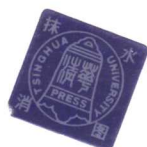


数据库应用开发技术丛书

JSP

数据库高级教程

刘 彬 编著



清华大学出版社

数据库应用开发技术丛书

JSP 数据库高级教程

刘 彬 编著

清华大学出版社

北 京

内 容 简 介

JSP(Java Server Pages)是由 Sun Microsystems 公司倡导的、许多公司参与一起建立的一种动态网页技术标准,它可以实现网页中的动态部分和静态的 HTML 相分离,可以无缝地运行在 Unix, Linux, Windows 等操作平台上,从而建立安全、跨平台的动态网站。

本书详细地讲解了 JSP 数据库编程技术,全书共分 19 章。第 1 章~第 15 章主要讲述了 JSP 语言基础、JSP 程序的配置与发布、数据库基础、数据库的访问和连接、高级数据库技术、数据文件操作高级应用、二进制数据的处理、触发器、存储过程、JSP 与 Servlet 高级开发应用、JSP 与 JavaBean 高级开发应用、EJB 开发、在 JSP 中使用 XML 交换数据、图片处理及显示、基于 Session 的身份验证等内容,第 16 章~第 19 章介绍了留言板、邮件收发系统、BBS 论坛系统和在线聊天室系统,从而详细地讲解了 JSP 数据库编程技巧。

本书语言简洁,内容丰富,可作为高等院校相关专业的教材和辅导用书,也可作为 JSP 开发人员和广大用户的自学或参考用书。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

JSP 数据库高级教程/刘彬 编著. —北京:清华大学出版社,2006.3

(数据库应用开发技术丛书)

ISBN 7-302-12373-X

I. J… II. 刘… III. ①Java 语言—程序设计—教材 ②因特网—程序设计—教材 IV. ①TP312
②TP393.4

中国版本图书馆 CIP 数据核字(2006)第 002731 号

出 版 者:清华大学出版社 地 址:北京清华大学学研大厦
http://www.tup.com.cn 邮 编:100084
社 总 机:010-62770175 客户服务:010-62776969

组稿编辑:王 定

文稿编辑:鲍 芳

封面设计:久久度文化

版式设计:康 博

印 刷 者:北京密云胶印厂

装 订 者:北京鑫海金澳胶印有限公司

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:27.25 字数:629 千字

版 次:2006 年 3 月第 1 版 2006 年 3 月第 1 次印刷

书 号:ISBN 7-302-12373-X/TP·7932

印 数:1~4000

定 价:38.80 元

前 言

Java Server Page(简称为 JSP)是由 Sun 公司于 1999 年 6 月推出的一种基于 Java Servlet 以及整个 Java 体系的 Web 开发技术,它集成了 Java 面向对象的程序语言能力与跨平台的优势,并且与超文本标记语言紧密结合,与传统的 CGI 程序相比,JSP 程序不仅编写、执行更容易,而且大幅提高了系统的执行性能。

本书是介绍 JSP 数据库开发的高级教程,共分为 19 章,涵盖了 JSP 编程的方方面面,不仅包括 JSP 的基本语法、JSP 开发环境的配置、JSP 的开发工具等内容,还包括数据库编程基础、JSP 的文件操作、JavaBean 技术、Servlet 编程技术、EJB 开发、XML 应用等高级应用。此外,本书还通过几个实例介绍了 JSP 数据库开发的完整过程,使读者在掌握 JSP 基本开发方法和技巧后,能够将其应用到实践中,从而领悟 JSP 的精髓。本书章节内容安排如下。

第 1 章主要介绍了 JSP 语言的一些基本语法和 JSP 应用开发体系,包括 JSP 中的脚本元素、指令、动作,JBUILDER 2005 的开发环境、Sun One Studio 开发环境的设置与应用,并介绍了 JSP 的一些安全性问题。

第 2 章主要介绍了 JSP 的环境配置和 JSP 的内置对象,包括 Tomcat 开发环境的配置、WebLogic 的配置、部署文件的配置、JSP 的内置对象、提高 JSP 程序的性能、JSP 程序的发布等内容。

第 3 章主要介绍了 JSP 访问数据库的基本知识,包括创建数据库、创建表和视图、数据库的备份和恢复、常用 SQL 语句的使用、数据库的安全问题等内容。

第 4 章主要介绍了访问和连接数据库的相关知识,包括 JDBC 基础知识、通过 JDBC 连接常用数据库,使用连接池连接数据库、数据库的一些常用操作内容。

第 5 章主要介绍了一些高级数据库技术,包括数据排序、主/从表处理、事务处理、数据库视图处理、处理多表数据、分页显示记录和加快数据库的访问方法等内容。

第 6 章主要介绍了数据文件操作高级应用,包括数据流、File 类和读写文件、文件浏览等几个文件操作的实例,讲解了文件操作的应用。

第 7 章主要介绍了二进制数据的处理操作。通过普通二进制数据处理、图像文件处理、声音和视频文件处理等几个实例讲解了其操作方法。

第 8 章主要介绍了触发器的应用,包括创建、查看、修改、删除触发器和游标的使用方法等内容,并通过一个触发器应用实例来讲解触发器的使用方法。

第 9 章主要介绍了 JSP 的存储过程,包括系统存储过程、用户存储过程、自动执行的存储过程、扩展存储过程和用户定义函数等内容,并通过实例来介绍 JSP 中如何调用 SQL Server 2000 中的用户存储过程。

第 10 章主要介绍了 JSP 与 Servlet 高级开发应用,包括 Servlet 的实现原理、页面间共享信息、Servlet 中的对象、通过 Servlet 访问数据库、通过 Servlet 实现多层数据库应用程序和 Sevlet 的性能和安全性等内容。

第 11 章主要介绍了 JSP 与 JavaBeans 高级开发应用,包括开发 JavaBeans 组件、在页面中使用 JavaBeans 组件和使用通用数据库 Bean 访问数据库等内容,并通过一个 JSP 页面内调用 JavaBean 的实例介绍 JavaBean 的具体的使用方法。

第 12 章主要介绍了 Session Bean 与 Entity Bean 的相关知识,包括开发 EJB 组件的方法、部署和访问 EJB、通过数据库访问 Beans 和开发数据库组件等内容。

第 13 章主要介绍了 JSP 中 XML 的应用,包括 XML 的两种显示格式、几种解析 XML 的方法、将 JSP 页面数据写入 XML 文档、通过 Servlet 访问 XML 数据和通过 JavaBean 访问 XML 数据等内容,并通过一个 JSP+XML 动态网站开发实例介绍了在 JSP 中使用 XML 交换数据。

第 14~19 章通过几个综合实例详细地介绍了使用 JSP 开发 Web 应用程序的步骤和过程。

本书按照由浅入深的原则,对不同的知识进行了详细的讲解,使读者在学习的过程中能更深入地理解。另外,读者在阅读本书时,可通过 <http://www.tupwk.com.cn/database> 下载本书代码和反馈意见,或浏览和下载《数据库应用开发技术丛书》其他图书信息。

本书由刘彬执笔编写,此外,顾晔、孟士强、黄亦飞、赵昌剑、于陈华、王从明、葛爱琳、宋婷婷、张清欢、潘媛、欧阳志强、蒋忠、沈学兵、韩建国、杨树平等同志在整理材料方面给予了编者很大的帮助,在此,编者对他们表示衷心的感谢。

本书内容翔实、结构紧凑、条理清晰、覆盖知识点全面。但由于本书涉及的的内容丰富,加之篇幅、时间所限,书中难免有不足之处,敬请广大读者提出宝贵意见。

编 者

目 录

第 1 章 JSP 语言基础	1
1.1 JSP 语言概述	1
1.2 JSP 中的脚本元素	3
1.2.1 声明	3
1.2.2 表达式	3
1.2.3 脚本代码	3
1.3 JSP 中的指令	4
1.3.1 Page 指令	4
1.3.2 include 指令	7
1.3.3 taglib 指令	8
1.4 JSP 中的动作	8
1.4.1 <jsp:useBean>动作	9
1.4.2 <jsp:include>动作	10
1.4.3 <jsp:setProperty> 动作	10
1.4.4 <jsp:getProperty>动作	12
1.4.5 <jsp:forward>动作	13
1.4.6 <jsp:plugin>动作	14
1.5 动态 Web 页面设计	15
1.6 JSP 的开发工具	17
1.6.1 JBuilder 开发环境	17
1.6.2 Sun ONE Studio 开发环境	25
1.7 创建一个简单的 JSP 应用	29
1.8 JSP 的安全性	31
1.8.1 JSP 安全性的实现方法	31
1.8.2 源代码暴露问题	32
1.8.3 远程程序执行问题和其他问题	33
1.9 小结	34
第 2 章 JSP 程序的配置和发布	35
2.1 Tomcat 服务器的配置	35
2.2 WebLogic 配置	37
2.3 context.xml 文件配置	42
2.3.1 Context 元素的通用属性和附加属性	42
2.3.2 配置 context.xml 文件	43

2.4	web.xml 文件配置	45
2.5	JSP 的内置对象	49
2.5.1	Request 对象	49
2.5.2	Response 对象	52
2.5.3	Session 对象	54
2.5.4	其他内置对象	56
2.6	提高 JSP 程序的性能	58
2.7	JSP 程序的发布	59
2.8	小结	61
第 3 章	数据库基础	62
3.1	Oracle 数据库	62
3.2	MySQL 数据库	63
3.3	SQL Server 数据库	64
3.3.1	SQL Server 2000 的特性	64
3.3.2	启动 SQL Server 服务	65
3.4	创建数据库	65
3.5	创建表和视图	67
3.5.1	创建数据表	67
3.5.2	创建视图	69
3.6	数据库的备份和恢复	71
3.6.1	数据库的备份	71
3.6.2	数据库的恢复	75
3.7	高级 SQL 语句	76
3.7.1	SQL 查询语句	76
3.7.2	删除语句	77
3.7.3	插入语句	77
3.7.4	更新语句	78
3.7.5	索引	79
3.7.6	流程控制语句和变量	81
3.7.7	系统函数	85
3.8	数据库的安全	86
3.9	数据库设计实例	87
3.9.1	数据库逻辑结构分析	88
3.9.2	表和字段分析	88
3.9.3	创建数据库和数据表	90
3.10	小结	90

第 4 章 数据库的访问和连接	91
4.1 JDBC 基础	91
4.1.1 JDBC 体系结构	91
4.1.2 JDBC 接口	92
4.2 MVC 设计模式	93
4.3 数据库连接实例	94
4.3.1 Oracle 数据库	95
4.3.2 SQL Server 2000 数据库	97
4.3.3 DB2 数据库	98
4.3.4 Informix 数据库	99
4.3.5 Sybase 数据库	100
4.3.6 MySQL 数据库	100
4.3.7 PostgreSQL 数据库	101
4.4 使用连接池连接数据库	101
4.4.1 连接池管理策略	102
4.4.2 连接池的使用方法	103
4.5 数据遍历与定位	107
4.6 数据查找	109
4.7 数据修改和更新	113
4.8 数据删除	118
4.9 数据过滤	122
4.10 小结	124
第 5 章 高级数据库技术	125
5.1 数据排序	125
5.2 主/从表处理	127
5.3 事务处理	129
5.4 数据库视图处理	132
5.5 处理多表数据	134
5.6 分页显示记录	136
5.7 加快数据库的访问方法	139
5.8 小结	141
第 6 章 数据文件操作高级应用	142
6.1 文件存储数据概述	142
6.2 数据流和 File 类	142
6.2.1 数据流	142
6.2.2 File 类	143
6.3 读写文本文件	144

6.4	读写二进制文件	146
6.4.1	读取二进制文件	146
6.4.2	写入二进制文件	147
6.5	访问 Excel 电子表格中的数据	148
6.6	文件上传的实现	150
6.7	文件下载的实现	154
6.8	文件的浏览	154
6.9	创建和删除目录	156
6.10	小结	157
第 7 章	二进制数据的处理	158
7.1	二进制数据概述	158
7.2	普通二进制数据处理	158
7.2.1	将二进制数据写入到数据库中	158
7.2.2	读取数据库中的二进制数据	160
7.3	图像文件处理	162
7.3.1	将图像以二进制数据格式存储到数据库中	162
7.3.2	从数据库中读出图像	165
7.4	声音文件处理	167
7.4.1	将声音文件存储到数据库中	167
7.4.2	从数据库中读取声音文件	168
7.5	视频文件处理	171
7.5.1	将视频文件存储到数据库中	171
7.5.2	从数据库中读取视频文件	172
7.6	小结	174
第 8 章	触发器	175
8.1	触发器简介	175
8.2	创建触发器	176
8.2.1	使用 SQL 语句创建触发器	176
8.2.2	使用企业管理器创建触发器	178
8.3	查看触发器	180
8.4	修改触发器	182
8.5	删除触发器	183
8.6	游标使用技术	184
8.6.1	创建游标	184
8.6.2	打开游标	185
8.6.3	使用游标	185
8.6.4	关闭游标	186

8.6.5 删除游标	186
8.7 触发器应用实例	187
8.7.1 创建触发器	187
8.7.2 编写 JSP 测试页面	187
8.8 小结	193
第 9 章 存储过程	194
9.1 存储过程简介	194
9.2 系统存储过程	194
9.3 用户存储过程	196
9.3.1 创建存储过程	196
9.3.2 执行存储过程	197
9.4 自动执行的存储过程	198
9.5 扩展存储过程	199
9.6 用户定义函数	199
9.6.1 创建用户定义函数	199
9.6.2 调用用户定义函数	200
9.7 存储过程应用实例	202
9.7.1 创建存储过程	202
9.7.2 创建 JSP 页面	203
9.8 小结	204
第 10 章 JSP 与 Servlet 高级开发应用	205
10.1 Servlet 的实现原理	205
10.2 在 Servlet 和 JSP 页面间共享信息	205
10.3 Servlet 中的对象	207
10.4 在 JSP 中通过 Servlet 访问数据库	210
10.5 通过 Servlet 实现多层数据库应用程序	213
10.5.1 数据层	214
10.5.2 应用层	214
10.5.3 表示层	214
10.5.4 多层应用程序的优点	214
10.6 Servlet 的性能和安全性	214
10.7 小结	216
第 11 章 JSP 与 JavaBeans 高级开发应用	217
11.1 JavaBeans 概述	217
11.2 Servlet 与 JavaBeans 比较	218
11.3 开发 JavaBeans 组件	219
11.4 在页面中使用 JavaBeans 组件	219

11.4.1	<jsp: useBean>	220
11.4.2	<jsp: setProperty>	220
11.4.3	<jsp: getProperty>	221
11.5	使用通用数据库 Bean 访问数据库	221
11.5.1	编写 JavaBean 文件	221
11.5.2	编写 JSP 文件	223
11.6	JavaBeans 应用实例	224
11.6.1	编写 JavaBean 文件	225
11.6.2	创建主页面	226
11.6.3	编写 JSP 页面	227
11.7	小结	228
第 12 章	EJB 开发	229
12.1	EJB 和 JavaBeans	229
12.2	Session Bean	230
12.2.1	创建 Session Bean	233
12.2.2	删除一个 Session Bean	233
12.2.3	钝化与活化	234
12.3	Entity Bean	234
12.3.1	Entity Bean 和 Session Bean 的区别	234
12.3.2	CMP 模式和 BMP 模式	235
12.3.3	Entity Bean 的创建与删除	235
12.4	开发 EJB 组件	239
12.4.1	编写主接口	239
12.4.2	编写远程接口	240
12.4.3	编写 Bean 类实现程序	240
12.4.4	编写部署文件	241
12.4.5	编译和打包	243
12.4.6	编译生成容器代码	243
12.4.7	编写客户端测试程序代码	244
12.5	部署和访问 EJB	244
12.5.1	BEA WebLogic	244
12.5.2	IBM WebSphere	245
12.5.3	JBoss	245
12.5.4	客户端访问 EJB	245
12.6	自定义数据库访问 Beans	247
12.7	开发特定功能的数据库组件	251
12.8	开发分布式多层 Web 程序	256
12.9	小结	257

第 13 章 在 JSP 中使用 XML 交换数据	258
13.1 XML 基本语法	258
13.2 XML 的两种显示格式	259
13.2.1 使用 CSS 显示 XML 文档	259
13.2.2 使用 XSL 显示 XML 文档	262
13.3 XML 文件的访问接口	264
13.3.1 DOM 解析接口	264
13.3.2 SAX 解析接口	266
13.3.3 JDOM 解析接口	272
13.4 将 JSP 页面数据写入 XML 文档	275
13.5 通过 Servlet 访问 XML 数据	276
13.6 通过 JavaBeans 访问 XML 数据	277
13.7 JSP+XML 动态网站开发实例	279
13.7.1 程序起始界面结构	279
13.7.2 用户管理	280
13.7.3 留言管理	298
13.8 小结	310
第 14 章 图片处理及显示	311
14.1 概述	311
14.2 问题分析	311
14.2.1 系统功能分析	311
14.2.2 数据库建立	312
14.3 程序实现	314
14.3.1 公用组件	314
14.3.2 浏览图片	315
14.3.3 发表评论	318
14.3.4 管理员登录	320
14.3.5 上传图片	321
14.3.6 删除图片	322
14.3.7 其他功能	325
14.4 运行发布	325
14.5 小结	326
第 15 章 基于 Session 的身份验证	327
15.1 概述	327
15.2 问题分析	327
15.2.1 系统功能分析	327
15.2.2 数据库建立	328

15.3	程序实现	329
15.3.1	数据库访问	329
15.3.2	用户注册	332
15.3.3	用户登录	338
15.3.4	管理员登录	340
15.4	运行与发布	341
15.5	小结	343
第 16 章	留言板	344
16.1	概述	344
16.2	问题分析	344
16.2.1	系统功能分析	345
16.2.2	建立数据库	345
16.3	程序实现	346
16.3.1	查看留言	346
16.3.2	发表留言	350
16.3.3	删除留言	354
16.3.4	管理员设置	355
16.4	运行与发布	358
16.5	小结	358
第 17 章	邮件收发系统	359
17.1	概述	359
17.2	问题分析	360
17.2.1	系统功能分析	360
17.2.2	数据库建立	360
17.3	程序实现	361
17.3.1	通过 SMTP 收发邮件	361
17.3.2	为邮件添加附件	367
17.3.3	下载邮件附件	368
17.4	运行与发布	370
17.5	小结	370
第 18 章	BBS 论坛系统	371
18.1	概述	371
18.2	问题分析	372
18.2.1	系统功能分析	372
18.2.2	数据库建立	373
18.3	程序实现	375
18.3.1	信息浏览	375

18.3.2	信息发布	383
18.3.3	信息回复	385
18.3.4	信息管理	387
18.4	运行与发布	391
18.5	小结	391
第 19 章	在线聊天室	392
19.1	概述	392
19.2	问题分析	392
19.2.1	系统功能分析	393
19.2.2	数据库建立	393
19.3	程序实现	394
19.3.1	新用户注册	394
19.3.2	用户登录	401
19.3.3	进行聊天	403
19.3.4	用户退出	410
19.4	运行与发布	411
19.5	小结	411
附录	JSP 语法参考	412

第1章 JSP语言基础

JSP(Java Server Pages)是由 Sun Microsystems 公司倡导的、许多公司参与一起建立的一种动态网页技术标准，是一种基于 Java Servlet 的 Web 开发技术。JSP 允许在 HTML 页面内嵌入 Java 代码。由于 Java 语言是一种简单易用、完全面向对象、具有平台无关性且安全可靠的面向 Internet 的开发工具，所以 JSP 是一种功能强大、可以实现跨平台操作的动态网页开发技术。它使得构造基于 Web 的应用程序更加容易和快捷，而这些应用程序能够与各种 Web 服务器、应用服务器、浏览器和开发工具共同工作，目前已经成为开发动态网页的主流技术之一。

1.1 JSP 语言概述

JSP(Java Server Pages)是由 Sun Microsystems 公司倡导的、许多公司参与一起建立的一种动态网页技术标准。JSP 技术是用 Java 语言作为脚本语言的，JSP 网页为整个服务器端的 JAVA 库单元提供了一个接口来服务于 HTTP 的应用程序。

在传统的网页 HTML 文件(*.htm, *.html)中加入 Java 程序片段(Scriptlet)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 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 e-mail 等等，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最低，可以实现无 Plugin，无 ActiveX，无 Java Applet，甚至无 Frame。

作为一种基于文本的、以显示为中心的开发技术，JSP 以 Java Servlet 为基础，提供了 Java Servlet 的所有优点，并且当与一个 Java Beans 类结合在一起时，提供了一种使内容和显示逻辑分开的简单方式。分开内容和显示逻辑使得更新页面外观的人员不必懂得 Java 代码，而更新 Java Beans 类的人员也不必是设计网页的行家里手，就可以用带有 Java Beans 类的 JSP 页面来定义 Web 模板，以建立一个由具有相似的外观的页面组成的网站。数据由 Java Beans 类提供，这样在模板中就没有 Java 代码，这就意味着这些模板可以由一个 HTML 编写人员来维护。当然，也可以利用 Java Servlet 来控制网站的逻辑，通过 Java Servlet 调用 JSP 文件的方式来将网站的逻辑和内容分离。

一般来说，在实际的 JSP 引擎中，JSP 页面在执行时是编译式，而不是解释式的。解释式的动态网页开发工具如 ASP，PHP3 等由于速度等原因已经满足不了当前大型电子商务应用的需要，传统的开发技术都在向编译执行的方式转变。通常来说，JSP 页面一般是翻译为 Servlet 的 Java 源文件，再经过 Java 编译器编译为 Servlet 的 class 文件。这样，原来的 Servlet 引擎可以直接服务于 JSP，而 JSP 引擎就仅仅需要将 JSP 转译为

Servlet 就可以了。这里要注意的是, JSP 规范书中并没有规定如何将 JSP 页面转译为 Servlet, 因此, 不同的 JSP 引擎转译的结果也是不一样的。在 JSP 文件转译为 Servlet 以后, 每次客户机(通常是用户的 Web 浏览器)向服务器请求这一个 JSP 文件的时候, 服务器将检查自上次编译后 JSP 文件是否有改变, 如果没有改变, 就直接执行 Servlet, 而不用再重新编译, 其效率是相当高的。一般来说, JSP 文件的编译是在第一个用户访问到这个 JSP 页面时发生的, 而这第一个用户通常是开发人员自己, 这样正式放在服务器上让用户访问的 JSP 文件一般都已经有了对应的编译好的 Servlet 了。许多服务器都可以设置使 JSP 文件在第一个用户访问之前就预先编译好, 这样执行效率就更高了。

在选择使用一个 Java Servlet, 还是一个 JSP 页面时, 需要注意 Java Servlet 是一个程序设计工具, 它最适用于不需要频繁修改的低级应用功能; 而 JSP 页面则通过以显示为中心的描述性的方法将动态内容和逻辑结合在一起。对于使用一个 JSP 页面的简单的基于 Web 的应用程序, 可以使用定制标记或者 Scriptlet (Scriptlet 是直接嵌入在 JSP 页面中的很小的 Java 代码段), 而不是使用 JavaBeans 类来将内容与应用逻辑结合起来定制标记被打包到一个标记库中, 并被引入到一个 JSP 页面中。

简单地说, JSP 实际上就是 Java, 只是它是一种特别的 Java 语言。它在 Java 中加入了一个特殊的引擎, 这个引擎将 HTTPServlet 这个类的一些对象自动进行初始化, 以便于用户使用, 而用户不用再去操心前面的工作。同时这个引擎又引入了 `<jsp:*`, `<%@` 等一系列的特别语法, 使用这些语法来进行一些特定的操作, 如引用文件、URL 转向、JavaBean 引用等这样的操作, 都是由引擎自动完成初始化的工作。用户可以将这个引擎看作一个 JSP 到 Java Servlet 的生成器或是翻译器, 它可以帮助用户完成很多初始化的工作。

和其他 Web 开发工具相比, JSP 有着它自己强大的优势。

- 程序可以跨平台执行。JSP 可以让开发人员在任意环境中进行开发, 在任意环境中进行系统部署, 在任意环境中扩展应用程序。
- 多样化和功能强大的开发工具支持。Java 有许多非常优秀的开发工具, 而且有许多可以免费得到, 并且其中的许多已经可以顺利地运行于多种平台之下。
- 强大的可伸缩性。从只有一个小的 jar 文件就可以运行 servlet/jsp 到由多台服务器进行集群和负载均衡, 到多台 Application 进行事务处理, 一台服务器到无数台服务器, Java 显示了巨大的生命力。

当然, JSP 也有它的不足, Java 的一些优势也是它致命的问题所在。例如, 跨平台的功能和极度的伸缩能力极大地增加了产品的复杂性。也就是说, 它在扩展时需要分成多少块, 那么 java 系统中就有多少种产品, 所以用户可能会看到 jre, jdk, jsdk, jswdk 等, 而实际上它们是密不可分的。只要将它们有效地搭配在一起, 就可以产生强大的效能。当然, 这同时也使应用程序变得非常复杂。

另外, JSP 程序调试也不是很不方便的。JSP 页面首先被转化为一个 java 文件 (Servlet), 然后再被编译。这样, 出错信息实际上指向的是经过转化的那个 java 文件而不是 jsp 本身。例如在 .jsp 文件第一行的错误可能被显示为第 20 行。如果试图通过反跟踪来解决这个问题有时会使结果显得很滑稽。(在 Resin 的引擎中, 错误已经可以指向 .jsp 文件的出错行)。

1.2 JSP 中的脚本元素

在 JSP 中, 程序的主要部分就是脚本元素。JSP 的脚本元素包括 3 个部分: 声明(Declaration)、表达式(Expression)和脚本代码(Scriptlet)。声明用来定义一个变量或方法, 表达式可以将计算结果输出, 而脚本代码部分将是一些 Java 程序片断, 用于实现各种功能。下面分别介绍这三种脚本元素。

```
package org;  
import java.io.*;  
import javax.servlet.http.*;  
import java.sql.*;
```

1.2.1 声明

JSP 中的声明用来定义一个或多个合法的变量(包括普通变量和类变量)和方法, 并不输出任何的文本到输出流去, 声明的变量和方法将在 JSP 页面初始化时被初始化。

JSP 声明的语法格式为: `<%! declaration; [ declaration; ] ... %>`, 其中 declaration 为声明的变量、方法的名称和内容, 例如:

```
<%! int j = 0; %>  
<%! int x, y, z; %>  
<%! Circle a = new Circle(3.0); %>  
<%! String str="welcome!"; %>
```

在声明变量和方法时, 需要注意以下几点。

- 声明必须以“;”结尾。
- 可以直接使用在 `<% @ page %>` 中被包含进来的已经声明的变量和方法, 而不需要对它们重新进行声明。
- 一个声明仅在一个页面中有效。如果想在每个页面都用到一些声明, 最好把它们写成一个单独的文件, 然后用 `<%@ include %>` 或 `<jsp:include %>` 元素包含进来。

1.2.2 表达式

JSP 中的表达式可以被看作一种简单的输出形式, 可以将计算结果转换成一个字符串并输出。包含一个符合 JSP 语法的表达式的语法格式为: `<%= expression %>`。例如:

```
<p><%= getDate() %></p> //输出系统当前时间  
<%= "Hello" %> //输出 Hello  
<%= 1+3+5 %> //输出 9
```

注意:

JSP 的表达式中没有分号, 除非在加引号的字符串部分才使用分号。

1.2.3 脚本代码

JSP 脚本(Scriptlet), 即 JSP 中的代码部分, 是一段 java 程序的代码。它几乎可以使