

项目开发实践系列

附赠光盘内含书中程序源代码

JSP

项目开发实践

池雅庆 周琚 岳元媛 编著

- 培养应用程序开发大局观：从全局设计的高度为读者讲解如何使用JSP语言开发符合要求和规范的Web应用程序。
- 演绎应用程序开发全过程：从系统设计者的角度综合考虑，完整地讲述了从需求到设计，再到实现的整个Web应用程序的开发过程。
- 面向实际开发，解决实际问题：本书案例包括学生管理系统、在线考试系统、图书馆管理系统、在线购物系统、聊天室系统以及通用考勤系统，具有很好的代表性。

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

项目开发实践系列

JSP 项目开发实践

池雅庆 周 珺 岳元媛 编著



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书系统全面地介绍了使用 JSP 语言开发 Web 应用程序的全过程。本书撰写另辟蹊径,从系统设计者的角度综合考虑,完整地讲述了从需求到设计,再到实现整个 Web 应用程序的开发过程。在编写过程中,坚持理论与实践并重,面向实际开发,解决实际问题。

本书首先对 JSP 语言开发环境和语法进行简单而必要的介绍,然后根据经典开发案例讲解如何进行 JSP 数据库应用程序的设计和开发。这些案例涵盖教育、电子商务、网络通信及公司考勤等方面,具有很好的代表性。

本书可作为高等院校软件工程专业学生的课外辅导教材,也可作为 JSP Web 应用程序开发者的参考书及相关公司员工培训的教学资料。

图书在版编目(CIP)数据

JSP 项目开发实践/池雅庆,周珺,岳元媛编著.
北京:中国铁道出版社,2006.10
(项目开发实践系列)
ISBN 7-113-07520-7

I. J... II. ①池... ②周... ③岳... III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 124538 号

书 名: JSP 项目开发实践

作 者: 池雅庆 周 珺 岳元媛

出版发行: 中国铁道出版社(100054,北京市宣武区右安门西街8号)

策划编辑: 严晓舟 魏 春

责任编辑: 苏 茜 林菁菁 包 宁

封面设计: 高 洋

封面制作: 白 雪

责任校对: 王雪飞

印 刷: 三河市国英印务有限公司

开 本: 787×1092 1/16 印张: 17.25 字数: 402 千

版 本: 2006 年 11 月第 1 版 2006 年 11 月第 1 次印刷

印 数: 1~4 000 册

书 号: ISBN 7-113-07520-7/TP·2080

定 价: 30.00 元(含盘)

版权所有 侵权必究

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

前言

自互联网兴起以来，它在人们生活中所占据的地位变得越来越重要。现在，互联网几乎已经深入到了人们生活的方方面面，而且这种影响将更进一步地深入下去。随着用户规模的不断扩大，互联网上所蕴含的商机越来越被社会重视；同时，它作为一个快捷、方便而又廉价的信息交流平台深受人们青睐。

社会信息化的加速发展，呼唤更多的基于 Internet 的应用程序出现。在众多的 Web 应用程序开发语言当中，Java 语言以其简单易用、完全面向对象、具有平台无关性且安全可靠等特点成为面向 Internet 应用程序的主要开发工具。自从 1995 年正式问世以来，Java 的快速发展已经让整个 Web 世界发生了翻天覆地的变化。作为 Java 技术家族中重要组成部分的 JSP 和 J2EE，已成为越来越多开发者首选的 Web 开发工具。

JSP 语言基于 Java 的动态网页开发技术，是许多 Web 工程师建站的首选语言。目前，介绍 JSP 开发技术的书籍很多，但是，这些书籍大多着眼于 JSP 技术本身，而没有上升到整站设计和建设的高度。为了弥补这个缺陷，本书特意精选了几个经典的开发案例，试图从全局设计的高度为读者讲解如何使用 JSP 语言开发符合要求和规范的 Web 应用程序。

本书共分 8 章，大体可以分成两个部分：第 1 章和第 2 章是对 JSP 语言开发环境和语法简单而必要的介绍，第 3 章～第 8 章则使用了 6 个经典开发案例讲解如何进行 JSP 数据库应用程序的设计和开发。这些案例涵盖教育（第 3 章“学生管理系统”、第 5 章“在线考试系统”、第 8 章“图书馆管理系统”）、电子商务（第 4 章“在线购物系统”）、网络通信（第 6 章“聊天室系统”）及公司考勤（第 7 章“通用考勤系统”）等方面，具有很好的代表性。

如前所述，本书的特色在于从建站全局的角度来讲解 JSP 开发应用，所以本书中的案例介绍都遵循典型的设计开发流程：首先收集和分析应用系统的需求，做出需求分析。然后根据需求，使用通用建模语言（UML）对系统进行建模。系统模型确立之后，对系统数据库进行设计和建模，并从设计系统的界面开始，最后根据设计编写代码，实现整个系统。

编者在编写过程中，参阅了许多相关书籍和教材，从中汲取了许多有益的内容，恕不在此一一列举。参加编著的主要人员有池雅庆博士、周珺助教、岳元媛硕士、王耀硕士等人，王鹏、吴珊、宋旻、刘欣、杜方冬、程璐、胡宇峰、李俊、陈先来、姜新玲、刘莉、申镇、赵城利等人参与了本书的资料收集与程序调试。另外，本书出版是在中国铁道出版社的大力支持下完成的，特此表示感谢。

由于编者水平有限，书中疏漏之处在所难免，望广大读者批评指正！

编者

2006 年 11 月

目 录

第 1 章 JSP 和 J2EE 开发环境指南.....	1
1-1 JSP 基本开发环境.....	1
1-2 JSP 运行平台的建立.....	2
1-2-1 安装 J2SDK 开发工具.....	2
1-2-2 安装 Tomcat.....	4
1-2-3 安装数据库驱动程序.....	6
1-2-4 中文乱码问题.....	8
1-3 JSP 开发 Web 的几种主要方式.....	10
1-3-1 直接使用 JSP.....	10
1-3-2 JSP + JavaBeans.....	10
1-3-3 JSP + JavaBeans + Servlet.....	10
1-4 J2EE 开发环境的建立.....	10
1-4-1 配置 J2EE 环境变量.....	10
1-4-2 建立 Web 服务器.....	11
1-5 小结.....	15
第 2 章 JSP 编程基础.....	16
2-1 JSP 基本语法.....	16
2-1-1 通用语法规则.....	16
2-1-2 注释.....	17
2-1-3 声明.....	17
2-1-4 表达式.....	18
2-1-5 Scriptlet.....	18
2-2 JSP 指令元素.....	18
2-2-1 page 指令.....	19
2-2-2 include 指令.....	20
2-2-3 taglib 指令.....	20
2-2-4 JSP 动作指令.....	21
2-3 JSP 内部对象.....	23
2-3-1 Out 对象.....	24
2-3-2 Request 对象.....	25
2-3-3 Response 对象.....	25
2-3-4 Application 对象.....	26
2-3-5 Exception 对象.....	26
2-3-6 Session 对象.....	27

2-4	Java Servlet	28
2-4-1	Servlet 的生命周期	28
2-4-2	与 HTTP 相关的 Servlet API	29
2-4-3	系统信息	32
2-5	JavaBeans	33
2-5-1	JavaBeans 的属性	33
2-5-2	JavaBeans 的事件	36
2-5-3	持久化	39
2-5-4	用户化	39
2-6	小结	40
第 3 章	学生成绩管理系统	41
3-1	收集与分析应用需求	41
3-1-1	收集需求	41
3-1-2	分析需求	42
3-2	系统建模	42
3-2-1	学生成绩管理系统的用例分析	42
3-2-2	学生成绩管理系统的领域分析	47
3-2-3	学生成绩管理系统的系统设计	48
3-3	设计数据库	57
3-3-1	概念设计	57
3-3-2	数据库的实现	58
3-4	设计用户界面	59
3-4-1	登录模块	59
3-4-2	学生模块	60
3-4-3	教师模块	61
3-4-4	管理员模块	62
3-5	系统结构和流程	63
3-6	代码实现	66
3-7	小结	78
第 4 章	网络购物系统	79
4-1	收集与分析应用需求	79
4-1-1	收集需求	79
4-1-2	分析需求	81
4-2	系统建模	81
4-2-1	网络购物系统的用例分析	81
4-2-2	网络购物管理系统的领域分析	83
4-2-3	网络购物系统的系统设计	84
4-3	设计数据库	89

4-4	设计用户界面.....	92
4-4-1	登录模块.....	92
4-4-2	管理模块.....	93
4-4-3	用户界面模块.....	94
4-4-4	购物界面模块.....	95
4-5	系统结构和流程.....	97
4-6	代码实现	98
4-7	小结	118
第 5 章	在线考试系统	119
5-1	收集与分析应用需求.....	119
5-1-1	收集需求.....	119
5-1-2	分析需求.....	120
5-2	系统建模	120
5-2-1	在线考试系统的用例分析	120
5-2-2	在线考试系统的领域分析	122
5-2-3	在线考试系统的系统设计	123
5-3	设计数据库.....	131
5-4	设计用户界面.....	135
5-4-1	管理员界面模块.....	135
5-4-2	学生界面模块.....	136
5-4-3	教师界面模块.....	139
5-5	代码实现	140
5-6	小结	156
第 6 章	聊天室系统的设计与实现	157
6-1	聊天室系统介绍.....	157
6-1-1	客户/服务器模式.....	157
6-1-2	聊天室功能介绍	158
6-1-3	聊天室系统实现.....	159
6-2	Java 网络编程基础.....	159
6-2-1	Socket 类.....	159
6-2-2	DatagramSocket 和 DatagramPacket 类.....	160
6-3	聊天室系统设计.....	161
6-3-1	系统建模.....	161
6-3-2	聊天室系统的系统设计	163
6-4	设计数据库.....	170
6-4-1	数据库需求分析.....	170
6-4-2	数据库概念分析.....	171
6-4-3	数据库逻辑结构设计	172

6-5	设计用户界面.....	173
6-5-1	客户端界面.....	174
6-5-2	服务器界面.....	175
6-5-3	聊天室 Web 管理界面.....	175
6-6	服务器端设计实现.....	177
6-6-1	系统通信包设计.....	177
6-6-2	系统类模块设计.....	178
6-6-3	重点代码分析.....	179
6-7	Web 管理端设计实现.....	192
6-7-1	Web 管理端模块设计.....	192
6-7-2	Web 管理端代码分析.....	193
6-8	小结.....	197
第 7 章	通用考勤管理系统.....	198
7-1	收集与分析应用需求.....	198
7-1-1	收集需求.....	198
7-1-2	分析需求.....	199
7-2	系统建模.....	200
7-2-1	考勤系统的用例分析.....	200
7-2-2	考勤系统的领域分析.....	202
7-2-3	考勤系统的系统设计.....	203
7-3	设计数据库.....	209
7-3-1	实体关系分析.....	209
7-3-2	表字段分析和数据库建模.....	210
7-4	设计用户界面.....	211
7-4-1	员工操作界面.....	211
7-4-2	出勤登记界面.....	212
7-4-3	人事管理界面.....	213
7-4-4	系统维护界面.....	214
7-5	设计 ActionForm.....	214
7-6	设计 Action 和 Action 映射.....	223
7-7	小结.....	224
第 8 章	图书馆管理系统.....	226
8-1	收集与分析应用需求.....	226
8-1-1	收集需求.....	226
8-1-2	分析需求.....	226
8-2	系统建模.....	227
8-2-1	图书馆管理系统的用例分析.....	227
8-2-2	图书馆管理系统的领域分析.....	232
8-2-3	图书馆管理系统的系统设计.....	233

8-3	设计数据库.....	240
8-3-1	概念设计.....	240
8-3-2	关系数据库的逻辑设计.....	241
8-3-3	数据库的实现.....	242
8-4	设计用户界面.....	244
8-4-1	图书查询界面.....	244
8-4-2	借阅管理界面.....	245
8-4-3	图书归还界面.....	246
8-4-4	图书管理界面.....	246
8-4-5	借阅证管理.....	247
8-4-6	读者规则管理界面.....	247
8-4-7	读者信息查询界面.....	248
8-5	代码实现.....	250
8-6	小结.....	264

JSP 和 J2EE 开发环境指南

Java 是一种简单易用、完全面向对象、具有平台无关性、安全可靠且主要面向 Internet 的开发工具。自从 1995 年正式问世以来,Java 的快速发展已经让整个 Web 世界发生了翻天覆地的变化。作为 Java 技术家族中重要组成部分的 JSP 和 J2EE,已成为越来越多开发者首选的 Web 开发工具。本章将详细介绍 JSP 和 J2EE 开发环境的配置方法,为读者尽快熟悉这两种工具的开发步骤提供帮助。

1-1 JSP 基本开发环境

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 环境有很多种途径,但主要的工作大体都可以分为 3 部分:安装 JDK、配置 JSP 环境引擎和安装数据库驱动程序。

JSP 建立在 Java Servlets 模型之上,所以开发和运行 JSP 必须先安装 Java 的编译和测试工具 JDK。JDK 是整个 Java 的核心,包括 Java 运行环境、一些 Java 工具和 Java 的基础类库。所有的 Java 应用服务器都内置了某个版本的 JDK。因此,掌握 JDK 是学好 Java 的第一步。对于初学者来说,推荐使用 Sun 公司的 JDK。在安装前先要下载 JDK,可以从 SUN 公司的网站上下载,下载页面为: <http://java.sun.com/j2se/1.4.2/download.html>。

而所谓的 JSP 环境引擎指的就是 JSP 的运行环境,就像 ASP 的运行环境是 IIS,PHP 的运行环境可以是 Apache+PHP。JSP 的一大特点是它支持多平台,不仅支持 Windows 同样也支持 Linux 和 UNIX,现在流行的 JSP 引擎主要有 Tomcat、JSWDK、Resin、JRun、WebSphere、WebLogic、GNUJSP 等,其中最常用到的是 Tomcat。

Tomcat 由著名的 Apache 开发,并受到 Sun 公司的大力支持,所以它很可能成为下一代 Java Web Server 的主流。Tomcat 是 Apache 上实现 JSP + Java Bean 环境的接口程序,集成了 JSP 和 Servlet。相信 Tomcat 将不断完善成一个理想的 JSP 和 Servlet 开发和支撑平台。

Tomcat 的功能包含 3 个主要部分:

- ◆ Catalina, 一个符合 Servlet API 规范 2.3 的 Servlet 容器。
- ◆ Jasper, 一个符合 JSP 规范 1.2 的 JSP 编译器和运行环境。
- ◆ Webapps, Tomcat 中包含的一些例子和用于测试的 Web 例程,以及相关文档。

作为一个开放源代码的软件, Tomcat 有着自己独特的优势。

首先,它容易得到。任何人都可以从互联网上自由下载这个软件。

其次,对于开发人员,特别是 Java 开发人员, Tomcat 提供了全部源代码,包括 Servlet 引擎、JSP 引擎、HTTP 服务器等,无论是对哪一方面感兴趣的程序员,都可以从这些由世界顶尖的程序员书写的代码中获得收益。

最后,由于源代码的开放及世界上许多程序员卓有成效的工作, Tomcat 已经可以和大部分的主流服务器一起工作,例如,以模块的形式被载入 Apache,以 ISAPI 的形式被载入 IIS 或 PWS,以 NSAPI 的形式被载入 Netscape Enterprise Server。

除了 JDK 和 JSP 引擎外,一个 JSP Web 应用程序至少还需要一个多用户数据库,当前常见的 Web 数据库连接技术有: CGI 技术、WebAPI 技术、RAD 技术和 JDBC 技术等。Java 中连接数据库的技术是 JDBC (Java DataBase Connectivity)。很多数据库系统带有 JDBC 驱动程序,Java 程序就通过 JDBC 驱动程序与数据库相连,执行查询、提取数据等操作。Sun 公司还开发了 JDBC-ODBC bridge,用此技术 Java 程序就可以访问带有 ODBC 驱动程序的数据库,所以 Java 程序能访问诸如 Oracle、Sybase、MySQL、MS SQL Server 和 MS Access 等数据库。

1-2 JSP 运行平台的建立

前面已经介绍了 JSP 的运行平台主要由 JDK、JSP 引擎和数据库驱动组成,因此建立 JSP 运行平台的主要工作是安装 J2SDK 开发工具和 JSP 引擎,以及配置数据库驱动。

1-2-1 安装 J2SDK 开发工具

J2SDK 是 Java2 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 J2SDK。

首先,在 Windows 下,运行下载的 j2sdk-1_4_2_07-windows-i586-p.exe 文件,根据安装向导安装到一个指定目录,例如,安装到 D:\j2sdk。

接着,需要添加环境变量,保证程序的正确运行,这是安装过程中的重点和难点。

(1) 在 Windows 98 操作系统中,可以用记事本直接编辑 Autoexec.bat (此文件在 C 盘根目录下),添加下面命令行:

```
PATH = %PATH%;D:\j2sdk\bin
SET JAVA_HOME=D:\j2sdk
SET CLASSPATH=D:\j2sdk\lib\tools.jar
```

保存后,重新启动计算机,所添加的环境变量生效。

(2) 在 Windows 2000 或更高版本的操作系统中,按如下方式配置环境变量。右击“我的电脑”图标,在弹出的菜单中选择“属性”命令,弹出“系统属性”对话框,选择“高级”选项卡,如图 1-1 所示。

单击“环境变量”按钮,弹出“环境变量”对话框,如图 1-2 所示,就可以编辑系统的环境变量了。

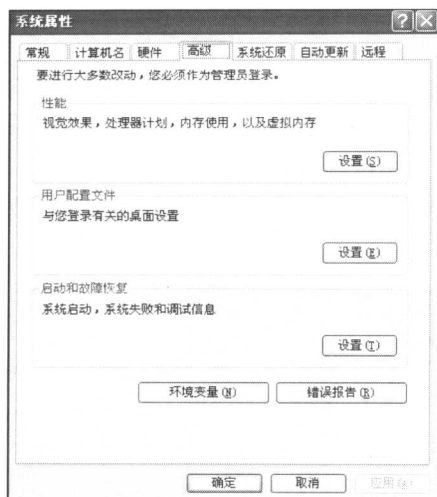


图 1-1 “系统属性”对话框



图 1-2 “环境变量”对话框

添加 PATH、JAVA_HOME 和 CLASSPATH 3 个变量，变量值同上。

配置好环境变量后，打开一个新的 DOS 窗口，输入 java 命令，出现如图 1-3 界面，则 JDK 已经生效了。

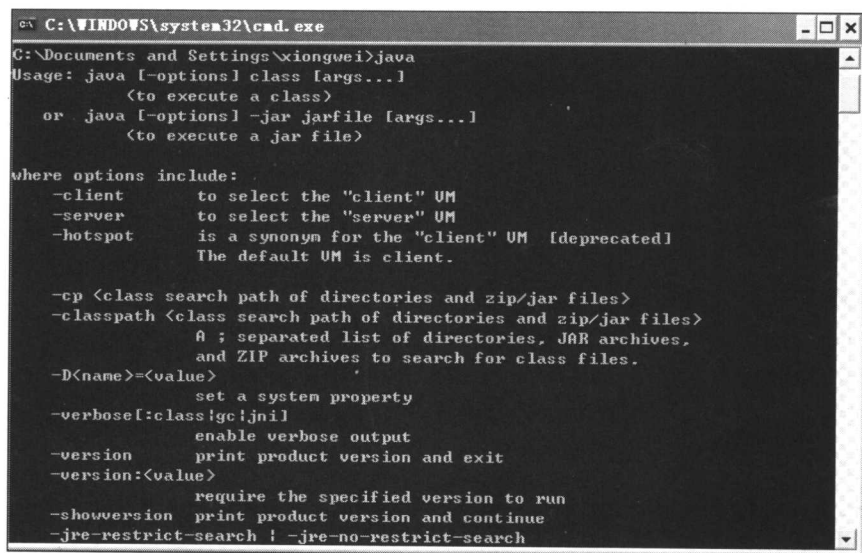


图 1-3 JDK 命令

接着可以在记事本里写一个简单的 Java 程序来测试 J2SDK 是否已安装成功:

```
public class Test {
    public static void main (String args []) {
        System.out.println ("This is a test program.");
    }
}
```

将上面的这段程序命名为 Test.java 并保存（注意，扩展名为.java）。

然后打开 DOS 窗口，进入 Test.java 所在目录，编译和运行源文件。

此时如果输出 This is a test program. 字样的话，说明安装成功了，如果没有输出上述字样，需要仔细检查一下配置情况。

J2SDK 带有一个应用程序控制台，通过它可以直接运行一些程序，非常方便。选择“开始”→“所有程序”→Java Web Start→Java Web Start 命令，即可启动控制台。控制台如图 1-4 所示。

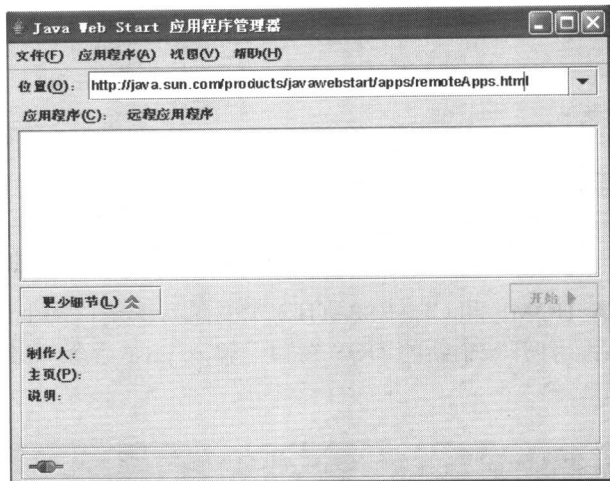


图 1-4 Java Web Start 应用程序管理器

1-2-2 安装 Tomcat

本文以 Tomcat 为例配置 JSP 引擎。运行下载的 jakarta-tomcat-5.0.30.exe，按照一般的 Windows 程序安装步骤即可安装好 Tomcat，安装时它会自动寻找系统中 J2SDK 的位置。假设程序安装到 D:\tomcat5 目录。

接着，同 JDK 一样，需要配置 Tomcat 的环境变量。

右击“我的电脑”图标，在弹出的菜单中选择“属性”命令，弹出“系统属性”对话框，选择“高级”选项卡，单击“环境变量”按钮，弹出“环境变量”对话框，在“系统变量”中添加以下环境变量：

```
TOMCAT_HOME= D:\tomcat5  
CATALINA_HOME= D:\tomcat5
```

然后修改环境变量中的 CLASSPATH，把 Tomcat 安装目录下的 common\lib 下的 servlet.jar 追加到 CLASSPATH 中去，修改后的 CLASSPATH 如下：

```
CLASSPATH=.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar;%CATALINA_HOME%\common\lib\servlet.jar
```

设置完成后，启动 Tomcat 后，打开浏览器，在地址栏中输入 http://localhost:8080（Tomcat 默认端口为 8080），如果在浏览器中看到如图 1-5 所示的 Tomcat 的欢迎界面，表示 Tomcat 工作正常。

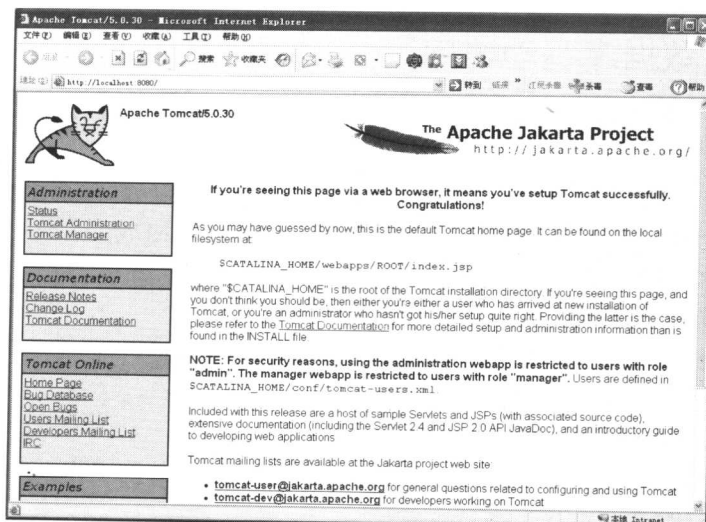


图 1-5 Tomcat 的欢迎界面

我们还可以写一个测试页面来测试默认服务。

准备一个测试用的 JSP 网页，打开文本编辑器，比如记事本，输入下列代码，并保存为 test.jsp（注意扩展名为.jsp）。

```
<HTML>
<HEAD>
<TITLE>JSP 测试页面</TITLE>
</HEAD>
<BODY>
<%out.println("<h1>Hello World! </h1>");;%>
</BODY>
</HTML>
```

把刚才准备好的 test.jsp 放在 D:\tomcat5\webapps\ROOT 目录下，在地址栏中输入 http://localhost:8080/test.jsp，如果出现如图 1-6 所示的页面，说明 JSP 环境安装成功了。

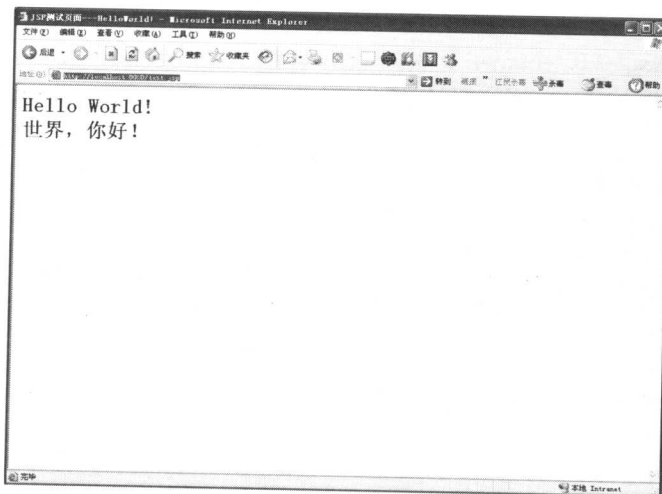


图 1-6 Tomcat 测试页面

1-2-3 安装数据库驱动程序

Java 支持多种数据库系统, 本文以目前使用较多的 MySQL 为例, 介绍 JSP 数据库驱动的配置过程。

MySQL 是一个多用户、多线程的 SQL 数据库, 是一个客户机/服务器结构的具体应用。它由一个服务器守护程序 `mysqld` 和很多不同的客户程序及库组成。

MySQL 包含如下功能特点:

- ◆ 可以同时处理几乎不限数量的用户请求;
- ◆ 处理多达 50 000 000 条以上的记录;
- ◆ 命令执行速度快;
- ◆ 简单有效的用户特权系统。

通常, 要对 MySQL 和 Tomcat 进行整合配置, 才能让其共同工作。

驱动安装的一般流程如下。

1. 程序安装以及驱动下载

MySQL 的安装很简单, 只需按照安装提示进行。安装完成后, 自动启动 MySQL 服务 (MySQL 服务默认随操作系统启动, 并有 `winmysqladmin.exe` 最小化在系统托盘上), 并打开一个命令行界面, 到 MySQL 安装目录下, 执行 `mysql`, 正常的话将会输出一些欢迎信息。

从 <http://download.sourceforge.net/> 下载 MySQL 的 JDBC 驱动 `mm.mysql`。`mm.mysql` 是一个符合 JDBC 2 规范的 MySQL JDBC 驱动程序, 当前版本为 2.0 pre 5 (beta)。假设将刚下载的 `mm.mysql-2.0.4-bin.jar` 文件放到 `D:\j2sdk\lib` 中。

2. 环境变量配置

在系统变量 `CLASSPATH` 中添加下列值:

`D:\j2sdk\lib\mm.mysql-2.0.4-bin.jar`。

3. 整合配置

对 MySQL 与 Tomcat 做相应的配置, 让其配合工作。

(1) 将 `mm.mysql-2.0.4-bin.jar` 复制到 Tomcat 的 `common/lib` 目录下。

(2) 测试是否连接正常; 将如下代码保存成 `test.jsp` 放到 `D:\tomcat5\webapps\ROOT` 目录下。

```
<%@ page contentType="text/html; charset=gb2312" %>
<%
    java.sql.Connection conn;
    java.lang.String strConn;
    Class.forName ("org.gjt.mm.mysql.Driver").newInstance();
    conn = java.sql.DriverManager.
        getConnection("jdbc:mysql://localhost/test","root","");
%>
```

(3) 在浏览器的地址栏中输入 `http://localhost:8080/test.jsp`, 如果运行以后, 出现空白页面, 即表示 MySQL 与 Tomcat 整合成功。

4. 用户自建数据库测试

(1) 启动 Tomcat

(2) 建立数据库及表

首先在 MySQL 里面建立一个测试用表，并且插入几条数据。操作的 SQL 代码如下：

```
create database test;
use test;
create talbe admin(id int(4)auto_increment primary key,name varchar(20));
INSERT INTO user(name)VALUES('test');
```

运行后，就建立了一个数据库 test 和表 user，并且插入了一条记录。

(3) 显示数据库中的记录

将如下代码保存成 test2.jsp 放到 D:\tomcat5\webapps\ROOT 目录下：

```
<%@ page contentType="text/html;charset=GBK" %>
<%@ page language="java" import="java.sql.*"%>
<%
    Connection conn = null;
    Class.forName("org.gjt.mm.mysql.Driver").newInstance();
    conn = java.sql.DriverManager.
        getConnection("jdbc:mysql://localhost/test","root","");
    if(conn==null){
        System.out.println("get Conn Error");
    }
    Statement stmt=conn.createStatement();
    ResultSet RS_result=null;
%>
<html>
<head>
<title>测试</title></head>
<body>
<%
    RS_result=stmt.executeQuery("select * from user");
    String Name;
    while(RS_result.next()){
        Name=RS_result.getString("name");
        %>
        <%=Name%>
        <%
    }
    RS_result.close();
    stmt.close();
    conn.close();
%>
</body>
</html>
```

(4) 运行

打开浏览器，在地址栏中输入 `http://localhost:8080/test2.jsp`，运行后如果页面上显示 `test` 字样，即表示读取数据库成功。

1-2-4 中文乱码问题

在 JSP 中文开发过程中经常遇到乱码问题，许多人为此深为烦恼，下面介绍中文乱码问题的来源和解决方案。

1. 中文问题的来源

计算机最初的操作系统支持的编码方式是单字节字符编码，于是，在计算机中一切处理程序最初都是以单字节编码的英文为准进行处理。随着计算机的发展，为了适应世界其他民族的语言，人们提出了 Unicode 编码，它采用双字节编码，兼容英文字符和其他民族的双字节字符编码。所以，目前大多数国际性的软件内部均采用 Unicode 编码，在软件运行时，它获得本地支持系统（多数是操作系统）默认支持的编码格式，然后再将软件内部的 Unicode 码转化为本地系统默认支持的格式显示出来。Java 的 JDK 和 JVM 即是如此。汉字是双字节编码语言，为了能让计算机处理中文，国家制定了 GB2312、GBK、GBK2K 等标准以适应计算机处理的需求。所以，大部分的操作系统为了适应处理中文的需求，均定制有中文操作系统，它们采用的是 GBK 或 GB2312 编码格式以正确显示汉字。例如，Windows 2000 中文版默认采用的是 GBK 编码显示。

由于 Java 语言内部采用 Unicode 编码，所以在 Java 程序运行时，就存在着一个从 Unicode 编码和对应的操作系统及浏览器支持的编码格式转换输入、输出的问题，这个转换过程有着一系列的步骤，如果其中任何一步出错，显示出来的汉字就会出现乱码，这就是常见的 Java 中文问题。

同时，Java 是一个跨平台的编程语言，使用它编写的程序不仅能在中文 Windows 操作系统上运行，也能在中文 Linux 等操作系统上运行，同时也要求能在英文等操作系统上运行。这种移植操作也会带来中文问题。

此外，如果使用英文的操作系统和英文的 IE 浏览器来运行带中文字符的程序或浏览中文网页时，因为它们本身就不支持中文，也会带来中文问题。并且几乎所有的浏览器默认在传递参数时都是以 UTF-8 编码格式来传递，而不是按中文编码传递，所以，传递中文参数时也会出现乱码现象。

总之，以上几个方面是 Java 中的中文问题的主要来源，把由于以上原因造成的程序不能正确运行的问题称作：Java 中文问题。

2. 中文问题的解决

由于 JSP 在运行时是由 Web 容器进行动态编译的，如果没有指定 JSP 源文件的编码格式，则 JSP 编译器会采用服务器操作系统默认方法来对 JSP 文件进行编译。它在移植时最容易出现问題，例如，在 Windows 2000 中文版中可以运行很好的 JSP 文件复制到英文 Linux 中就无法运行，即使客户端都是一样的。通常此类问题的解决办法如下。

(1) 为保证 JSP 向客户端输出时是采用中文编码方式输出的，在 JSP 源文件头部加入以下一行代码：

```
<%@page contentType="text/html; charset=gb2312"%>
```