

“九五”国家重点电子出版物规划项目·计算机知识普及系列



## 编程高手成长之路 5

Foundmental Programming with JavaServer Pages

# JSP 深入编程

北京希望电子出版社 总策划

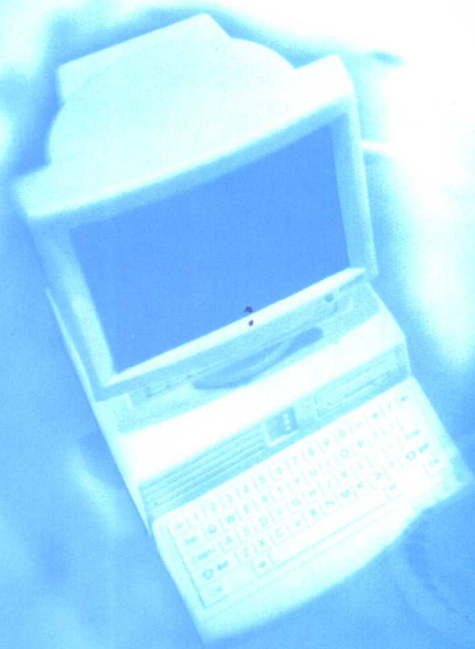
北京大学com工作室 创作

黄理 曹林有 张勇等 编著



北京希望电子出版社  
Beijing Hope Electronic Press  
www.bhp.com.cn

“九五”国家重点电子出版物规划项目 计算机知识普及系列



编程高手成长之路 5

Foundmental Programming with JavaServer Pages

# JSP 深入编程

北京希望电子出版社 总策划

北京大学 com 工作室 创作

黄理 曹林有 张勇等 编著



北京希望电子出版社  
Beijing Hope Electronic Press  
www.bhp.com.cn

## 内 容 简 介

这是一本面向初、中级编程人员的自学指导书，其重点放在 JSP 的基础知识与实际应用方面，全书由十四章、两个附录、一个参考文献组成，主要内容包括：JSP 简介、JSP 开发工具介绍、JSP 运行环境的配置、Java 语言基础、Java 与面向对象的程序设计、JSP 基本语法入门、JSP 内部对象详解（一）、JSP 内部对象详解（二）、数据库基础知识、JDBC——Java 中访问数据库的方法、Java 小服务器程序 Servlet、JSP/Servlet 综合开发实例、JSP 聊天程序——Jchat、JSP Forum 等。

本书由经验丰富的使用 JSP 组建网站的程序员编著，内文附有丰富的实例源码，供读者在自学习时参考。全书具有语言简明扼要，内容丰富，范例典型，理论与实践相结合的特点，不但是从事用 JSP 进行网站开发和设计的初、中级读者的自学指导书，同时也可作为社会网页设计或编程培训班的教材。

说明：与本书配套的面向中、高级用户的书《JSP 高级编程》也已正式出版，欢迎选购。

本版 CD 包括部分程序源代码。

系 列 书 名：“九五”国家重点电子出版物规划项目 希望计算机知识普及系列  
编程高手成长之路（5）

书 名：JSP 深入编程 Fundamental Programming with JavaServer Pages

总 策 划：北京希望电子出版社

文 本 著 者：北京大学com工作室 创作 黄理 曹林有 张勇等 编著

责 任 编 辑：马红华

C D 制 作 者：希望多媒体开发中心

C D 测 试 者：希望多媒体测试部

出 版、发 行 者：北京希望电子出版社

地 址：北京中关村大街 26 号，100080

网址：[www.bhp.com.cn](http://www.bhp.com.cn)

E-mail：[lwm@hope.com.cn](mailto:lwm@hope.com.cn)

电话：010-62562329,62541992,62637101,62637102,62633308,62633309（发行）

010-62613322-215（门市） 010-62613322-308（编辑部）

经 销：各地新华书店、软件连锁店

排 版：希望图书输出中心 周玉

CD 生 产 者：北京中新联光盘有限责任公司

文 本 印 刷 者：北京媛明印刷厂

开 本 / 规 格：787 毫米×1092 毫米 1/16 开本 31.5 印张 731 千字

版 次 / 印 次：2001 年 10 月第 1 版 2001 年 10 月第 1 次印刷

印 数：0001—5000 册

本 版 号：ISBN 7-980015-47-9

定 价：42.00 元（本版 CD）

说明：凡我社光盘配套图书若有缺页、倒页、脱页、自然破损，本社负责调换。

# 前 言

JSP (JavaServer Pages) 是目前十分流行的一种技术, 它主要运行于开发服务端的脚本程序和动态生成网站的内容。JSP 技术和目前同样流行的 ASP 技术、PHP 技术是同样性质的、同一层次的, 它们的作用是一样的。但是 JSP 技术与后面两种技术相比, 有着十分突出的优越性。关于 JSP 技术与 ASP 技术、PHP 技术的比较, 我们在书中另有论述, 这里只想强调一点: JSP 技术有 J2EE 平台支持, 发展前途不可限量。众所周知, J2EE 平台提供了 Java 企业应用编程接口 (Java Enterprise APIs), 为企业计算以及电子商务应用系统提供了有关的技术和强大的类库支持, J2EE 平台包含十几种技术, JSP 技术正是其中的一种核心技术。J2EE 的发展势头十分迅猛, 在可以预见的将来, Sun 的 J2EE 平台可能是唯一可以与微软的 .Net 构架相互抗衡的平台。在这个意义上说, 基于 J2EE 平台的 JSP 技术与基于 .Net 平台的 ASP+ 技术之争, 不正好就是 J2EE 平台与 .Net 平台之争的折射吗? 因此, JSP 技术以及它的基础——J2EE 技术十分值得我们去关注。在国外, 采用 JSP+J2EE 技术构架电子商务网站已经是大行其道了, 应用十分普遍。在国内, 这一项技术还是方兴未艾, 采用这一项技术架构的网站还不多, 不过大致的趋势已经出现了, 这真是一个令人兴奋的消息。为了帮助广大读者了解 JSP/J2EE 技术, 掌握 JSP/J2EE 技术, 我们编写了这《JSP 深入编程》和《JSP 高级编程》两本书, 前者侧重于 JSP 技术的基础知识与基本应用, 后者侧重于 JSP 技术和其他技术联合使用, 本书正是其中第一本。

本书按顺序讲述了以下知识点:

- JSP 技术的基本原理。
- JSP 技术在开发电子商务网站方面的应用。
- JSP 技术与 ASP, PHP 技术的比较。
- JSP/Java 运行环境的选择与配置。
- JSP/Java 开发工具的介绍与选择。
- Java 语言的基础语法知识。
- 面向对象编程技术在 Java 语言中的实现。
- JSP 的基本语法。
- JSP 的编译指令。
- JSP 的操作指令。
- JSP 程序的 XML 语法和 JSP 文档。
- JSP 的九个内部对象的详细用法。
- global.jsp 文件、System 类、Runtime 类的用法。
- 数据库的基本原理。
- 标准 SQL 语法简介。
- MySQL, SQL Server, Oracle, DB2 等流行数据库系统的分析比较。

- 发布 ODBC 数据源的方法。
- JDBC API 的详细介绍，如何利用 JDBC 技术访问数据库系统。
- Java Servlet 技术。
- Java Servlet API 完全介绍。
- Servlet+JSP 的开发模式。
- 开发实例：网上商店。
- 开发实例：聊天室——Jchat。
- 开发实例：BBS 论坛——JSP Forum。

顾名思义，本书是关于 JSP 技术的入门书籍，主要介绍 JSP 技术最基本、最核心、最容易掌握的方面，还有一些别的技术虽然不属于 JSP 技术的范畴，但又是必需的，所以我们也一并介绍，例如 Servlet 技术、JDBC 技术、数据库的知识等等。本书面向初学者，开始的时候难度很低，根本没有任何门槛，以后每一章都有所提升，循序渐进，由易到难，当你读完本书的 14 章以后，相信你对 JSP 技术/Java 技术已经有了相当程度的了解，可以独立编写 JSP•Java 程序了。即使你以前对编程一窍不通，完全也可以读懂本书的内容。

目前市场上 JSP 技术入门级别的书籍还是比较多的，本书的优点在于：

(1) 比较详细地介绍了目前最流行的、最新的 JSP/Java 的运行环境、开发环境，一共花了两章的篇幅，我们之所以这样安排，主要是考虑到本书的读者大部分都是初学者，在这关键的一步可能会遇到很多困难，如果不很好地解决这些困难，读者可能会丧失学习 JSP 技术的信心。

(2) 本书介绍的内容基于最新的技术规范，如 JSP 1.2 技术规范、Java Servlet API 2.3 规范，这可以使读者紧跟技术发展的潮流，在这知识爆炸的年代不至于落伍。

(3) 本书全面地介绍了 JSP 技术的方方面面，尤其是 JSP 的 9 个内部对象的用法、JSP 的编译指令、JSP 的操作指令，十分具有参考价值。

(4) 本书按照最新的 JSP 1.2 技术规范草案 2 的描述，对 JSP 程序的 XML 语法、JSP 文档、JSP 的 XML 视图做了全面描述。

(5) 本书全面介绍了 java.sql 包 7 个最主要的访问数据库的接口/类，帮助读者掌握这几个类/接口的用法，并利用它们编写访问数据库的程序。

(6) 本书详细描述了 Java Servlet API 各个接口/类的用法。

(7) 本书所举的 3 个例子：网上商店——Jchat, JSP Forum 十分具有代表性，综合程度很高，很好地覆盖了本书的各个主要知识点。尤其是最后一个例子——JSP Forum，其规模十分庞大，程序结构、系统结构相当复杂，我们把全部源代码都列出来，并帮助读者理解程序的代码。读者如果能够完全理解这 3 个大程序的实现原理，那么本书的知识就掌握得差不多了。

(8) 除了上述三个比较大的例子之外，本书还使用了大量的小例子，几乎每一个重要的知识点都有若干个小的例子或者程序片断，这对读者掌握这些知识点有很大的帮助。

本书的优点就介绍到这里，很显然，本书的优点远不止这 8 点，还有很多很多，读者只有在读过以后才能够体会出来。如上所述，我们向读者郑重推荐本书，希望本书能成为你学习 JSP 技术的入门书。

本书由北京大学 com 工作室组织编写。由于时间仓促，笔者的水平有限，书中的谬误一定很多。不足之处，请读者指正。


本书的成功出版，首先归功于本书的主要作者北大黄理同学，他深厚的计算机理论积累和丰富的实践经验才使得本书兼具理论指导及实务操作性，其工作的严谨态度以及出色的语言驾驭功底相信读者在阅读本书时自有体会。也感谢其他许许多多辛勤劳动与无私的帮助，轻易便可以列出很多：北大计算机系的李积善、水木清华(smth.org)的 javafancy、北大未名站的 javalover，还有 ROBBY,lz.lan,snowleaf 以及可爱的 Rainbow。

本书技术支持的联系方式：

com\_pku@263.net

http://162.105.106.162:8080

注：访问前，需要事先 mail 联系，以便启动服务器。





北京希望电子出版社  
Beijing Hope Electronic Press  
www.bhp.com.cn

# 隆重推出



CX-93471  
定价:20.00 元



CX-93486  
定价:20.00 元



CX-93445  
定价:20.00 元



CX-93444  
定价:20.00 元



CX-93485  
定价:20.00 元



CX-83499  
定价:28.00 元



CX-83466  
定价:35.00 元



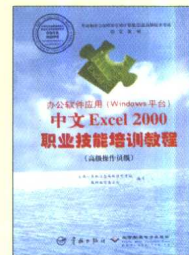
CX-83506  
定价:25.00 元



CX-2373  
定价:12.00 元



CX-83459  
定价:33.00 元



CX-2877  
定价:23.00 元



CX-3348  
定价:30.00 元



CX-83297  
定价:25.00 元



CX-3287  
定价:25.00 元



CX-3339  
定价:38.00 元



CX-83380  
定价:30.00 元



CX-83378  
定价:25.00 元



CX-3322  
定价:32.00 元



CX-3381  
定价:20.00 元



CX-3425  
定价:33.00 元



CX-82955  
定价:35.00 元



CX-82780  
定价:39.00 元



CX-83003  
定价:46.00 元



CX-83065  
定价:48.00 元



CX-3145  
定价:25.00 元



CX-2407  
定价:30.00 元



CX-2406  
定价:30.00 元



CX-2777  
定价:35.00 元



CX-2341  
定价:18.00 元



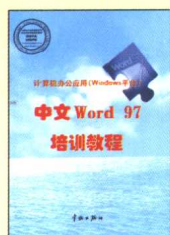
CX-2677  
定价:18.00 元



CX-3248  
定价:33.00 元

社址: 北京中关村大街 26 号 (黄庄路口东)  
电话: (010) 62562329 62541992

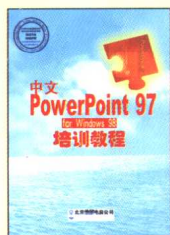
通讯: 北京中关村 083 信箱 (100080)  
传真: (010) 62579874 62633308



CX-2339  
定价:23.00元



CX-82868  
定价:28.00元



CX-2676  
定价:23.00元



CX-82876  
定价:35.00元



CX-82965  
定价:30.00元



CX-2971  
定价:38.00元



CX-3000  
定价:25.00元



CX-83172  
定价:25.00元



CX-3028  
定价:38.00元



CX-82680  
定价:20.00元



CX-3025  
定价:30.00元



CX-3064  
定价:25.00元



CX-3082  
定价:25.00元



CX-2340  
定价:20.00元



CX-3170  
定价:35.00元



CX-3171  
定价:33.00元



CX-83023  
定价:20.00元



CX-82935  
定价:28.00元



CX-82857  
定价:25.00元



CX-3027  
定价:29.00元



CX-2675  
定价:20.00元



CX-2778  
定价:22.00元



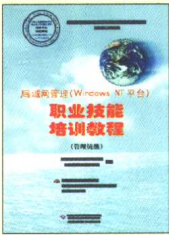
CX-2371  
定价:30.00元



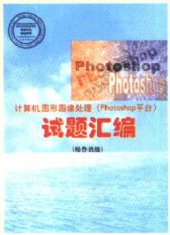
CX-3145  
定价:25.00元



CX-3001  
定价:39.00元



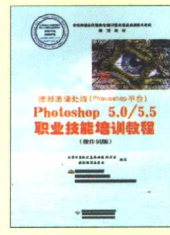
CX-3002  
定价:39.00元



CX-82721  
定价:35.00元



CX-82877  
定价:25.00元



CX-3026  
定价:35.00元



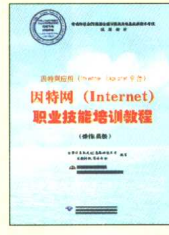
CX-3066  
定价:28.00元



CX-3158  
定价:30.00元



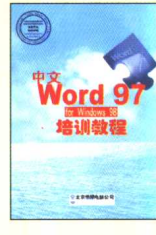
CX-3052  
定价:19.00元



CX-3076  
定价:23.00元



CX-3207  
定价:26.00元



CX-2674  
定价:23.00元

*Handwritten signature: WJS151/06*



# 目 录

第一章 JSP 简介 .....	1	4.3.2 运算符, 关键字 .....	85
1.1 动态网站概述 .....	1	4.3.3 流程控制 .....	86
1.2 JSP 技术简介 .....	3	4.3.4 异常处理 .....	90
1.3 JSP 与 ASP, PHP, ISAPI, CGI 的比较 .....	4	本章小结 .....	100
1.4 使用 JSP/J2EE 技术开发电子商务平台 .....	13	第五章 Java 与面向对象的程序设计 .....	101
1.5 如何快速掌握 JSP .....	17	5.1 面向对象的程序设计思想 .....	101
1.6 本书的使用说明 .....	19	5.1.1 基本原理 .....	101
本章小结 .....	20	5.1.2 封装 .....	102
第二章 JSP 开发工具介绍 .....	21	5.1.3 多态性 .....	103
2.1 Java/JSP 集成开发环境 .....	21	5.1.4 继承 .....	103
2.2 JSP/Web 程序开发工具 .....	26	5.2 Java 语言的面向对象特性 .....	104
2.3 JSP 程序编辑工具 .....	31	5.2.1 类 .....	104
本章小结 .....	32	5.2.2 抽象类和继承 .....	110
第三章 JSP 运行环境的配置 .....	33	5.2.3 重载 (多态) 和覆盖 .....	113
3.1 Windows 环境下 JSWDK1.0.1 的配置 .....	34	5.2.4 接口 .....	117
3.2 Windows 环境下 JRun 3.0 的配置 .....	40	5.3 Java 工具类库和 Enumeration 接口 .....	118
3.3 Windows 环境下 WebLogic 5.1 的配置 .....	46	本章小结 .....	122
3.4 Windows 环境下 Tomcat 的配置 .....	53	第六章 JSP 基本语法入门 .....	123
3.4.1 Tomcat 3.1.x 的配置方法 .....	53	6.1 JSP 标准语法 .....	123
3.4.2 Tomcat 3.2.x 的配置方法 .....	59	6.1.1 用 JSP 说 “Hello World!” .....	123
3.4.3 Tomcat 4.0 的配置方法 .....	63	6.1.2 JSP 注释 .....	124
3.5 Windows 环境下 Resin 的配置 .....	66	6.1.3 JSP 声明 .....	127
3.5.1 Resin 1.1.x 的配置方法 .....	66	6.1.4 JSP 表达式和程序段 .....	130
3.5.2 Resin 2.0 的配置方法 .....	69	6.1.5 结合使用 JSP 和 HTML .....	133
3.6 Windows 环境下 PWS+Resin .....	72	6.2 JSP 编译指令 .....	135
本章小结 .....	74	6.2.1 include 编译指令 .....	135
第四章 Java 语言基础 .....	75	6.2.2 page 编译指令 .....	138
4.1 Java 简介 .....	75	6.2.3 taglib 编译指令 .....	140
4.2 JDK 介绍 .....	76	6.3 JSP 操作指令 (Actions) .....	141
4.2.1 JDK 设置 .....	76	6.3.1 <jsp:forward>操作指令 .....	141
4.2.2 JDK 中主要命令行程序 .....	76	6.3.2 <jsp:param>操作指令 .....	142
4.2.3 Hello.java .....	79	6.3.3 <jsp:include>操作指令 .....	143
4.3 Java 语法 .....	80	6.3.4 <jsp:plugin>操作指令 .....	145
4.3.1 数据类型、常量、变量 .....	80	6.3.5 <jsp:useBean>操作指令 .....	149

6.3.6	<jsp:setProperty>操作指令	151
6.3.7	<jsp:getProperty>操作指令	151
6.4	JSP 文档 (JSP Documents)	151
6.4.1	对 JSP 程序使用 XML 语法 (XML Syntax)	152
6.4.2	JSP 文档语法	152
6.4.3	JSP 文档的 XML 视图	155
6.4.4	JSP 文档实例	158
	本章小结	161
<b>第七章</b>	<b>JSP 内部对象详解 (一)</b>	<b>162</b>
7.1	Out 对象	162
7.2	Request 对象	169
7.3	Response 对象	181
7.4	Exception 对象	185
7.5	Config 对象	187
7.6	Page 对象	189
	本章小结	190
<b>第八章</b>	<b>JSP 内部对象详解 (二)</b>	<b>191</b>
8.1	Application 对象	191
8.2	Session 对象	200
8.3	PageContext 对象	210
8.4	global.jsa	217
8.5	java.lang.System 类	219
8.6	java.lang.Runtime 类	221
	本章小结	223
<b>第九章</b>	<b>数据库基础知识</b>	<b>224</b>
9.1	数据库基本概念	224
9.2	当前流行的数据库系统	226
9.3	SQL 语言语法	230
9.4	数据库设计原则	235
9.5	使用 ODBC 数据源管理器发布 数据库服务	236
	本章小结	240
<b>第十章</b>	<b>JDBC——Java 中访问数据库的方法</b>	<b>241</b>
10.1	JDBC 技术简介	241
10.1.1	JDBC 技术原理	241
10.1.2	JDBC 技术的发展	243
10.1.3	ODBC、JDBC 和 SQLJ 的比较	244
10.2	java.sql 包介绍	246
10.2.1	JDBC 驱动程序管理器—— DriverManager 类	246
10.2.2	JDBC 与数据库的连接—— Connection 接口	248
10.2.3	执行普通 SQL 语句—— Statement 接口	250
10.2.4	执行预编译的 SQL 语句—— PreparedStatement 接口	253
10.2.5	执行数据库存储过程—— CallableStatement 接口	256
10.2.6	获取数据库的结构—— DatabaseMetaData 接口	258
10.2.7	数据库记录集—— ResultSet 接口	262
10.2.8	获取数据库记录集的结构 ——ResultSetMetaData 接口	266
10.3	JDBC 实例	266
10.3.1	获取数据库的结构	266
10.3.2	往数据库中插入资料	270
10.3.3	从数据库中检索资料	272
	本章小结	276
<b>第十一章</b>	<b>Java 小服务器程序 Servlet</b>	<b>277</b>
11.1	Servlet 基础知识	277
11.1.1	Servlet 程序	277
11.1.2	Servlet 技术与 CGI 技术 的比较	278
11.2	Servlet 编程	279
11.2.1	Servlet 程序的运行环境	279
11.2.2	使用 Servlet 程序说 Hello World!	281
11.2.3	Servlet 程序结构	282
11.2.4	Servlet 程序的生命周期	287
11.2.5	Servlet 程序中使用 JDBC 访问数据库	289
11.2.6	SHTML(HTML+Servlet)	293
11.2.7	javax.servlet 包	294
11.2.8	javax.servlet.http 包	299
11.3	Servlet 和 JSP	312
11.3.1	Servlet 有何缺陷	312

11.3.2 Servlet+JSP 的开发模式.....	312	14.1.1 JSP Forum 的结构.....	380
11.3.3 Servlet 程序和 JSP 程序的通信.....	313	14.1.2 开发平台准备.....	383
本章小结.....	313	14.1.3 数据库结构.....	384
<b>第十二章 JSP/Servlet 综合开发实例</b> .....	<b>314</b>	<b>14.2 通用模块分析</b> .....	<b>388</b>
12.1 开发准备.....	314	14.2.1 建立数据库连接 (database.jsp).....	388
12.1.1 平台选择.....	314	14.2.2 样式单文件(defaultcss.jsp)....	389
12.1.2 数据库结构.....	315	14.2.3 JavaScript 代码 (menuaction.jsp).....	390
12.2 程序分析(一).....	319	14.2.4 工具栏代码(toptable.jsp).....	392
12.2.1 indexServlet.java.....	319	14.2.5 菜单代码(menu.jsp).....	395
12.2.2 listServlet.java.....	321	<b>14.3 核心模块分析</b> .....	<b>399</b>
12.2.3 index.jsp.....	324	14.3.1 首页(index.jsp).....	399
12.2.4 login.jsp.....	324	14.3.2 登录程序(i_login.jsp & login.jsp).....	400
12.2.5 dologin.jsp.....	326	14.3.3 每日十大热门话题 (welcome.jsp).....	408
12.2.6 logout.jsp.....	328	14.3.4 板块浏览(list.jsp).....	413
12.2.7 main.jsp.....	329	14.3.5 文章浏览(view.jsp).....	424
12.3 程序分析(二).....	330	14.3.6 回复文章(i_reply.jsp & reply.jsp).....	434
12.3.1 cart.jsp.....	330	14.3.7 删除文章(i_delete.jsp & delete.jsp).....	443
12.3.2 list.jsp.....	332	14.3.8 修改文章(i_edit.jsp & edit.jsp).....	451
12.3.3 add.jsp.....	333	14.3.9 退出系统(i_logout.jsp).....	459
12.3.4 remove.jsp.....	335	<b>14.4 辅助模块分析</b> .....	<b>460</b>
12.3.5 clear.jsp.....	336	14.4.1 系统信息(sysmsg.jsp).....	460
12.3.6 order.jsp.....	337	14.4.2 检索用户(searchuser.jsp & i_searchuser.jsp).....	464
12.3.7 doOrder.jsp.....	339	14.4.3 用户信息(usermsg.jsp & viewuser.jsp).....	471
12.3.8 init.jsp.....	341	14.4.4 检索文章(search.jsp & i_search.jsp).....	478
本章小结.....	342	本章小结.....	485
<b>第十三章 JSP 聊天室程序——JChat</b> .....	<b>343</b>	<b>附录一 JSP/Servlet 资源列表</b> .....	<b>486</b>
13.1 JChat 的结构.....	343	<b>附录二 JSP 的中文问题及其解决方法</b> .....	<b>489</b>
13.2 初始化聊天室.....	345	<b>参考文献</b> .....	<b>493</b>
13.3 登录聊天室(login.jsp).....	347		
13.4 显示聊天室状态(fancy.jsp).....	353		
13.5 发送聊天信息(sendmsg.jsp).....	357		
13.6 显示聊天信息(showmsg.jsp).....	370		
13.7 离开聊天室(leavechat.jsp).....	373		
13.8 辅助功能.....	374		
13.9 管理聊天室.....	375		
本章小结.....	378		
<b>第十四章 JSP Forum</b> .....	<b>379</b>		
14.1 结构分析.....	379		

# 第一章 JSP 简介

## 1.1 动态网站概述

最初 WWW (World Wide Web) 只是作为在 Internet 上发布共享文件的一个工作平台, 随着 Internet 的迅速发展, 现在 WWW 不再只是作为简单的文档发表之用。实际上, 大部分企业或者社团或者政府机构的 Internet 网站应该更确切地被称为网络应用程序, 因为它们都需要进行复杂的逻辑处理来为客户端的访问者提供一个更吸引人的动态的消息, 或者是进行网上交易、网上结算等业务。

最早的网络服务器 (服务端) 和网络终端 (客户端) 的交互方式是这样的: 客户方使用浏览器向服务器发送一个请求, 服务器响应请求并返回一个普通的 HTML 文件, 在客户端的程序中 (如浏览器) 显示出一个格式化文本, 这种模式现在还经常使用。在这种模式下, 用户必须向服务器发送一个 Web 页面请求, 服务器接受这个请求并将该 Web 文件 (一般是 HTML 格式的文件) 送回到客户端的浏览器上, 因此客户可以使用浏览器显示并浏览这个 HTML 文件, 然后再向服务器发出新的请求, 服务器再次响应客户的请求。这种模式十分简单, 是一个获取静态文本、静态信息的好方法, 但是这种模式中, 用户和服务器之间的交互功能是十分有限的, 而且在这种模式下不能够实现大量信息的动态发布、动态更新。人们一直在寻找解决这个问题的方法。

在过去的几年中, Internet 作为世界范围内最重要的通讯网络之一蓬勃地发展起来了, World Wide Web 不但提供了强大的发布共享信息的功能, 而且确实把用户和厂商联系起来, 尽管用户和厂商之间可能隔着千山万水。不容置疑, Internet 已经成为了真正的信息高速公路, 而网络协议就成为了这个高速公路的“交通规则”。为了解决上面所提出来的问题, 新技术、新方法层出不穷, 在客户端, 浏览器不断添加对一些能为用户提供更高级的功能的脚本程序的支持, 例如 JavaScript 和 Visual Basic Script, 还有一些嵌入式的软件组件 (Plug-ins), 如 Java Applet 和 JavaBeans, ActiveX 控件等。通过这些技术, 我们可以实现基于客户端的动态技术。另一方面, 在服务器端, 各种各样的强大的服务端运行环境也开发出来了, 例如 CGI, ISAPI, JSP, ASP, ASP+, PHP 等, 综合使用这些技术, 使得大量信息的动态发布、动态更新成为可能, 客户端和服务端的交互操作也变得轻而易举了, 我们梦寐以求的理想似乎就要实现了。

随着互联网技术的不断发展、发布于网上的信息也随之几何级数的增长, 静态 HTML 页面的编写者越发显得力不从心, 如果全部的信息都要通过手工编辑, 然后发布到网上, 显然是无谓的人力和时间的浪费。而且静态页面只能千篇一律的给用户提供完全不变的信息, 无法提供人性化的服务。比如说, 没有办法根据不同的用户的好恶提供相应的新闻, 提供个性化的服务。新的形势下, 网站的开发者不得不考虑加入大量新的特性以吸引更多的点击量。现在的动态网站至少需要支持以下这些特性:

- 后台数据库支持。随着网上信息量的增加,以及企业和个人希望通过网络发布产品和个人信息的需求的增强,人们越来越需要一种能够通过 Web 程序访问服务端后台数据库的方式。这是静态页面根本不能实现的。
- 站点信息及时更新。以前,用户如果希望对传统静态页面的内容和信息进行更新或修改的话,只能采用逐一更改每个页面的方式。在互联网发展初期网上信息较少的时代,这种做法还是可以接受的。但是现在即使是网友们的个人站点也包含着各种各样的丰富内容,因此,如何及时、有效的更新页面信息也成为一个个急待解决的问题。
- 动态显示效果。所有的静态页面都是事先编写好的,是一成不变的,因此访问同一页面的用户看到的都将只是相同的内容,静态页面无法根据不同的用户做不同的页面显示。

为了克服静态页面的种种不足,人们将传统单机环境下的编程技术引入互联网络与 Web 技术相结合,从而形成新的网络编程技术。网络编程技术通过在传统的静态页面中加入各种程序和逻辑控制,在网络的客户端和服务端之间实现了动态和个性化的交流与互动。人们将这种使用网络编程技术创建的页面称为动态页面程序。

于是各种开发 Web 应用程序的编程语言工具应运而生,而且功能越来越强,使用上也越来越简单,Perl,ISAPI,ASP,ASP+,PHP,Servlet,JSP,XSP...,层出不穷,各有千秋。

网络编程技术一般采用脚本语言作为编程语言。脚本语言一般以某种现存编程语言为蓝本,对其进行简化,并且增强其在网络应用数据库访问上的功能,如 Perl 之于 C,ASP 之于 VBScript, JSP 之于 Java。熟悉的语法结构和编程风格,大大方便了网络应用程序的设计。

一般来说,脚本语言可以结合 HTML 语言共同使用。脚本语言可以嵌入到 HTML 标记之间,编程人员可以直接在脚本代码中加入 HTML 标记,或者在 HTML 标记之间加入脚本代码从而更好地实现页面控制,提供更加丰富的功能,见程序清单 1.1。

#### 程序清单 1.1

```
<!--
File Name:time.asp
Author:lea
Date:2001.3.19
Note:显示问候信息。
-->

<HTML>
<BODY>
<FONT COLOR="Green">
<% If Time < #12:00:00# And Time >= #00:00:00# Then %>
早上好!
<% ElseIf Time < #19:00:00# And Time >= #12:00:00# Then %>
下午好!
<% Else %>
晚上好!
```

```
<% End If %>
</BODY>
</HTML>
```

程序清单 1.1 (time.asp) 的功能是根据不同的时间段向访问者输出不同的问候信息。很简单的代码, 却实现了很人性化的功能。

不同的 Web 编程技术采用了与 HTML 不同的定界符, 将脚本代码包含在各自的定界符内, 以便区分在同一页面中的脚本代码和 HTML 代码。例如, JSP 和 ASP 代码使用的定界符为“<% %>”, PHP 使用的定界符为“<? ?>”或者“<?php ?>”; 并且, 使用不同脚本语言编写的 Web 动态程序都具有互不相同的后缀名。ASP 页面的后缀名为“.asp”, ASP+页面的后缀名为“.aspx”, PHP 页面的后缀名为“.php3”、“.php4”、“.php”、“.phtml”, JSP 页面的后缀名一般为“.jsp”, 有的服务器支持自定义文件名后缀。

动态页面被保存在 Web 服务器上, 当客户端用户向 Web 服务器发出访问动态页面的请求时, Web 服务器不会将程序的源代码直接返回客户端, 而是根据用户所请求的页面的后缀名确定该页面所使用的脚本语言, 然后把该页面提交给相应的语法解释引擎; 语法解释引擎扫描整个页面找到特定脚本语言的定界符, 并执行位于定界符内的脚本代码以实现不同的功能, 如访问数据库、发送电子邮件、执行算术或逻辑运算等, 并把执行结果返回 Web 服务器; 最后, Web 服务器把解释引擎的执行结果连同页面上的 HTML 内容以及各种客户端脚本一同传送到客户端。在客户端看到的内容是所请求的动态程序自动生成的 HTML 代码, 与传统静态页面并没有任何形式上的区别, 但是动态程序所生成的内容是动态变化的, 每次返回的结果都不一定相同。

## 1.2 JSP 技术简介

在以前, 开发 Internet 后台数据库应用程序是专业程序员做的事, 因为如同 ISAPI, CGI 类型的开发技术, 使用 C/C++, Perl, Pascal, Basic 作为开发语言, 上手十分困难, 程序代码难写难懂, 调试程序十分困难, 如果不是一个熟练的程序员, 并且没有无比的耐心, 那么是很难开发出一个完美的 Web 应用程序来的。后来, 代替 CGI, ISAPI 的技术方案纷纷出笼, 如 Microsoft 公司的 ASP (Active Server Pages), ASP+, C#; PHP (Personal HomePage Tools 或者又称为 Hypertext Preprocessor); Netscape 公司的 Liveware; Allaire 公司的 CFML; Sun 公司的 Java, JHTML, Servlet, JSP; Apache 组织的 XSP, Cocoon 等, JSP/Servlet 技术就是它们当中的佼佼者, 在 Web 开发领域一枝独秀, 独领风骚。

JSP 技术为创建显示动态生成内容的 Web 页面提供了一个简捷而快速的方法。JSP 技术的设计目的是使得构造基于 Internet 的应用程序更加容易和快捷, 而这些应用程序能够在各种 Web 服务器、应用程序服务器、客户端浏览器下顺利运行。

在介绍 JSP 技术以前, 有必要简单介绍一下 Servlet 技术, 因为 Servlet 技术就是 JSP 技术的前身, JSP 技术其实就是 Servlet 技术的一个高层封装。

Java Servlet 程序又称为 Java 服务器小程序, 它本质上就是一个 Java 类, 运行于服务器中, 它接受客户端的请求, 自动生成 HTML 代码向客户端输出。我们编写好 Servlet 程序后, 使用 javac.exe 程序将它编译为 class 文件, 保存于服务器上的特定目录下。当服务器启动时,

会将 Servlet 程序的二进制代码载入内存，等候客户端的请求；当客户端的请求到达服务器，服务器会自动产生一个新的服务线程，执行相应的 Servlet 程序，并将结果返回客户端。如果有新的请求到达，服务器也是产生一个新的服务线程来响应请求。Servlet 程序的优点在于运行效率十分高，系统资源消耗比较少，因为服务器对于每个新的请求只需要产生一个新的服务线程就可以了。Servlet 程序的缺点是不能够直接使用 HTML 标记，必须使用 Java 语言输出所有的数据，包括 HTML 标记在内，编写起来相对比较简单，不能够使用可视化工具减轻劳动量，而且 Servlet 程序所产生页面样式很难控制。如果我们需要改变页面的样式，哪怕是只改变某个标题的颜色，也必须重新编译程序，然后将其发布到服务器中。

为了弥补 Servlet 程序的这些缺点，Sun 公司在 Servlet 技术的基础上开发了 JSP 技术，JSP 技术吸取了 Servlet 技术的长处，摒弃了 Servlet 技术的短处，JSP 程序其实就是在 HTML 代码中嵌入 Java 代码段，这些 Java 代码段可以完成各种各样的功能，例如访问数据库系统、发送电子邮件、显示在线用户名单等。JSP 程序可以使用可视化工具做前期开发，以便减小开发工作的劳动量。在开发 JSP 程序的时候，一般是首先利用 Dreamweaver, UltraDev 等工具设计网页的外观、样式，然后再在 HTML 文件中手工加入 Java 程序段。编写好 JSP 程序后，不需要编译它，我们只需要把它存放到服务器的特定目录下面就可以了，当服务器接到对 JSP 程序的请求时，它会首先把 JSP 程序发送到一个语法分析器中，这个语法分析器将会把这个 JSP 程序翻译为一个 Java 程序文件，然后调用 javac.exe 程序（或者其他可用的 Java 语言编译器）将这个 Java 程序文件编译为 Servlet 类，即一个标准的 Java class 文件。这时，服务器的 JSP 引擎（即服务器上的 Java 运行环境）将把这个 class 载入内存，运行它，把结果送往客户端，客户端的浏览器上出现的就是这个 JSP 程序的运行结果。当第二次请求这个 JSP 程序时，由于它已经被编译为字节码形式的 class 文件，所以 JSP 引擎就直接运行这个 class 文件，而不需要再次编译它，除非 JSP 程序被改动或者是服务器关闭后又重新启动了。当有多个用户请求同一个 JSP 程序时，服务器会产生多个线程而不是多个进程以响应客户的请求。因此，JSP 程序的运行效率是相当高的。

说到这里，读者对于 JSP 程序的运行原理可能已经有所了解，但是市场上有很多动态网站的解决方案，JSP 技术与它们相比，到底孰优孰劣呢？

### 1.3 JSP 与 ASP, PHP, ISAPI, CGI 的比较

当前世界上，流行的开发 Internet 应用程序的解决方法大概有以下几种方案：ASP, PHP, JSP, ISAPI, Perl 等，这些开发技术各有其优缺点。但是按照目前的情形来看，ASP/ASP+ 技术以及 JSP/Servlet 技术要成为未来开发的主流技术，关于 ASP/ASP+ 技术的知识，不是本书讨论的重点，本书主要讨论 JSP/Servlet 技术。那么我们为什么要选用 JSP 技术作为我们的电子商务一揽子解决方案呢？

总的来说，JSP 技术的优点有二，一是简单，二是强大。为什么说 JSP 技术简单呢？为了给读者一个形象的概念，我们先举一个简单的例子来说明用 JSP 开发 Web 程序是如此的简单方便，至于 ASP, PHP, JSP 在开发电子商务平台上的优缺点，我们在后面将作一番详细的比较。

在讲授计算机语言的教材中，第一个入门的程序往往就是著名的“Hello World!”程序，即输出一行类似于“Hello World”的文字，我们也未能免俗，下面程序清单 1.2 至程序清单 1.6 分别用 JSP, ASP, PHP, Perl, ISAPI 五种解决方法输出包含“Hello World!”这一行文字的 Web 页面。我们借此比较一下用这五种技术开发 Web 动态网站的难易程度。

#### 程序清单 1.2

```
<%--  
File Name:hello.jsp  
Author:fancy  
Date:2001.3.19  
Note:print hello world  
--%>  
  
<%@ page language="java" %>  
<%  
out.print("hello world!");  
%>
```

程序清单 1.2 (hello.jsp) 程序的运行效果如图 1.1 所示。

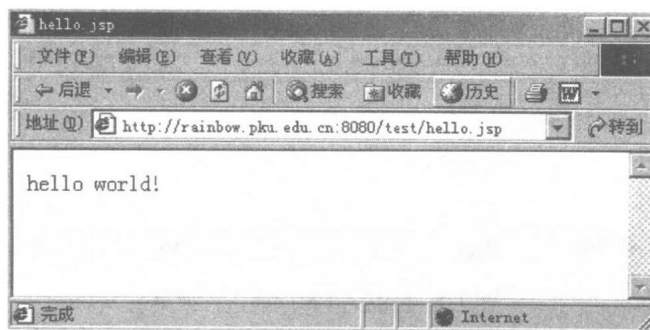


图 1.1 hello.jsp 程序的运行效果

#### 程序清单 1.3

```
<%--  
File Name:hello.jsp  
Author:fancy  
Date:2001.3.19  
Note:print hello world  
--%>  
  
<%@ language="vbscript" %>  
<%  
response.write "hello world"  
%>
```

程序清单 1.3 的运行效果如图 1.2 所示。



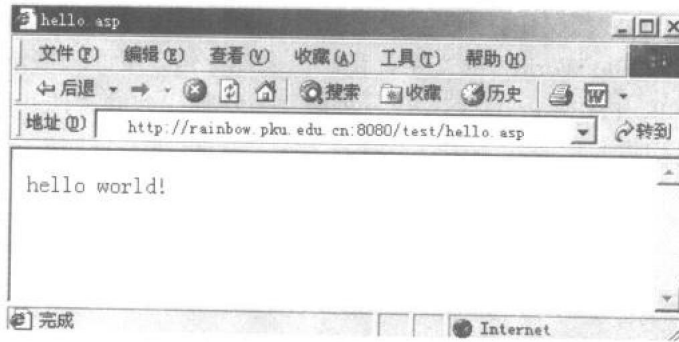


图 1.2 hello.asp 程序的运行效果

程序清单 1.4 hello.cgi

```
#!/usr/bin/perl  
print "Hello. World...\n";
```

程序清单 1.4 的运行效果如图 1.3 所示。

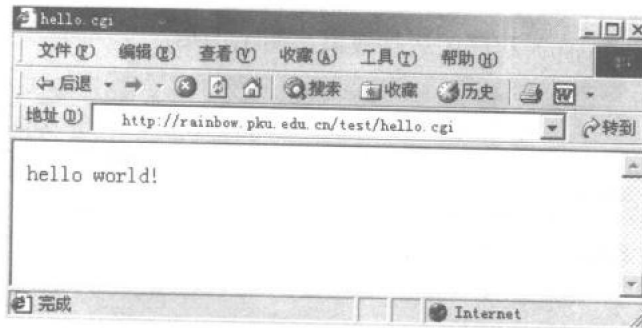


图 1.3 hello.cgi 程序的运行效果

程序清单 1.5

```
<!--  
File Name:hello.php3  
Author:fancy  
Date:2001.3.19  
Note:print hello world  
--!>  
  
<?php  
echo "hello world";  
?>
```

程序清单 1.5 (hello.php3) 的运行效果如图 1.4 所示。