



求是科技



JSP

通用模块及 典型系统开发

● 求是科技
王夕宁 王晓平 编著



人民邮电出版社
POSTS & TELECOM PRESS

JSP

通用模块及 典型系统开发

● 求是科技
王夕宁 王晓平 编著



人民邮电出版社
POSTS & TELECOM PRESS

图书在版编目 (CIP) 数据

JSP 通用模块及典型系统开发实例导航 / 求是科技编著. —北京: 人民邮电出版社, 2006.2
ISBN 7-115-14329-3

I. J... II. 求... III. JAVA 语言—主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2006) 第 007927 号

内 容 提 要

本书内容分为两部分: JSP 基础知识介绍和 JSP 设计实例。基础知识部分 (第 1 章和第 2 章) 系统地介绍了 JSP 的知识, 包括概念、特点、环境设置、服务器配置、语法、基本的数据库操作等。JSP 设计实例部分 (第 3 章~第 9 章) 介绍了在实际网络应用中常用模块的实现方法, 包括登录模块、注册模块、数据库典型模块、BBS 系统、网上投票系统、新闻发布系统以及电子商务系统等。

本书由经验丰富的 JSP 网站开发人员编写, 适合广大中、高级编程人员, 以及网站开发人员和数据库开发人员阅读。

JSP 通用模块及典型系统开发实例导航

- ◆ 编 著 求是科技 王夕宁 王晓平
责任编辑 张立科
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京鸿佳印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 26
字数: 632 千字 2006 年 2 月第 1 版
印数: 1—6 000 册 2006 年 2 月北京第 1 次印刷

ISBN 7-115-14329-3/TP · 5187

定价: 44.00 元 (附光盘)

读者服务热线: (010) 67132692 印装质量热线: (010) 67129223

前 言

随着 Java 技术的日益成熟和流行, JSP 技术在网络编程中也变得越来越重要。由于 JSP 基于强大的 Java 语言, 具有极强的扩展能力, 良好的可扩展性, 以及与平台无关的开发特性, 在基于 Java 平台构建动态商务网站成为主流的今天, JSP 有着其他技术所不具备的优势。

本书通过丰富的实例向读者介绍如何使用 JSP 技术开发 Web 网络系统应用程序, 这些实例取材于 Internet 应用开发中最常使用的系统或模块, 有的代码直接取自实际运行的系统, 是几乎每一个使用 JSP 进行开发的网络应用程序员都要遇到的。同时使用的开发方法也非常具有代表性, 是使用频率极高的一些方法, 可以说是每一个程序员都需要切实掌握的方法。

第 1 章是 JSP 基础部分, 分别介绍了运行环境和开发环境的安装配置, 以及 JSP 的基础知识, 涉及了 JSP 运行原理、JSP 的特点和优势以及 JSP 语法等。

第 2 章是 JSP 2.0 新特性及内建对象解析, 首先讲述了 JSP 2.0 的一些新特性, 然后分别讲述了 JSP 的内建对象 request、response、out、session、pageContext、application、config、page、exception。

第 3 章和第 4 章分别介绍登录和注册这两个任何电子商务网站都要用到的基本功能模块。我们都是先从最简单的设计开始建立一个可以运行的系统, 然后逐步完善, 逐步优化, 逐步求精。

第 5 章在前面几章实例的基础上, 系统地介绍了 JSP 开发中的数据库工作原理。第 6 章~第 9 章的几个系统来自于真实的网络应用。其中, 第 6 章在基本知识已经相当熟悉的基础上介绍一个简单的 BBS 系统的开发。第 7 章为网上投票系统, 阐述了投票程序的基本工作原理以及应该注意的问题, 然后详细介绍了一个功能完善的投票系统。第 8 章为新闻发布系统, 介绍了一个新闻自动发布系统的编写过程, 这样一个系统基本上已经是一个小型的新闻网站了, 当然也可以整合到社区系统里, 作为公告板发布网站信息。第 9 章讲述了如何使用 JSP 技术构建一个完整的电子商务网站系统。

本书的几个部分之间互为关联又相对独立, 读者可以选择阅读某一个部分或者是通读全文。由于时间仓促和水平有限, 书中错误和不妥之处在所难免, 敬请读者批评指正。

编者

2006 年 2 月

目 录

第 1 章 JSP 基础	1
1.1 运行环境的安装配置	2
1.1.1 运行环境的要求	2
1.1.2 运行环境的安装配置	3
1.2 开发环境的安装配置	5
1.3 JSP 基础知识	10
1.3.1 JSP 运行原理	10
1.3.2 JSP 的特点与优势	13
1.3.3 JSP 语法简介	14
1.4 本章小结	17
第 2 章 JSP 2.0 新特性及内建对象解析	18
2.1 JSP 2.0 的新特性	18
2.1.1 简单表达式语言(EL)	18
2.1.2 JSP 片段	21
2.1.3 标记文件及简单标记处理程序	22
2.2 JSP 的内建对象概述	26
2.2.1 JSP 的内建对象一: request	27
2.2.2 JSP 的内建对象二: response	32
2.2.3 JSP 的内建对象三: out	35
2.2.4 JSP 的内建对象四: session	36
2.2.5 JSP 的内建对象五: pageContext	39
2.2.6 JSP 的内建对象六: application	42
2.2.7 JSP 的内建对象七: config	43
2.2.8 JSP 的内建对象八: page	43
2.2.9 JSP 的内建对象九: exception	44
2.3 本章小结	44
第 3 章 登录模块设计	45
3.1 最基本的登录模块	45
3.1.1 客户端界面模块 login.html	46
3.1.2 服务器端数据处理模块 verifylogin.jsp	48
3.2 添加输入验证功能	50
3.2.1 添加客户端验证代码	51
3.2.2 添加服务器端验证代码	53

3.2.3	限制字符串的长度	55
3.3	JSP 技术网站开发的两种模式	56
3.3.1	模式一: JSP+Java Bean	56
3.3.2	模式二: JSP+Servlet+Java bean	57
3.3.3	应用模式二开发登录模块	57
3.4	本章小结	64
第 4 章	注册模块设计	65
4.1	最基本的注册模块	65
4.1.1	扩充 UserInfo 表	66
4.1.2	客户端注册页面	67
4.1.3	注册确认处理	70
4.1.4	注册确认页面	72
4.1.5	注册处理	75
4.1.6	注册处理页面	78
4.2	增强功能的注册模块	80
4.2.1	改进的客户端注册页面	80
4.2.2	改进的注册处理	85
4.3	完整功能的注册模块	88
4.4	本章小结	93
第 5 章	数据库操作模块	94
5.1	数据库及 SQL 语句	94
5.1.1	建表、修改和删除表的语句	94
5.1.2	查询语句	95
5.1.3	插入、更新和删除语句	95
5.1.4	条件子句	95
5.2	JDBC 技术工作原理	96
5.3	JDBC 4 种类型的驱动	98
5.3.1	JDBC-ODBC Bridge	98
5.3.2	JDBC- Native Bridge	98
5.3.3	JDBC Network Bridge	99
5.3.4	Pure Java JDBC Driver	99
5.4	JDBC 接口	99
5.4.1	Driver	99
5.4.2	DriverManager	99
5.4.3	Connection	102
5.4.4	Statement	103
5.4.5	ResultSet	106
5.5	数据库连接实例	108
5.5.1	建立与 Oracle 的连接	108

5.5.2 建立与 MySQL 的连接	109
5.5.3 建立与 Microsoft SQL Server 7.0/2000 数据库的连接	111
5.5.4 建立与 DB2 数据库的连接	112
5.5.5 建立与 Informix 数据库的连接	113
5.5.6 建立与 Sybase 数据库的连接	113
5.6 数据库连接池	114
5.6.1 数据库连接池基本概念	114
5.6.2 数据库连接池技术的优点	115
5.6.3 数据库连接池原理及设计实现	116
5.6.4 数据库连接池类实现	124
5.7 本章小结	133
第 6 章 讨论区 BBS 模块设计	134
6.1 留言板	134
6.1.1 撰写留言模块	135
6.1.2 浏览留言列表模块	139
6.1.3 阅读留言模块	142
6.1.4 留言管理模块	144
6.1.5 添加新功能后的浏览模块	153
6.2 建立 BBS 讨论区	156
6.2.1 建立所需要的表	156
6.2.2 系统登录及错误提示页面	158
6.2.3 留言列表	163
6.2.4 阅读留言	170
6.2.5 撰写帖子	175
6.2.6 修改留言	181
6.2.7 回复帖子	186
6.2.8 删除帖子	190
6.2.9 BBS 业务处理模块	191
6.3 本章小结	210
第 7 章 投票系统模块设计	212
7.1 一个简单的投票器	212
7.1.1 投票模块 (simple_vote.jsp)	212
7.1.2 显示模块 (simple_showvote.jsp)	214
7.1.3 计数模块	216
7.2 禁止重复投票	219
7.2.1 用 Session 变量来跟踪是否已投票	219
7.2.2 用 Cookies 来跟踪是否已投票	219
7.3 一个功能完善的投票系统	222
7.3.1 系统的文件构成和数据结构分析	222

7.3.2	公用 Java Bean 及业务处理模块	223
7.3.3	管理员登录及身份验证模块	238
7.3.4	管理界面首页	243
7.3.5	新增投票项目模块	248
7.3.6	删除投票项目模块	253
7.3.7	修改投票项目模块	255
7.3.8	显示投票项目模块	260
7.3.9	投票处理模块	263
7.3.10	投票结果显示模块	266
7.4	本章小结	270
第 8 章	新闻自动发布系统	271
8.1	一个完整的新闻发布系统	271
8.1.1	表结构设计	271
8.1.2	后台管理	272
8.2	栏目管理	277
8.2.1	栏目管理显示模块	277
8.2.2	添加栏目模块	280
8.2.3	修改栏目模块	284
8.2.4	删除栏目模块	288
8.3	管理员管理	289
8.3.1	管理员显示模块	289
8.3.2	添加管理员模块	292
8.3.3	修改管理员模块	296
8.3.4	删除管理员模块	303
8.4	新闻管理	304
8.4.1	新闻管理显示模块	304
8.4.2	新闻发布模块	309
8.4.3	修改新闻模块	313
8.4.4	删除新闻模块	317
8.5	新闻系统前端模块	319
8.5.1	新闻首页显示模块	319
8.5.2	新闻内容显示模块	326
8.5.3	新闻搜索模块	333
8.6	本章小结	338
第 9 章	电子商务系统	339
9.1	电子商务网站规划和设计	339
9.1.1	网站的整体结构	339
9.1.2	站点的创建	342
9.1.3	建立数据库	342

9.1.4	公用页面	346
9.1.5	公用 Java 类	352
9.2	用户注册登录	359
9.2.1	用户注册	360
9.2.2	用户登录	367
9.2.3	密码查询	369
9.2.4	用户注销	371
9.2.5	用户列表	372
9.3	商品管理	376
9.3.1	查询商品	376
9.3.2	商品分类	380
9.3.3	商品列表	383
9.3.4	发布商品	385
9.3.5	最新、特价、缺货商品列表	388
9.4	购物车	391
9.4.1	选购商品放入购物车	392
9.4.2	显示购物车中商品信息	394
9.5	用户订单	398
9.5.1	提交订单	398
9.5.2	订单列表	401
9.5.3	处理订单	402
9.6	本章小结	404

第 1 章 JSP 基础

随着网络技术的不断发展,单纯的静态页面已经不能满足需要。因为静态页面是由纯粹的 HTML 语言组成的,它没有交互性。因此,为了满足实际的需要,许多动态网站开发技术(如 ASP、JSP、PHP)出现在众多 Web 站点开发人员面前。这时的网页文件扩展名也不再只是简单的“.htm”、“.html”,还有“.php”、“.asp”、“.jsp”、“.shtml”等,这些都是采用动态网页技术制作出来的。

在早期,开发网络数据库应用程序主要采用 CGI 技术(Common Gateway Interface 公用网关接口技术)。编写 CGI 程序可以使用不同的程序语言,如 Perl、Visual Basic、Delphi 或 C/C++ 等。虽然 CGI 技术已经发展成熟而且功能强大,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。

在这样的背景下,新的技术方案纷纷面世,如 Microsoft 的 ASP(Active Server Page),Tcx 的 PHP(Hypertext Preprocessor),Sun 的 Java/Jsp/Servlet 等。其中 Sun 的 Java/JSP/Servlet 技术被许多人认为是未来最有发展前途的动态网站技术。

JSP(Java Server Pages)是由 Sun Microsystems 公司倡导,许多公司参与建立的一种动态网页技术标准。其发展的历程是这样的:

1998 年 4 月, Sun 公司发布 JSP 0.90 规范;

1999 年 1 月, Sun 公司发布 JSP 0.92 规范,推出支持 JSP 的 Web 服务器 Java Web Server 2.0;

1999 年 11 月, Sun 公司发布 JSP 1.1 规范,同时推出 JSWDK 1.0.1 和 Servlet 2.2 版;

2000 年 9 月, Sun 公司发布 JSP 1.2 规范,同时推出 Java Servlet API 2.3 版;

2003 年, Sun 公司的 J2EE 版本已经升级到了 1.4 版,其中包括发布了 JSP 2.0 规范,同时推出 Java Servlet 2.4 版本。

本书中的所有示例均基于最新的规范,即 JSP 2.0 与 Servlet 2.4。

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,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。

JSP 基于强大的 Java 语言,具有良好的可扩展性,与 Java Enterprise API 紧密地集成在一起,在网络数据库应用开发领域中具有得天独厚的优势。基于 Java 平台构建网络程序已经被越来越多的人认为是未来最有发展前途的技术之一。

1.1 运行环境的安装配置

1.1.1 运行环境的要求

下面首先将讲述一下 JSP 对运行环境的一些要求。本节中将从硬件条件、操作系统和软件环境 3 种基本的方面进行阐述。

1. 对硬件条件的要求

JSP 对硬件要求的最低配置，如表 1-1 所示。

表 1-1 对硬件条件的要求

CPU	使用 586 或更高级别的处理器
内存	64MB 及其以上
硬盘	100MB 以上
光驱	8 速以上 CD-ROM

2. 对操作系统的要求

JSP 能够运行在绝大多数的操作系统上，在普通用户中用到的系统绝大多数为 Windows 系列、UNIX 系列和 Linux 系列，如表 1-2 所示。

表 1-2 对操作系统的要求

操作系统	Windows 98, Windows NT 4.0, Windows 2000 professional, Windows 2000 server, Windows XP 等
	UNIX 系统
	Redhat Linux, Turbo Linux 等

3. 对服务器软件环境的要求

我们知道，要使一个网站正常运行，除了必要的硬件外，还要有相应的服务器软件。因此我们需要安装如表 1-3 所示的软件，以便使得网站正常运行。

表 1-3 对软件环境的要求

Web 浏览器	任何支持 HTML TABLE 和 FORM 元素的浏览器 同时浏览器应支持如下 HTML 元素，如 Frames、Tables、客户端的图片映射、文件上载（可选）、Javascript、Cookie（可选）等
Web 服务器	BEA Weblogic、IBM WebSphere Server、Apache Tomcat 等
数据库	Oracle、Sybase、SQL Server、DB2、MySQL 等
Java2 SDK	Java2 SDK、Standard Edition Version 1.4.2 以上

1.1.2 运行环境的安装配置

1. Java 2 SDK 的安装配置

Java 2 SDK 的安装过程非常简单。在这里，只对 Java 2 SDK 的安装作以简单的说明。下面以安装 Java (TM) 2 SDK Standard Edition 1.4.2_05 为例进行说明。

从 SUN 官方网站 <http://java.sun.com/j2se/1.4.2/download.html> 上可以下载 Java 2 SDK 的安装程序，下载完毕后，双击 j2sdk1.4.2_05-win.exe 文件即可进行安装。首先出现版权协议说明界面，如图 1-1 所示。

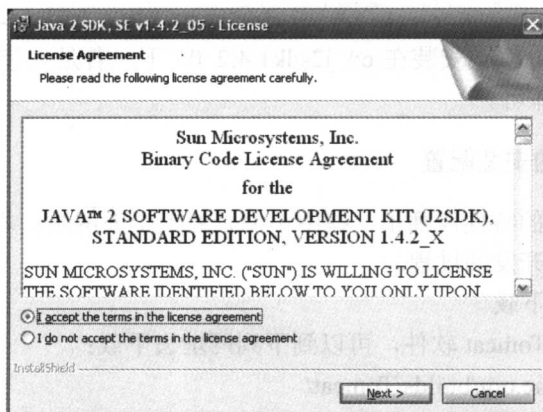


图 1-1 版权协议说明界面

接受版权协议后，单击“Next”按钮，出现如图 1-2 所示的界面。选择所有的安装部件和安装的路径，如 c:\j2sdk1.4.2_05，然后单击“Next”按钮进行安装。安装完毕后，系统会显示安装成功界面，如图 1-3 所示。单击“Finish”按钮结束安装。

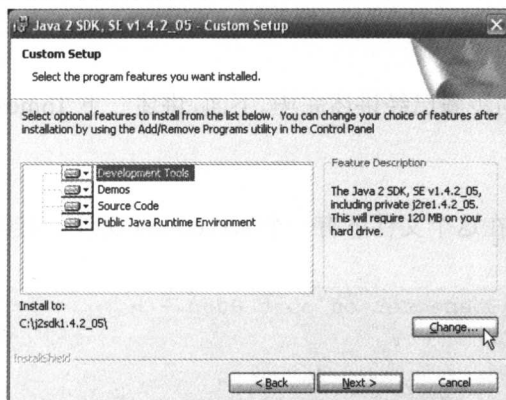


图 1-2 选择安装部件及安装路径

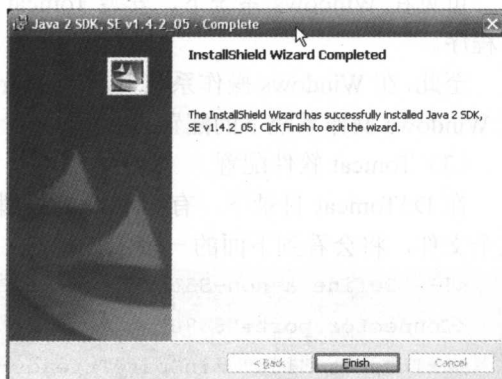


图 1-3 Java 2 SDK 安装成功

除了配置标准的 Java 环境变量外，还应该设置 PATH 环境变量与 CLASSPATH 环境变量。

(1) PATH 命令行路径参数的设置

创建的脚本示例通常都会在创建或运行某个示例时，自动地设置 PATH 环境变量。PATH 是用来指定可执行应用程序的路径。

设置的方法如下：

```
PATH=%PATH%; c:\j2sdk1.4.2_05
```

在上述设置中，JDK 被安装在 c:\j2sdk1.4.2_05 目录下。若是安装在其他的目录下，上述代码只要做相应的改变即可。

(2) CLASSPATH 类路径参数的配置

所创建的脚本示例通常都会在创建或运行某个示例时，自动地设置 CLASSPATH 环境变量。CLASSPATH 是用来指定在初始化 JRE (Java Run Environment Java 运行环境) 时需要使用的类文件与文档文件 (也就是 JAR 文件)。

设置的方法如下：

```
set CLASSPATH= c:\j2sdk1.4.2_05\lib\Tools.jar; c:\j2sdk1.4.2_05\lib\dt.jar
```

在上述设置中，JDK 被安装在 c:\j2sdk1.4.2_05 下。若是安装在其他的目录下，上述代码只要做相应的改变即可。

2. Web 服务器的安装配置

Tomcat 凭着开源的特性风靡于一些中小型企业或个人网站。本章以在 Windows+Tomcat 作为案例，介绍 Web 开发的过程。

(1) Tomcat 软件下载

如果读者还没有 Tomcat 软件，可以到下列网址去下载：

<http://jakarta.apache.org/builds/Tomcat/>

根据版本号下载所需的版本 (本章采用版本 5.0.28)，得到的文件是一个压缩文件，名字叫 Jakarta-Tomcat-5.0.28.zip。

(2) Tomcat 软件安装

用 WinZIP 或 WinRAR 等解压缩软件把 Jakarta-Tomcat-5.0.28.zip 解压缩到一个目录下，例如把它解压缩到 D 盘。这时，系统会自动在用户计算机系统的 D 盘中创建 Tomcat 子目录。

可见在 Windows 系统下，安装 Tomcat 非常简单。只要解压缩就行，不需要运行其他任何程序。

至此，在 Windows 操作系统下，安装 Tomcat 的步骤已经讲述完毕。下面，讲述一下 Tomcat 在 Windows 操作系统下的配置。

(3) Tomcat 软件配置

在 D:\Tomcat 目录下，有一个 conf 文件夹，在这个文件下有一个 server.xml 文件。打开这个文件，将会看到下面的一段代码：

```
<!-- Define a non-SSL Coyote HTTP/1.1 Connector on port 8080 -->
<Connector port="8080"
maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" redirectPort="8443" acceptCount="100"
    debug="0" connectionTimeout="20000"
    disableUploadTimeout="true"/>
```

Tomcat 可以作为一个独立的服务器使用，所以 Tomcat 有自己的端口号，默认为 8080。这个端口号可以修改，建议设置为用户自己的计算机上还没有被占用的端口号。本章采用了默认端口号 8080。

1.2 开发环境的安装配置

Lomboz 是一个基于 Eclipse 开发平台的用于进行 Web 开发的插件，它将很多 Java 应用服务器、J2EE 组件和 Web 应用开发集成到 Eclipse 中，可以帮助 Java 开发者使用 Eclipse 建立、测试、部署 J2EE 应用。对于基于 JSP 技术的 Web 开发人员来说，Lomboz 无疑是一个很好的开发利器，它本身能够支持 JSP 的编写、调试等工作。具体来说，Lomboz 有以下相关的支持，即：支持 JSP 编写，包括语法着色和语法检查；支持 JSP 的调试。

下面具体介绍 Lomboz 的安装过程。

(1) 从 Lomboz 的下载地址 (<http://lomboz.objectweb.org/downloads/download.php>) 下载软件 lomboz-emf-gef-jem-I20050526.zip，解压后运行 eclipse.exe 文件，出现如图 1-4 所示的界面。

(2) 在该界面中选择一个目录作为工作空间，然后单击“OK”按钮，打开如图 1-5 所示的欢迎界面。

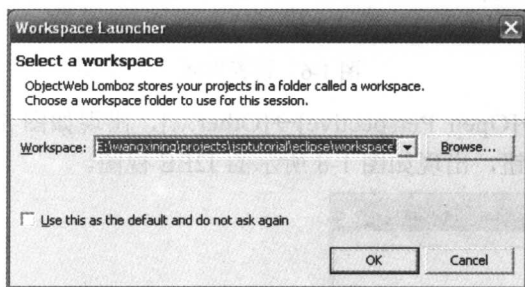


图 1-4 Eclipse 启动界面

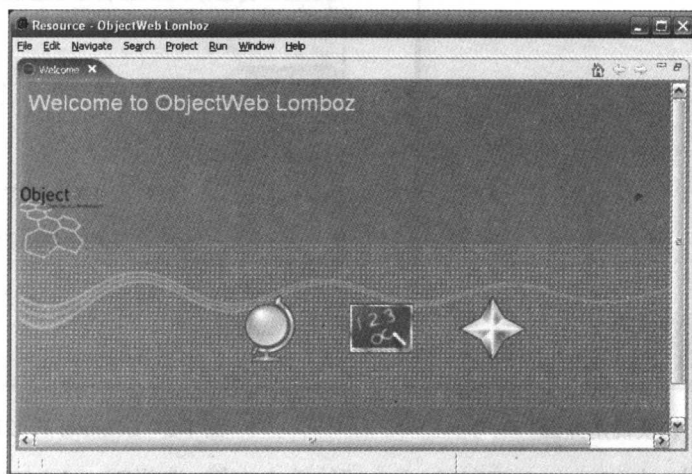


图 1-5 欢迎界面

选择菜单[Window]→[Preferences]→[Java]→[Build Path]，出现如图 1-6 所示界面。在“Source and output folder”栏内选中“Folders”项，并在“Source folder name”与“Output folder name”文本框中分别输入“src”、“bin”。

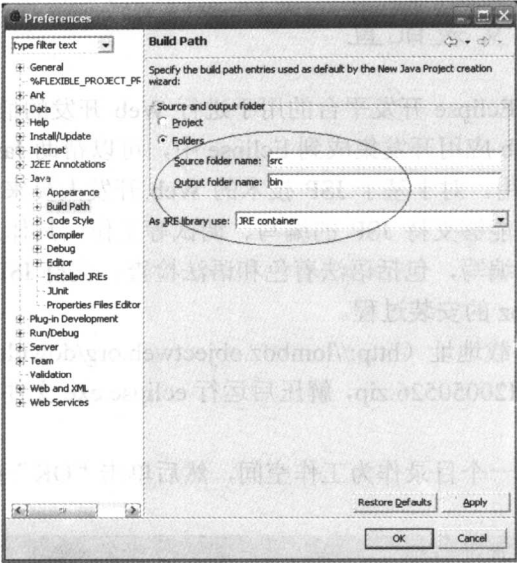


图 1-6 设置界面

选择菜单[Window]→[Open Perspective]→[Other...], 出现如图 1-7 所示界面。从其中选择 J2EE, 单击“OK”按钮, 出现如图 1-8 所示的 J2EE 视图。

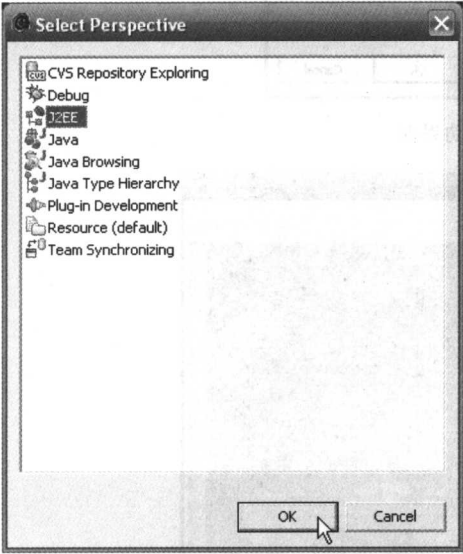


图 1-7 切换视角界面

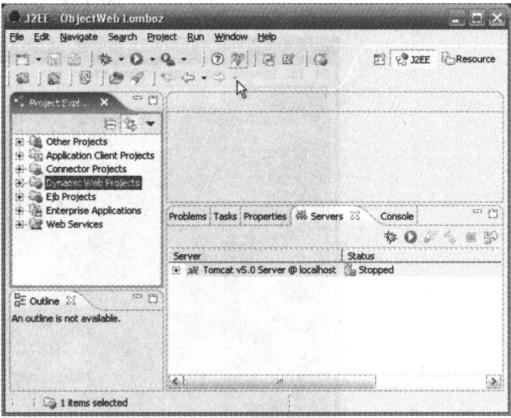


图 1-8 Lomboz 的 J2EE 视图

在 J2EE 视图中, 在“Dynamic Web Projects”处单击右键, 选择[New]→[J2EE Web Project], 如图 1-9 所示, 然后出现如图 1-10 所示的界面。在图 1-10 中, 为 Web 工程命名为“HelloWorld Project”, 模块版本为 2.4, 其他设置保持默认设置。在“Target Server”处单击“New...”按钮, 出现如图 1-11 所示的新建服务器运行时的界面。

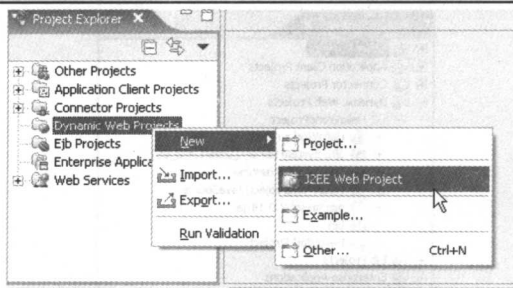


图 1-9 新建 Web 工程

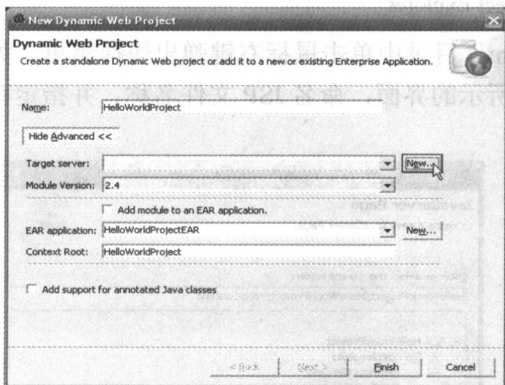


图 1-10 Web 工程界面

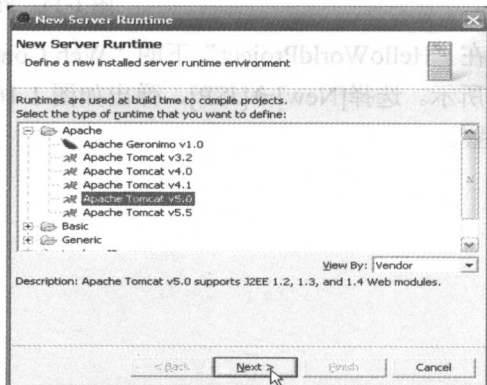


图 1-11 新建服务器运行时的界面

选择“Apache”下的“Apache Tomcat V5.0”，单击“Next”按钮，出现如图 1-12 所示的界面。在图 1-12 界面中，指定新建 Tomcat 服务器的名称、Tomcat 安装路径以及 JRE 安装路径。设置完成后，单击“Finish”按钮。这时 Web 工程设置界面如图 1-13 所示。其中“Target Server”处列出了新建的“Apache Tomcat v 5.0”。

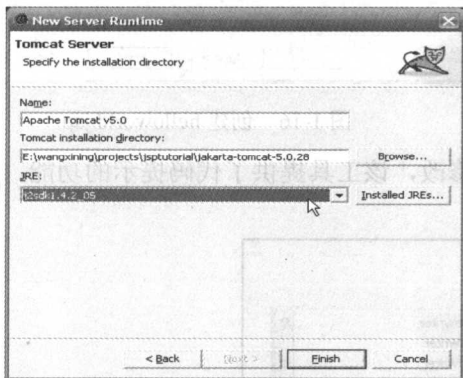


图 1-12 指定 Tomcat 安装路径

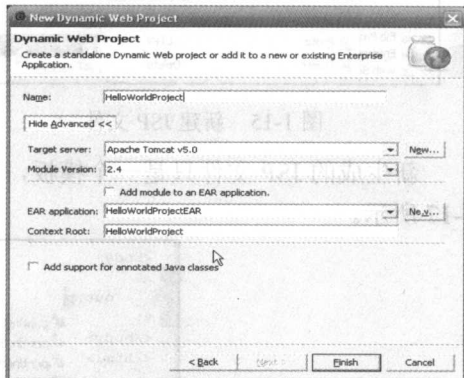


图 1-13 设置完成后的 Web 工程界面

单击“Finish”按钮，完成 Web 工程的建立。此时，在“Project Explorer”下会出现建立的 Web 工程，如图 1-14 所示。

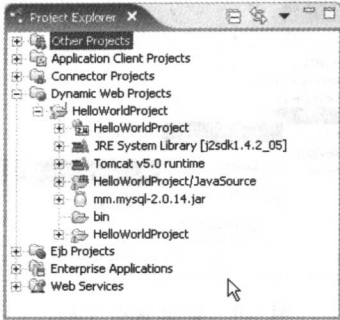


图 1-14 Project Explorer

在“HelloWorldProject”下的“Web Content”目录中单击鼠标右键弹出快捷菜单，如图 1-15 所示。选择[New]→[JSP]，弹出如图 1-16 所示的界面，命名 JSP 文件名称，并指定存放的位置。

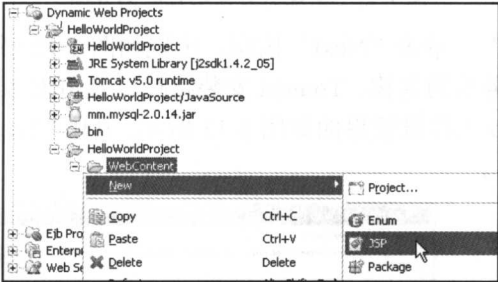


图 1-15 新建 JSP 文件

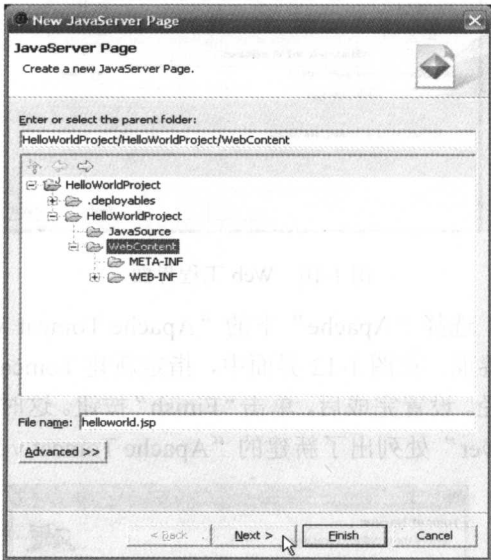


图 1-16 创建 helloworld.jsp

新生成的 JSP 文件只是一个模板，可以进行修改，该工具提供了代码提示的功能，如图 1-17 所示。

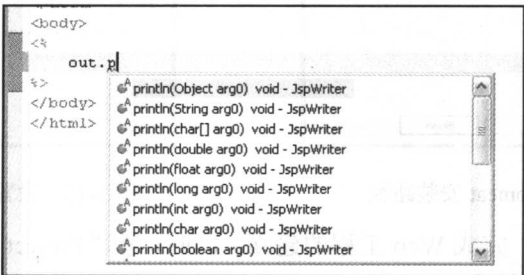


图 1-17 代码提示

修改后的 helloworld.jsp 文件内容如下：