面的介绍。

本书内容丰富、讲解清晰,在讲解过程中力避代码复杂冗长。简短的实例特别有助于初学者仿效理解、把握问题的精髓和对应用程序框架的整体认识;综合实例为读者开发大型的应用程序提供经典范例。本书的创新之处在于为读者提供了开发的过程,而不局限于每个知识点。

本书可作为高等院校计算机或相关专业的教材,也可供广大工程技术人员学习参考。

本书源代码和电子教案可到中国水利水电出版社或万水书苑网站上免费下载,网址为 <http://www.waterpub.com.cn/softdown/>或 <http://wsbookshow.com>。

图书在版编目(CIP)数据

JSP程序设计实用教程 / 梁建武主编. -- 2版. --
北京:中国水利水电出版社,2013.12
21世纪高等学校精品规划教材
ISBN 978-7-5170-1600-7

I. ①J… II. ①梁… III. ①JAVA语言—网页制作工
具—高等学校—教材 IV. ①TP312②TP393.092

中国版本图书馆CIP数据核字(2013)第317497号

策划编辑:雷顺加 责任编辑:李 炎 加工编辑:史永生 封面设计:李 佳

书 名	21 世纪高等学校精品规划教材 JSP 程序设计实用教程(第二版)
作 者	主 编 梁建武 副主编 何英姿 陈语林 赵 晶
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn
经 售	电话: (010) 68367658(发行部)、82562819(万水) 北京科水图书销售中心(零售) 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京泽宇印刷有限公司
规 格	184mm×260mm 16 开本 19.25 印张 473 千字
版 次	2007 年 5 月第 1 版 2007 年 5 月第 1 次印刷 2013 年 12 月第 2 版 2013 年 12 月第 1 次印刷
印 数	0001—4000 册
定 价	35.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

前 言

JSP 是当前主流的 Web 开发技术,它是一种基于 Java 的服务器语言。由于 JSP 的强大功能和显著优势,JSP 技术已经成为网络时代的宠儿。应用 JSP 技术不仅能制作出具有各种功能的 Web 页面,还能开发基于 B/S 结构的大型软件系统,如 MIS、ERP 系统等。

本书第 1 章介绍 JSP 的基本概念和开发环境的配置,以及一些常用开发工具,还介绍了 Web 应用程序的工作原理以及 B/S、C/S 等软件编程体系结构,同时在第一版的基础上更新了开发环境版本。第 2 章介绍 Web 网页的基础知识——HTML 语言以及 CSS 层叠样式表,同时还介绍了网页制作工具 Dreamweaver 的配置和使用。第 3 章介绍开发动态网页的脚本语言 JavaScript 和 JSP 必备语言 Java,并将这两种语言进行简要的比较。从第 4 章到第 8 章详细介绍 JSP 语法,其中包括 JSP 基本语法、JSP 内置对象、JSP 与数据库开发、JavaBean 的使用、JSP 与文件操作等,并通过实例和习题让读者深化理解和巩固要点。其中第 6 章删除了第一版中内容比较过时的 Access 数据库内容。第 9 章和第 10 章分别对两个大型系统进行实例分析,带领读者学习基于 JSP+JavaBean 开发模式的 JSP 应用技术。这两章都从系统的总体设计起步,讲述了系统数据库设计以及各功能模块的具体设计,对应第 6 章数据库内容,第 9 章把上一版本使用的 Access 数据库改成目前更为流行的 SQL Server 数据库,与上一版不同的是,第 9 章源代码单独作为一个项目从整本书的项目代码中分离。为了减少篇幅,把少量类似的功能模块留给读者自己完成,读者可以参照中国水利水电出版社网站上的代码。第 11 章对 JSP 程序设计进行总结与展望,介绍了 Servlet 技术、MVC 模型、Java 对象持久化技术等开发大型系统所应掌握的知识以及简单介绍现在广泛使用的 SSH 框架知识。此外,在各个章节对第一版都有更新、删除、完善,如图片印刷出来看不清楚的问题,语言逻辑错误问题等。

本书的最大特点在于对 JSP 中的每个知识点都精心编排了必要的实例。通过对实例的学习,读者会对所学知识有更深入的理解,并能更全面地掌握用 JSP 开发 Web 动态网站、B/S 结构的大型软件等系统的思路、技巧和体系。

本书深入浅出、循序渐进、选材适当、结构严谨,所有界面和代码都通过了编者的验证调试,同时本书配有完整代码,读者可到中国水利水电出版社网站 <http://www.waterpub.com.cn/softdown/>或万水书苑 <http://wsbookshow.com> 上下载。

本书不仅可以作为高等院校或相关专业的教材,也可以作为 Web 技术开发人员的参考书。

本书由梁建武担任主编,何英姿、陈语林、赵晶担任副主编。各章主要编写人员分工如下:梁建武负责全书的体系结构并编写了第 3、4、6、7、10 章及全书统稿,何英姿负责全书的稽核并编写了第 8、9、11 章,陈语林编写了 1、5 章,赵晶编写了第 2 章。此外,参与本书编写工作的还有程浩辉、李洪臣、陈语林、张雷、杜伟、付世凤、何志斌、刘军军、李华伟、谭海龙、

文拯等。

本书编写过程中,得到了许多专家和同仁的热情帮助和大力支持,还得到中国水利水电出版社万水分社的领导和编辑的指导与帮助,谨此向他们表示最真挚的感谢!

由于计算机技术发展十分迅速以及作者水平所限,加之时间仓促,书中疏漏和错误在所难免,敬请广大读者批评指正。

编者 于中南大学

2013年9月

目 录

前言

第 1 章 JSP 和 Web 应用程序.....	1	习题二.....	43
1.1 JSP 概述.....	1	第 3 章 JavaScript 和 Java 语言.....	45
1.1.1 什么是 JSP.....	1	3.1 JavaScript.....	45
1.1.2 JSP 的优点.....	2	3.1.1 JavaScript 概述.....	45
1.2 JSP 开发环境的配置.....	2	3.1.2 JavaScript 语言结构.....	47
1.2.1 JDK 的安装和配置.....	2	3.1.3 JavaScript 的事件驱动.....	51
1.2.2 Tomcat 的安装与配置.....	3	3.1.4 JavaScript 的对象.....	53
1.2.3 Tomcat 的目录结构.....	4	3.1.5 JavaScript 实例.....	65
1.3 JSP 常用开发工具.....	5	3.2 Java 编程语言.....	69
1.3.1 EditPlus.....	5	3.2.1 数据类型与关键字.....	70
1.3.2 Eclipse.....	6	3.2.2 运算符和表达式.....	71
1.4 Web 应用程序.....	7	3.2.3 流程控制语句.....	73
1.4.1 Web 应用程序的工作原理.....	7	3.2.4 面向对象程序设计.....	81
1.4.2 Web 服务器和应用程序服务器.....	8	3.2.5 多线程.....	83
1.5 软件编程体系.....	9	本章小结.....	87
1.5.1 C/S 和 B/S.....	9	习题三.....	89
1.5.2 C/S 和 B/S 之比较.....	9	第 4 章 JSP 语法.....	91
本章小结.....	11	4.1 JSP 语法概述.....	91
习题一.....	11	4.1.1 JSP 工作原理.....	91
第 2 章 JSP 页面制作基础.....	13	4.1.2 JSP 页面结构.....	92
2.1 HTML 语言.....	13	4.2 注释.....	92
2.1.1 HTML 概述.....	13	4.2.1 JSP 隐藏注释.....	93
2.1.2 简单格式标记.....	15	4.2.2 HTML 注释.....	93
2.1.3 超链接与图片标记.....	22	4.3 JSP 指令.....	93
2.1.4 表格设计.....	26	4.3.1 page 指令.....	94
2.1.5 表单设计.....	29	4.3.2 include 指令.....	96
2.2 CSS 编程技术.....	32	4.3.3 taglib 指令.....	97
2.2.1 CSS 的定义方法.....	33	4.4 JSP 脚本元素.....	97
2.2.2 使用 CSS 显示 HTML 文档.....	34	4.4.1 JSP 声明.....	97
2.3 Dreamweaver.....	36	4.4.2 JSP 表达式.....	98
2.3.1 操作界面.....	36	4.4.3 脚本小程序.....	98
2.3.2 用 Dreamweaver 建立 JSP 站点.....	38	4.5 JSP 动作.....	100
本章小结.....	41	4.5.1 include 动作元素.....	100

4.5.2	forword 动作元素	101	6.3.1	数据库的连接过程	147
4.5.3	plugin 动作元素	103	6.3.2	数据库基本操作	148
4.5.4	param 动作元素	103	6.4	JSP 数据库编程实例	150
4.5.5	useBean 及 setProperty 和 getProperty 动作元素	104	6.4.1	用户管理信息系统	150
	本章小结	106	6.4.2	分页显示的问题	158
	习题四	107		本章小结	162
第 5 章	JSP 内置对象	109		习题六	162
5.1	request 对象	109	第 7 章	JSP 与 JavaBean	164
5.1.1	request 应用实例	111	7.1	JavaBean 概述	164
5.1.2	窗体传递中文的问题	117	7.1.1	JavaBean 简介	164
5.2	response 对象	118	7.1.2	JavaBean 的任务	165
5.2.1	动态响应 contentType 方法	118	7.2	JavaBean 的应用	167
5.2.2	设定表头的方法	119	7.2.1	如何创建 JavaBean	167
5.2.3	设定响应状态码的方法	119	7.2.2	如何调用 JavaBean	167
5.2.4	response 重定向	120	7.2.3	JavaBean 的存放目录	168
5.2.5	其他方法	120	7.3	JavaBean 应用实例	169
5.2.6	Cookie 对象的使用	121	7.3.1	使用 JavaBean 连接数据库	169
5.3	其他内置对象	123	7.3.2	进程条的实现	171
5.3.1	out 内置对象	123		本章小结	177
5.3.2	session 对象	125		习题七	178
5.3.3	application 对象	128	第 8 章	JSP 与文件操作	179
5.3.4	page 对象	130	8.1	File 对象	179
5.3.5	config 对象	130	8.1.1	获取文件的属性	180
5.3.6	exception 对象	131	8.1.2	创建文件和文件夹	181
5.3.7	pageContext 对象	131	8.1.3	删除文件和文件夹	182
	本章小结	132	8.2	输入输出流	183
	习题五	133	8.2.1	字节流	183
第 6 章	JSP 与数据库开发	135	8.2.2	字符流	185
6.1	数据库基础	135	8.3	文件上传下载	187
6.1.1	关系数据库简介	136	8.3.1	实现多个文件上传的两种方法	187
6.1.2	MySQL 数据库	137	8.3.2	JSP 上传组件	188
6.1.3	SQL Server 数据库	139	8.3.3	上传下载实例	191
6.1.4	Oracle 数据库	141	8.4	网页计数器的实现	196
6.2	JDBC 基础	142		本章小结	198
6.2.1	JDBC 简介	142		习题八	198
6.2.2	JDBC 原理和驱动	143	第 9 章	网上教学系统	200
6.2.3	常用的 JDBC 类与方法	144	9.1	系统总体设计	200
6.3	JSP 中使用数据库	147	9.1.1	系统功能模块	200
			9.1.2	系统总体框架	201

9.2 系统数据库设计	201	10.3.1 系统公用模块	266
9.2.1 数据库需求分析	202	10.3.2 销售系统基本信息	270
9.2.2 数据表设计	202	10.3.3 销售业绩统计查询	276
9.2.3 创建数据表	204	10.3.4 部署和发布系统	286
9.3 各功能模块的具体实现	204	本章小结	286
9.3.1 系统公用模块	204	第 11 章 JSP 展望	287
9.3.2 登录模块	206	11.1 JSP 技术难点	287
9.3.3 管理员模块	210	11.2 Servlet 技术	289
9.3.4 学生模块	244	11.2.1 Servlet 的生命周期	290
9.3.5 教师模块	253	11.2.2 Servlet 与表单的交互	291
9.3.6 部署和发布系统	262	11.2.3 Servlet 控制会话	291
本章小结	262	11.3 MVC 模型	292
第 10 章 销售业绩统计系统	263	11.3.1 常用三层结构模式	292
10.1 系统总体设计	263	11.3.2 MVC 模型	293
10.1.1 系统功能模块	263	11.3.3 MVC 模式的优缺点	295
10.1.2 系统总体框架	264	11.4 Java 对象持久化技术	295
10.1.3 系统的编程思想实现	264	11.4.1 Struts 框架简介	296
10.2 系统数据库设计	265	11.4.2 Spring 简介	297
10.2.1 数据库需求分析	265	11.4.3 Hibernate 简介	298
10.2.2 数据表设计	265	本章小结	299
10.3 各功能模块的具体实现	266	参考文献	300

第 1 章 JSP 和 Web 应用程序



本章导读

JSP 是一种开发 Web 应用程序的新技术。自发布以来,它一直受到密切的关注。为什么 JSP 发展如此迅猛,原因之一是它基于 Java 技术,而 Java 极适合企业级计算。另一个原因在于 JSP 支持强大的 Web 应用程序开发模式,它可以把外观呈现与处理过程分隔开来,使得擅长图形制作、布局的网页设计师与精通服务器端技术(例如多线程资源池、数据库和高速缓存)的程序员能够协调地工作。尽管其他一些技术也支持类似的开发模式,例如 ASP、PHP 和 ColdFusion,但它们当中没有一种能具备 JSP 的所有优点。



本章要点

- JSP 概述
- JSP 开发环境的配置
- JSP 常用开发工具
- Web 应用程序
- 软件编程体系

通过学习以上内容,读者要初步了解动态网站开发的相关技术。

1.1 JSP 概述

JSP (Java Server Pages) 是由 Sun 公司于 1999 年 6 月在 Java 语言基础上开发出来的一种动态网页制作技术,在 Sun 正式发布 JSP 之后,这种新的 Web 应用开发技术很快引起了人们的关注。

1.1.1 什么是 JSP

网络技术日新月异,细心的网友会发现许多网页文件扩展名不再只是“.htm”、“.html”,还有“.jsp”、“.asp”等,这些网页都是采用动态网页技术制作出来的。

早期的动态网页主要采用 CGI 技术,但由于编程困难、效率低下、修改复杂,所以有逐渐被新技术取代的趋势。目前颇受关注的几种新技术有 JSP、PHP (Hypertext Preprocessor) 和 ASP (Active Server Pages)。

JSP 是一种动态网页技术标准,在传统的网页 HTML 文件 (*.htm, *.html) 中加入 Java 程序片段和 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 的形式返回给客户。程序片段可以操作数据库、重新定向网页等,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,可以实现无

Plugin, 无 ActiveX, 无 Java Applet, 甚至无 Frame。

注意: 动态网站并不是指具有“动态效果”的网站, 而是指网站的底层信息会不断变化, 例如实时显示更新的天气、新闻等信息。

1.1.2 JSP 的优点

JSP 技术在多个方面加速了动态 Web 页面的开发, 它具有很多优点。

首先, 它将 Web 页面设计工作与服务器逻辑设计工作分离。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最终页面; 使用 JSP 标识或者小脚本来生成页面上的动态内容。生成内容的逻辑被封装在标识和 JavaBean 组件中, 并且捆绑在小脚本中, 所有的脚本在服务器端运行。如果核心逻辑被封装在标识和 Bean 中, 那么其他人, 如 Web 管理人员和页面设计者, 能够管理和编辑修改 JSP 页面, 而不影响内容的生成。在服务器端, JSP 引擎解释 JSP 标识和小脚本, 生成所请求的内容 (例如, 通过访问 JavaBean 组件, 使用 JDBC TM 技术访问数据库, 或者包含文件), 并且将结果以 HTML (或者 XML) 页面的形式发送回浏览器。这有助于作者保护自己的代码, 而又保证任何基于 HTML 的 Web 浏览器的完全可用性。

其次, 强调可重用的组件。绝大多数 JSP 页面依赖于可重用的、跨平台的组件 (JavaBean 或者 Enterprise JavaBean TM 组件) 来执行应用程序所要求的更为复杂的处理。开发人员能够共享和交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。基于组件的方法加速了总体开发过程, 并且使得各种组织在他们现有的技能和优化结果的开发努力中得到平衡。

最后, 采用标识简化页面开发。Web 页面开发人员不会都是熟悉脚本语言的编程人员。JSP 技术封装了许多功能, 这些功能可以在 XML 标识中进行动态内容的生成。标准的 JSP 标识能够访问和实例化 JavaBean 组件, 设置或者检索组件属性, 下载 Applet, 以及执行用其他方法更难于编码和耗时的功能。

1.2 JSP 开发环境的配置

要学习 JSP 开发, 必须先搭建一个符合 JSP 规范的开发环境。Sun 推出的 JSP 是一种执行于服务器端的动态网页开发技术, 它基于 Java 技术。执行 JSP 时需要在 Web 服务器上架设一个编译 JSP 网页的引擎。配置 JSP 环境可以有多种途径, 但主要工作就是安装和配置 Web 服务器和 JSP 引擎。本书以实用为原则, 介绍了以 JDK+Tomcat 配置 JSP 环境的方法。

(1) JDK。Java 的软件开发工具, 是 Java 应用程序的基础。JSP 是基于 Java 技术的, 所以配置 JSP 环境之前必须要安装 JDK。本书使用的版本是 JDK 1.7。

(2) Tomcat 服务器。Tomcat 服务器是 Apache 组织开发的一种 JSP 引擎, 本身具有 Web 服务器的功能, 可以作为独立的 Web 服务器来使用。同时该软件也是免费的, 对于初学者来说, Tomcat 是一个很不错的选择。本书使用的版本是 Tomcat 7.0。

以上的软件都可以到 Sun 公司的网站免费下载。

1.2.1 JDK 的安装和配置

下载好 JDK, 单击安装, 选择好安装路径, 正确安装在计算机上。安装完成后右击桌面

上的“我的电脑”，选择“属性”，然后选择“高级”里面的“环境变量”，在打开的界面中需要设置三个系统变量 JAVA_HOME、Path、CLASSPATH。在没安装过 JDK 的环境下，Path 变量是本来存在的，而 JAVA_HOME 和 CLASSPATH 并不存在，需要新建。

(1) 新建一个系统变量，变量名为 JAVA_HOME，顾名思义该变量的作用就是声明 Java 的安装路径，笔者的安装路径为“D:\Program Files\Java\jdk1.7.0_09”。新建 JAVA_HOME 变量，如图 1.1 所示。

(2) 在系统变量里面找到 Path，然后单击“编辑”，Path 变量的含义就是系统在任何路径下都可以识别 Java 命令，在其变量值处填入“%JAVA_HOME%\bin”，修改 Path 变量，如图 1.2 所示。



图 1.1 新建 JAVA_HOME 变量截图

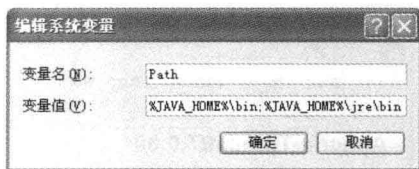


图 1.2 修改 Path 变量截图

(3) 再新建一个系统变量，变量名为 CLASSPATH，该变量的含义是为 Java 加载类 (class or lib) 路径，只有将类加载在 CLASSPATH 中，Java 命令才能识别。其值为“.;%JAVA_HOME%\lib\tools.jar;”，新建 CLASSPATH 变量，如图 1.3 所示。

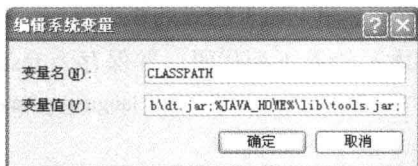


图 1.3 新建 CLASSPATH 变量截图

以上三个变量设置完毕，则单击“确定”按钮直至属性窗口消失，接下来是验证安装是否成功。先打开“开始”→“运行”，输入 cmd，进入 DOS 系统界面。然后输入 java -version 命令，如果安装成功，系统会显示 java version "1.7.0_09" (不同版本号则显示不同)。显示界面如图 1.4 所示。

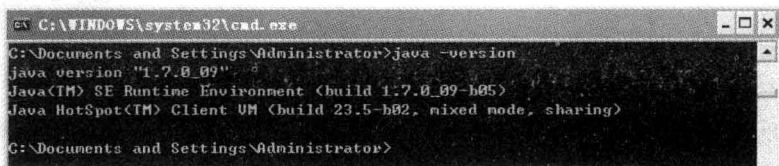


图 1.4 显示版本信息截图

1.2.2 Tomcat 的安装与配置

下载并安装 Tomcat，读者可自行选择安装路径，笔者的安装路径为 D:\Tomcat。安装程序会自动搜索 JDK 的安装路径，如果没有正确显示，则可以手工修改，同时访问端口号也可以更改，默认是 8080。接下来就开始拷贝文件了，成功安装后，程序会提示启动 Tomcat 并查看

readme 文档。

如果下载的是 zip 压缩包格式的，直接解压缩到 D:\Tomcat（当然也可以是其他目录），并且按照前面设置系统环境变量的方法设置新的系统环境变量 TOMCAT_HOME，值为安装路径 D:\Tomcat。默认访问端口是 8080，如果想改动，可以在文件夹 conf 下找到 server.xml 文件，用记事本打开，查找到 8080，然后改成你想设置的端口号即可。

至此安装与配置都已完成，重启计算机，在 Tomcat 的安装文件夹 bin 里找到 startup.bat 文件，双击即可启动 Tomcat，打开浏览器输入 http://localhost:8080（如端口更改，则将 8080 改成你所更改的数字，下同）即可看到 Tomcat 的相关信息，如果不能打开如图 1.5 所示的页面，则表示没有正确配置。

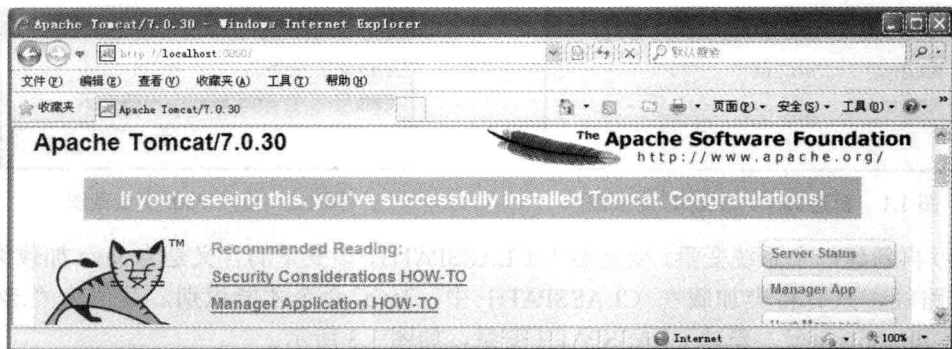


图 1.5 Tomcat 欢迎页面

打开文本编辑器，如记事本，输入下列代码，并保存为 test.jsp（注意扩展名为.jsp）。

```
<%@ page contentType="text/html; charset=gb2312" language="java" errorPage="" %>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title>JSP 测试页面</title>
</head>
<body>
<!--输出字符串"Hello World!"-->
<%out.println("<h1> Hello World! </h1>");%>
</body>
</html>
```

把 test.jsp 放在 D:\Tomcat\webapps\目录下，在地址栏中输入 http://localhost:8080/test.jsp，如果浏览器中显示“Hello World!”，则说明已经成功运行了 JSP 页面，运行结果如图 1.6 所示。



图 1.6 JSP 测试页面

1.2.3 Tomcat 的目录结构

/bin——存放 Tomcat 的脚本文件。

/conf——存放 Tomcat 的配置文件。

/server 有三个子目录 classes, lib, webapps:

/server/lib——存放 Tomcat 所需要的各种 jar 文件（不能被其他 Web 服务器访问）。

/server/webapps——存放 Tomcat 两个自带 Web 应用——admin 应用和 manager 应用。

`/common/lib`——存放 Tomcat 服务器以及所有 Web 应用都可以访问的 jar 文件。

`/shared/lib`——存放所有 Web 应用都可以访问的 jar 文件（不能被 Tomcat 访问）。

/logs——存放 Tomcat 日志文件。

/webapps——当发布 Web 应用时，默认情况下把 Web 文件夹放于此目录下。

/work.jsp——生成的 servlet 放置在此目录下。

注意：在每个 Web 应用下/wen-inf/lib 下也可以放 jar 文件，但只对当前应用有效。

由于 Tomcat 本身具有 Web 服务器的功能，因此不必安装 Apache。但其处理静态 HTML 页面的速度比不上 Apache，且其作为 Web 服务器的功能远不如 Apache，因此通常把 Apache 和 Tomcat 集成起来，用 Apache 充当 Web 服务器，而 Tomcat 作为专用的 JSP 引擎。这种方案的配置比较复杂，但是能让 Apache 和 Tomcat 完美整合，实现强大的功能。有兴趣的读者可以查看相关资料进行设置。

1.3 JSP 常用开发工具

JSP 引擎搭建起来后就可以着手使用开发工具进行 JSP 的编程了, 现下流行的 JSP 开发工具主要有 Eclipse、JBuilder、NetBeans、EditPlus、UltraEdit、Dreamweaver 等。最简单的方法是用记事本创建 JSP 文件, 然后将文件拷贝到 Webapps 目录下运行。本节主要介绍 EditPlus、Eclipse 两种工具的一些基本情况, 在第 2 章中再详细介绍 Dreamweaver, 实际开发时可以参照其各自的特点, 结合自身开发环境选择合适的开发工具。

1.3.1 EditPlus

EditPlus 是一款功能非常强大的文本编辑工具，它支持自定义工具组、自定义文件类型等功能，对于从事程序设计和网页制作的工作者实在是不可或缺！

EditPlus、UltraEdit、Notepad、记事本等是很多高手至今坚持使用的开发工具，其中 EditPlus 最为方便，可以只把它当作高彩显示代码的工具。EditPlus 支持 HTML、CSS、PHP、ASP、JSP、Perl、C/C++、Java、JavaScript 和 VBScript 的语法加亮，还可以自己扩展定制。不仅如此，EditPlus 经过设置后还能直接编译和运行 Java 等程序，读者可以在网上查阅相关的资料。

EditPlus 的运行界面如图 1.7 所示。

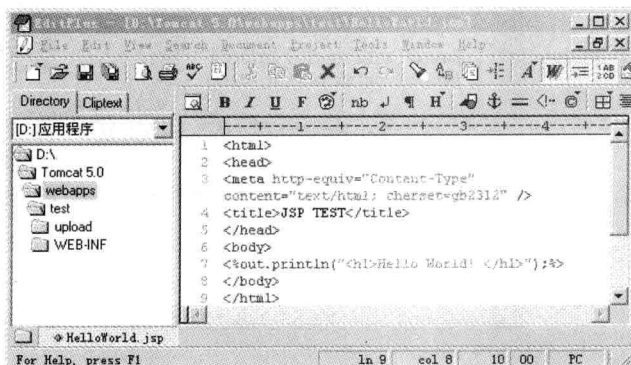


图 1.7 EditPlus 的运行界面

1.3.2 Eclipse

目前 Java 开发领域的各种集成开发环境（IDE）呈现出百花齐放的局面，在所有的 IDE 中，Eclipse 可以说是最有发展前途的产品之一。Eclipse 是一个开放的开发平台，通过插件系统，可以拥有几乎无限的扩展能力，因此越来越多的程序员使用它来开发程序。鉴于本书讲述的是 JSP 的开发，因此将重点讲述怎样使用 Eclipse 开发 JSP。

首先，下载 Eclipse 的 Win32 安装文件，官方网站 <http://www.eclipse.org/> 提供了较新版本的下载（读者可根据自己需求下载），直接运行 eclipse.exe，程序会自动找到 JDK 并完成相应的配置。下载 Sysdeo Eclipse Tomcat 6.x，它是 Tomcat 在 Eclipse 上的一个插件，一直以来用它做 Eclipse 下 Tomcat 的启动开发平台，解压 TomcatPluginV31beta.zip 到 Eclipse 安装目录下的 plugins 目录中。

下面开始配置 Eclipse。

- （1）启动 Eclipse。
- （2）打开菜单 Window→Preferences。
- （3）在左侧选择 Tomcat，可以看到右侧出现一些表单。

（4）Tomcat Version 选择 Version 6.0.x，Tomcat Home 选择 Tomcat 的安装路径，Configuration File 中会自动填入 Tomcat 的配置文件 server.xml。

（5）展开左侧的 Tomcat 菜单，选择 JVM Settings，JRE 选择 Detected VM，单击 Apply 后单击 OK。

（6）工具栏中应该多了一个小猫的图标，如果没有的话，选择菜单 Window→Customize Perspective，展开 Other 选项，在 Tomcat 上打勾即可。

（7）单击 Start Tomcat 按钮，Tomcat 便在控制台中启动了。

运行界面如图 1.8 所示。

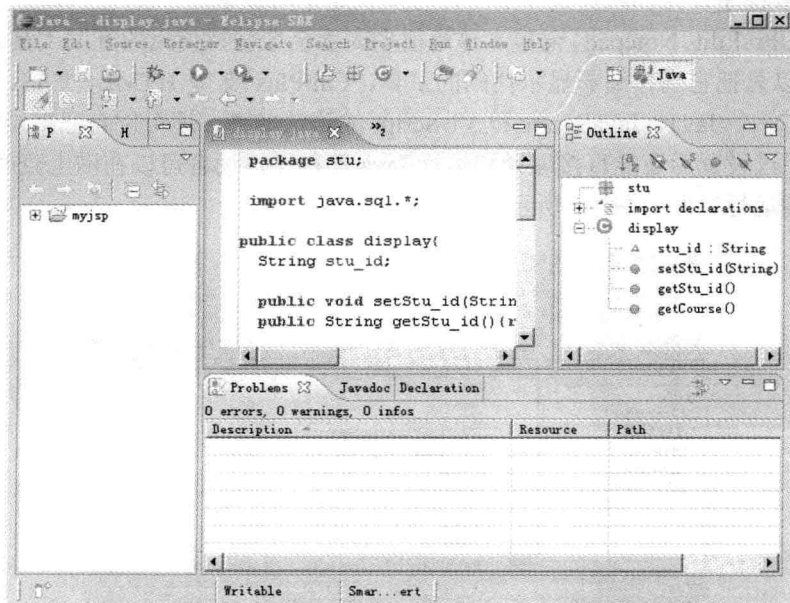


图 1.8 Eclipse 的运行界面

1.4 Web 应用程序

前面的几节中已经使用过 Web 应用程序术语，其所指的既不是一个真正意义上的 Web 网站，又不是一个传统的应用程序。换句话说，它是一些 Web 网页和用来完成某些任务的其他资源的一个集合。这些网页存储在 Web 服务器上，其部分内容或全部内容是未确定的。只有当用户请求 Web 服务器中的某个页面时，才确定该页的最终内容。页面内容基于用户的操作，随请求的不同而变化，这种页面称为动态页面，反之则称为静态页面。因而 Web 应用程序是一组静态和动态 Web 页的集合。

静态页面是指服务器接收到请求后，Web 服务器直接将该页发送到浏览器，不对其进行修改。相反，将动态页面发送到浏览器之前，服务器将对该页进行修改。例如，读者可以设计一个页面来显示学生名单，而这些信息（例如学生姓名和结果）在接到请求时根据查询条件再确定。

建立 Web 应用程序是为了解决多种问题，Web 应用程序的一般用途如下：

- (1) 用户可以快速方便地在一个内容丰富的 Web 站点上查找信息。
- (2) 收集、保存和分析用户提供的数据。
- (3) 对内容不断变化的 Web 站点进行更新。

1.4.1 Web 应用程序的工作原理

1. 处理静态页面的工作原理

一般的 Web 站点由一组相关的 HTML 页面和文件组成，这些页面和文件保存在运行 Web 服务器的计算机上。当用户在执行某些操作时，如单击 Web 页上的某个链接、在浏览器中选择一个书签、或在浏览器的“地址”文本框中输入一个 URL 并单击“转到”时，便生成一个页面请求。

静态页面的内容由网页设计人员确定，当接到请求时，内容不发生更改。页面的每一行代码都是在将页面放置到服务器之前由设计人员编写好的。严格来说，“静态”页可能不是完全静态的。例如，鼠标经过一个图像或一个 Flash 影片可以使静态页活动起来。但是，本系统所说的静态页面是指发送到浏览器时不进行修改的页面。

当 Web 服务器接收到对静态页的请求时处理流程如图 1.9 所示。

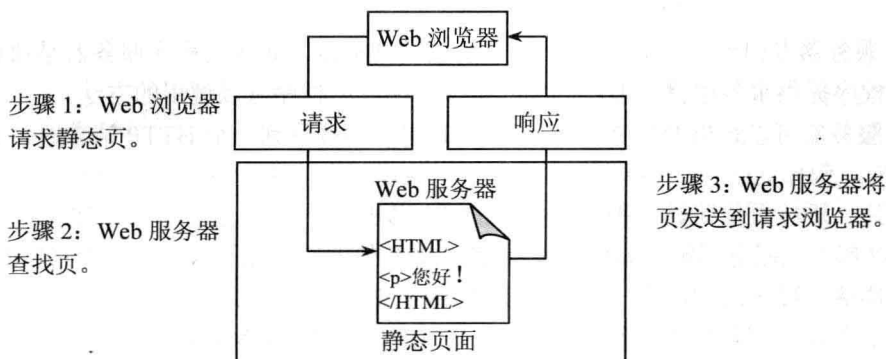


图 1.9 静态页面处理流程

2. 处理动态页面的工作原理

当 Web 服务器接收到对常规 Web 页的请求时，服务器将该页发送到请求浏览器，而不进行进一步的处理。当 Web 服务器接收到对动态页的请求时，它将做出不同的反应，将该页传递给一个负责完成页面的特殊软件扩展，这个特殊软件叫作应用程序服务器。

应用程序服务器读取页上的代码，根据代码中的指令完成页面，所得的结果将是一个静态页，应用程序服务器将该页传递回 Web 服务器，然后 Web 服务器将该页发送到浏览器。当该页到达时，浏览器得到的全部内容都是纯 HTML，处理流程如图 1.10 所示。读者可以通过浏览器查看源文件得到该 HTML 文件。

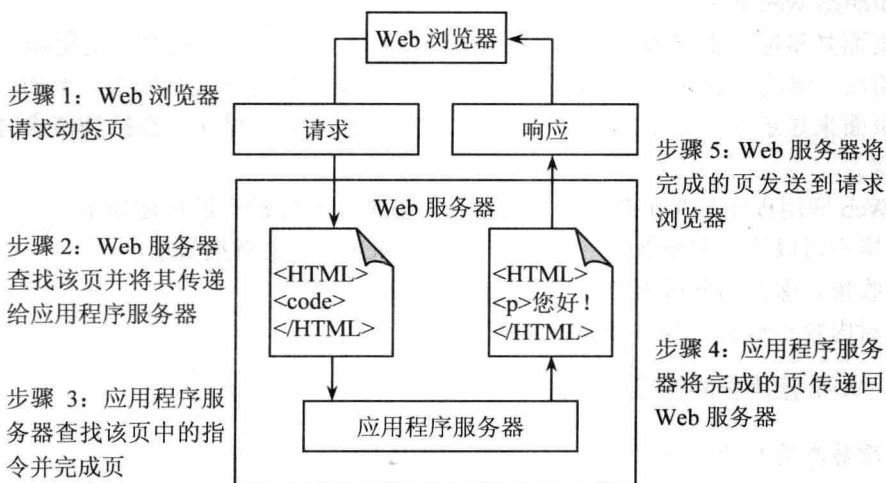


图 1.10 动态页面处理流程

JSP 页面的执行过程是通过 JSP 引擎把 JSP 标记符、JSP 页中的 Java 代码甚至连同静态 HTML 内容都转换为大块的 Java 代码，JSP 引擎和 JDK 在这里充当应用程序服务器的角色。这些代码块被 JSP 引擎组织到用户看不到的 Java Servlet 中去，然后 Servlet 自动把它们编译成 Java 字节码。这样，当网站的访问者请求一个 JSP 页时，在他不知道的情况下，一个已经生成的、预编译过的 Servlet 实际上将完成所有的工作。JSP 引擎将该页传回 Web 服务器，然后 Web 服务器将该页发送到浏览器。整个过程对用户而言是透明的。

1.4.2 Web 服务器和应用程序服务器

Web 服务器专门处理 HTTP 请求，一般通过浏览器。而应用程序服务器是通过很多协议来为应用程序提供事务逻辑处理，提供的是客户端应用程序可以调用的方法。

Web 服务器可以解析 HTTP 协议。当 Web 服务器接收到一个 HTTP 请求时，会返回一个 HTTP 响应，例如送回一个 HTML 页面。为了处理一个请求，Web 服务器可以响应一个静态页面或图片，进行页面跳转。把动态响应的产生委托给一些其他的程序，如 CGI 脚本、JSP 脚本、ASP 脚本、服务器端 JavaScript，或者一些其他的服务器端技术。无论它们的目的如何，这些服务器端的程序通常产生一个 HTML 的响应来让浏览器浏览。

Web 服务器的代理模型非常简单。当一个请求被送到 Web 服务器时，它只单纯地把请求传递给可以很好地处理请求的程序（服务器端脚本）。Web 服务器仅提供一个可以执行服务

器端程序和返回响应的环境,而不会超出职能范围。而服务器端程序通常具有事务处理、数据库连接和消息发送等功能。

应用程序服务器是一种软件框架,提供一个应用程序运行的环境。用于为应用程序提供安全、数据、事务支持、负载平衡、大型分布式系统管理等服务。

应用程序服务器的客户端可能会运行在一台 PC、一个 Web 服务器甚至是其他的应用程序服务器上。

另外,现在大多数应用程序服务器也包含了 Web 服务器,这就意味着可以把 Web 服务器当作是应用程序服务器的一个子集。虽然应用程序服务器包含了 Web 服务器的功能,但是开发者很少把应用程序服务器部署成这种功能。相反,如果需要,他们通常会把 Web 服务器和应用程序服务器独立配置。这种功能的分离有助于提高性能,而且给最佳产品的选取留有余地。

1.5 软件编程体系

软件编程体系主要分两种,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。现今市面上主流技术研发企业和用户对“B/S”和“C/S”技术谁优谁劣、谁代表技术潮流发展等问题争论不休。因此本节就此两项技术发展变化和应用前景做些探讨,供读者参考。

1.5.1 C/S 和 B/S

1. 什么是 C/S 结构

C/S 结构,简单地说就是传统意义上拥有客户端和服务器的网络软件或系统,即两层架构模型。传统开发环境有 VB 或 VC 等语言,随着 Java 体系以及 .NET 体系的发展,目前更流行后两种编程体系。通过在客户端安装应用程序,它可以充分利用两端硬件环境的优势,将任务合理分配到客户端和服务端来实现,服务器与本地用户数据交互方便,同时也方便控制客户端。不过这种机制要求每个用户都安装客户端,相比而言比较繁琐,同时对于跨平台的操作系统不能很好实现。

2. 什么是 B/S 结构

B/S 结构即浏览器和服务端结构。它是随着 Internet 技术的兴起,由 C/S 结构进行改进而形成的。在这种结构下,用户工作界面通过 WWW 浏览器来实现,极少部分事务逻辑在前端实现,主要事务逻辑在服务器端实现,形成所谓三层结构。这样就大大简化了客户端的负荷,减轻了系统维护与升级的成本和工作量,降低了用户的总体成本。

以目前的技术来看,局域网建立 B/S 结构的网络应用,并通过 Internet/Intranet 模式下数据库的应用,相对易于把握、成本也较低。它是一次到位的开发,能实现不同的人员,从不同的地点,以不同的接入方式(比如 LAN、WAN、Internet/Intranet 等)访问和操作共同的数据库;它能有效地保护数据平台和管理访问权限,服务器数据库也很安全。特别是在 Java 这样的跨平台语言出现之后,B/S 架构管理软件更显方便、快捷、高效。

1.5.2 C/S 和 B/S 之比较

C/S 和 B/S 是当今开发模式的两大主流技术。C/S 由美国 Borland 公司最早研发,B/S 由美国微软公司研发。目前,这两项技术已被世界各国所掌握,国内公司以 C/S 和 B/S 技术开发