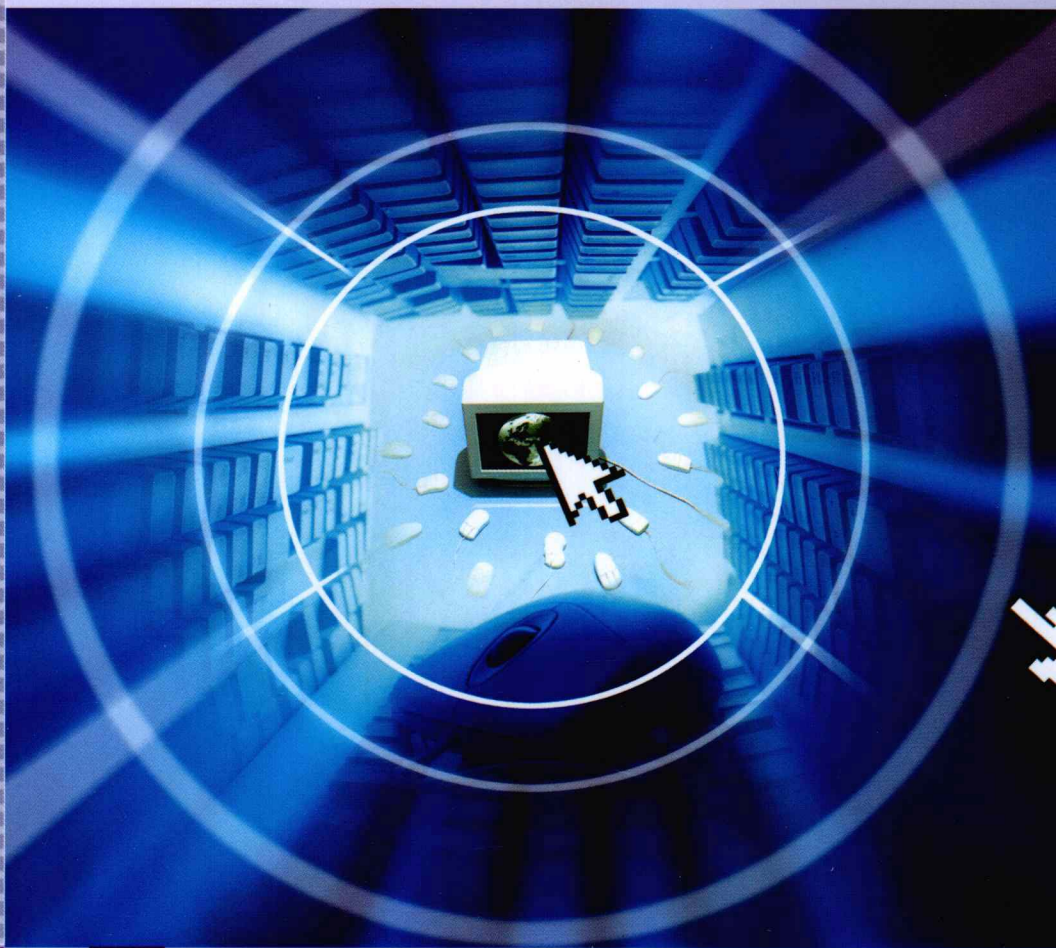




高职高专计算机实用教程系列规划教材

JSP 实用教程

叶若芬 主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE



高职高专计算机实用教程系列规划教材

JSP 实用教程

叶若芬 主 编

秦鹏翔 参 编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书共分7章,内容包括JSP简介、配置JSP运行环境、JSP基本语法、JSP内置对象、数据库操作、在JSP中使用bean、Servlet技术。全书内容遵循“循序渐进”的原则,逐步深入,理论联系实际,注重项目实践,每一章的实例练习有利于培养学生实际运用JSP知识的能力。

本书内容充实,步骤清晰,提供了大量实例,并在每章中配有由“理论知识”和“能力形成”组成的学习目标以及课后小结和复习题,能够全面地培养学生的能力。

本书适合作为高职高专计算机专业或相关专业Web开发课程的教材,也可作为JSP培训班的教材,或者供对JSP感兴趣的读者自学使用。

图书在版编目(CIP)数据

JSP实用教程/叶若芬主编;秦鹏翔编. —北京:中国铁道出版社,2008.6

(高职高专计算机实用教程系列规划教材)

ISBN 978-7-113-08756-2

I. J… II. ①叶…②秦… III. JAVA语言—主页制作—程序设计—高等学校:技术学校—教材 IV. TP393.092

中国版本图书馆CIP数据核字(2008)第092599号

书 名: JSP实用教程

作 者: 叶若芬 主编

策划编辑: 严晓舟 秦绪好

责任编辑: 王占清

封面设计: 付 巍

责任校对: 张 丹

编辑部电话: (010) 63583215

封面制作: 白 雪

责任印制: 李 佳

出版发行: 中国铁道出版社(北京市宣武区右安门西街8号 邮政编码: 100054)

印 刷: 三河市兴达印务有限公司印刷

版 次: 2008年7月第1版 2008年7月第1次印刷

开 本: 787mm×1092mm 1/16 印张: 11.75 字数: 274千

印 数: 5 000册

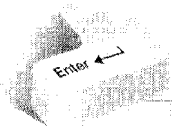
书 号: ISBN 978-7-113-08756-2/TP·2787

定 价: 18.00元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签,无标签者不得销售

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

FOREWORD

自 1999 年 Sun 公司推出 JSP 动态网页技术以来, JSP 就迅速、广泛地用于 Web 上的应用软件开发。在 Internet 上到处可见用 JSP 建立的电子商务、电子政务及各行各业的网站。用 JSP 技术设计的系统, 具有跨平台性、运行效率高等特点, 而且 JSP 学起来容易上手, 其开发工具日渐丰富, 尤其是第三方软件和开源的组件越来越多, 使得 JSP 倍受程序员的青睐。

作为高职高专的教材, 本书坚持循序渐进地介绍基本知识, 通过穿插小实例, 以深入浅出的方法加深读者的理解和应用能力, 带领读者亲自完成多个项目开发。教材中项目为实践中常用的、典型的应用问题, 书中所有实例的代码都经过了调试、运行, 便于读者上机使用, 以达到预习或复习的目的。

本书分为 7 章, 每一章都根据知识的编排强调基础技能或具有完整性和实用性的操作实例, 以便在实践中锻炼提高学生的实际能力。

第 1 章 JSP 简介: 制作第一个 JSP 程序, 通过对程序的讲解突出 JSP 技术的特点, 并简述 JSP 程序的执行过程。

第 2 章 配置 JSP 运行环境: 通过对 JSP 环境的搭架, 掌握 JSP 的基础技能, 包括安装和配置 Web 服务器、部署和运行 JSP 程序、创建自己的 Web 目录、更改 Web 服务器的端口和默认 Web 目录、介绍 JSP 开发工具的使用方法等。

第 3 章 JSP 基本语法: 主要是对 JSP 程序的基本组成元素进行讲解, 通过实例讲解 Java 程序片的插入、JSP 的指令标签和动作标签的作用和使用方法、静态页面的各种控件和组件。本章通过制作留言板系统来训练对 JSP 页面各元素的综合使用, 使读者掌握 JSP 的基本理论和基本技能, 并能在实际中逐步学会应用 JSP 技术。

第 4 章 JSP 内置对象: 使用 JSP 内置对象, 可以方便操作页面, 访问页面环境, 实现页面内、页面间、页面与环境之间的通信; 通过实例讲解了主要内置对象 request、response、session、application、page、out 的作用和使用方法。此外, 本章还配以登录系统实例的内容, 可使读者掌握 JSP 内置对象的实际应用。

第 5 章 数据库操作: 主要介绍页面与数据库之间的通信, 在讲解数据事务处理技术时, 主要结合学生成绩管理系统来讲述数据查询和数据插入、更新及删除操作。本章中以投票系统为实践项目来训练学生的基本技能。

第 6 章 JSP 中使用 bean: 在本章中通过写文件 bean 和读文件 bean 来学习 bean 的规范和使用方法, 并用在线考试系统项目实例来训练学生的实际操作能力。

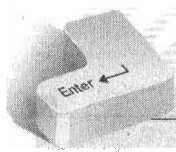
第 7 章 Servlet 技术: 作为 JSP 本质的 Servlet 技术, 主要让学生了解一下它的编程规范和部署、运行和生命周期等。

本书可作为 JSP 开发入门程序员的培训教材和自学教材，书中实例均通过精心调试，完全可以在上机运行书中的代码。为了能够对书中的实践项目得到足够的训练，要求学生至少要有 30 个课时的上机训练时间。

本书由叶若芬任主编，并编写第 1~6 章的内容，秦鹏翔编写了第 7 章的内容。参加本书代码调试工作的还有桑金歌、赵武、于国莉、徐丽媛、李锋、张文娟、刘娜、王磊、李文广、孙书明、刘红娜等。本书的出版得到中国铁道出版社有关领导和编辑的指导与帮助，谨此向他们表示衷心的感谢。

由于时间仓促，编者水平所限，书中疏漏和不足之处在所难免，敬请读者见谅并给予批评指正。

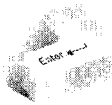
编 者
2008 年 5 月



目 录

CONTENTS

第 1 章 JSP 简介	1
1.1 JSP 概述	1
1.2 第一个 JSP 例子	2
1.3 JSP 的执行过程	3
1.4 JSP 的技术前景	3
1.5 本章小结	3
习题 1	4
第 2 章 配置 JSP 运行环境	5
2.1 JSP 技术支持的架构模型	5
2.2 应用程序服务器	6
2.3 JSP 运行环境	6
2.4 安装 J2SDK 和配置环境变量	8
2.5 安装 Tomcat	9
2.6 Tomcat 的目录结构	11
2.7 JSP 页面的执行流程	14
2.8 本章小结	15
习题 2	15
第 3 章 JSP 基本语法	17
3.1 Java 程序片	17
3.2 声明	19
3.3 表达式	21
3.4 程序段	22
3.5 JSP 指令	23
3.6 JSP 动作	27
3.7 字符使用惯例	31
3.8 JSP 异常	32
3.9 HTML 标记	32
3.10 综合编程实例	34
3.10.1 留言设计原理	35
3.10.2 输入留言	35
3.10.3 保存留言	36
3.10.4 查看留言	39

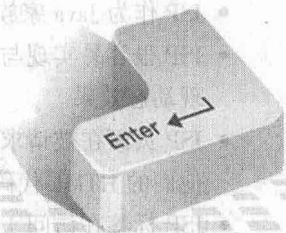


3.11 本章小结	40
习题 3	41
第 4 章 JSP 内置对象	42
4.1 request 对象	42
4.2 response 对象	49
4.2.1 改变浏览器文件输出类型	50
4.2.2 控制网页的刷新频率	52
4.3 session 对象	53
4.3.1 显示客户会话 ID	54
4.3.2 信息的保存和获取	58
4.3.3 计数器	61
4.4 application 对象	62
4.5 page 对象	64
4.6 out 对象	65
4.7 用户登录系统	66
4.8 本章小结	71
习题 4	71
第 5 章 数据库操作	73
5.1 JDBC 简介	73
5.2 JSP 中的 JDBC	74
5.3 JDBC 驱动	74
5.4 数据库事务处理	78
5.5 数据查询	79
5.5.1 顺序查询	79
5.5.2 游动查询	81
5.5.3 随机查询	85
5.5.4 参数查询	87
5.5.5 排序查询	94
5.5.6 使用通配符查询	97
5.6 数据更新	101
5.6.1 修改数据	101
5.6.2 添加数据	106
5.6.3 删除数据	112
5.7 网上投票	116
5.8 本章小结	124
习题 5	124

第 6 章 在 JSP 中使用 bean	126
6.1 什么是 bean?	126
6.2 编写 bean.....	127
6.3 JSP 使用 bean	132
6.4 访问 bean 属性	134
6.4.1 读取 bean 属性的举例	135
6.4.2 设置 bean 属性的举例	136
6.5 bean 的作用域	142
6.6 bean 的综合运用实例	147
6.6.1 读文件 bean	147
6.6.2 写文件 bean	153
6.6.3 查询数据库 bean	158
6.6.4 添加记录	161
6.7 标准考试 bean	166
6.8 本章小结	171
习题 6	172
第 7 章 Servlet 技术	173
7.1 什么是 Servlet.....	173
7.2 Servlet 技术的特点	174
7.3 Servlet 的生命周期	174
7.4 开发部署一个简单的 Servlet	175
7.5 JSP 与 Servlet 之间的关系	178
7.6 本章小结	179
习题 7	179
参考文献	180

第 1 章

○ JSP 简介



Java Server Pages (简称 JSP) 技术是由 Sun 公司倡导、经多家公司参与, 于 1999 年推出的一种创建动态 Web 网页的简单方法, 它是基于 Java Servlet 的 Web 开发技术, 利用这一技术可以开发动态的、高性能的 Web 应用程序。在 HTML 文件中加入 Java 程序片段和 JSP 标记, 就构成了 JSP 网页。本章站在开发者的角度对 JSP 做了全面的介绍, 其中包含一些简单的 JSP 实例。

主要内容

JSP 概念、简单 JSP 程序的开发和 JSP 程序的运行过程。

学习目标

理论知识 理解什么是 JSP 技术, 了解 JSP 页面的组成和 JSP 页面的执行过程。

能力形成 结合静态网页知识和 Java 基础知识会开发简单的 JSP 页面。

1.1 JSP 概述

JSP 技术支持开发者和设计者在现有业务基础上轻松地进行开发和维护工作。作为 Java 家族的一部分, JSP 实现了基于 Web 应用程序平台的独立性, 实现了前台用户界面和后台程序之间的分离, 使网页设计人员可以在不改变 JSP 程序的情况下更改前台用户界面。

那么 JSP 页面到底是什么? 就其基本形式而言, JSP 页面也是一种 HTML 页面, 只不过它包含了用于产生动态网页的代码, 这样的附加代码可能是 Javabean、JDBC 对象、EJB 对象或者远程方法调用 (RMI) 对象。例如, 一个 JSP 页面可能包含了用于产生静态网页文本或者图形的 HTML 代码, 同时也包含了用于调用 JDBC 对象访问数据库的方法。当网页在客户端浏览器上显示时, 其中既包含了静态的 HTML 文本, 也包含了从数据库中获得的信息。

用户界面与逻辑程序的分类使得网页设计者和 JSP 程序开发者之间的任务十分清晰, 即使很复杂的 JSP 代码也可以很容易地被升级或重用。由于 JSP 页面在需要访问时会被自动编译, 因此网页设计者在修改静态网页内容后并不需要重新编译 JSP 程序, 这也使得 JSP 比 Java Servlet 更加容易实现网页的动态化。本质上, JSP 是 Servlet API 的一种扩展, Servlet 所产生的 Web 页面不能



包含在 HTML 标签中,它离不开 Java 类文件的支持,而修改类文件需要重新编译 Servlet 程序。JSP 在使用前必须被编译为 Servlet 程序,因此 JSP 具有 Java 的特点,如对 Java API 的访问。使用 JSP 将带来很多前所未有的优势:

- JSP 作为 Java 家族的一部分,同样拥有 Java 语言的“一次编写、到处运行”的特点。
- JSP 很容易实现与静态网页模板的结合,例如与 HTML 或 XML 页面结合,然后加入代码实现动态效果。
- JSP 页面在被请求访问时自动编译成 Servlet 程序,使得静态网页的设计者可以放心地更新他们的 HTML 代码。当然, JSP 页面也可以在需要时进行预编译。
- JSP 标签在访问 Javabeen 组件的同时能够很好地管理这些组件,对静态网页设计者而言,这些事情虽然复杂但却是透明的。
- JSP 开发者可以定制 JSP 标签,静态网页设计者可以像使用 XML 那样使用这些标签。
- 静态网页设计者可以在不影响 JSP 动态程序的基础上改变静态网页部分内容。同样, JSP 程序设计者也可以在包含 JSP 程序的页面上修改 JSP 代码,但不影响静态网页部分。

总而言之,静态网页设计者无须了解 Java 语言或 Servlet 的任何知识,他们只需专注于 HTML 代码的编写;而 Java 程序设计专家只需要负责创建 JSP 应用程序。

1.2 第一个 JSP 例子

以下是 JSP 网页的例子,其中包括了静态 HTML 内容和 Java 程序代码片段,它会显示当前的执行时间和程序执行 2 000×2 000 次循环后的时间,该程序只用不到 3s 时间运行就结束。程序代码如下:

```
Test.jsp
<%@page contentType="text/html;charset=GBK"%>
<%@page import="java.util.*,java.text.*"%>
<html>
<head>
<title> 测试 JSP 运行时间</title>
</head>
<body>
<%
Date now=new Date();
out.println("执行前的时间为: "+DateFormat.getTimeInstance().format(now));
int count=0;
for(int i=0;i<2000;i++)
{
    for(int j=0;j<2000;j++)
    {
        count++;
    }
}
out.println("程序总共循环了: <font color='red'>"+count+"</font>次<br>");
Date now1=new Date();
out.println("执行后的时间为: "+DateFormat.getTimeInstance().format(now1));
%>
```

```

</body>
</html>

```

1.3 JSP 的执行过程

在学习 JSP 知识之前,首先了解一下 JSP 的执行过程,这对以后的学习会有很好的启发作用。JSP 的执行过程如图 1-1 所示。

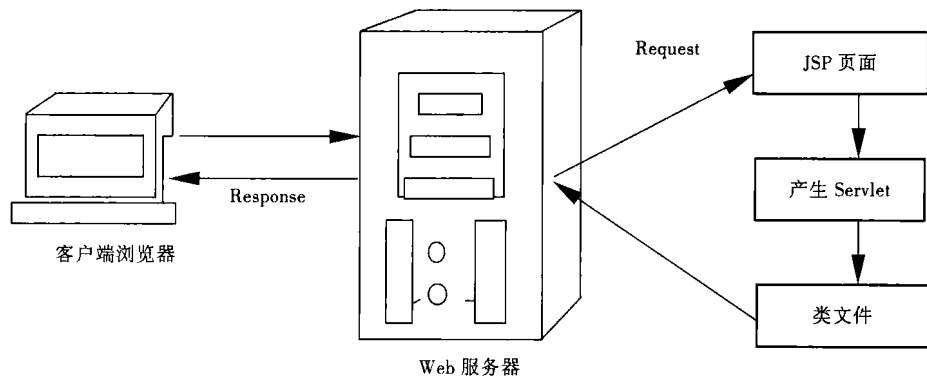


图 1-1 JSP 执行过程

首先,由客户端浏览器向 Web 服务器提出访问 JSP 页面的请求 (Request),然后由 JSP 引擎负责将 JSP 转换成 Servlet,产生的 Servlet 经过编译后生成类文件,然后再把类文件加载到内存执行。最后,由 Web 服务器将执行结果即响应 (Response) 给客户端浏览器。

JSP 需要编译成 Servlet,所以 JSP 的执行效率要低于 Servlet,但这仅限于第一次执行。由于 JSP 在第一次执行后编译生成类文件,当再次重复调用时,如果 JSP 引擎没有发现这个 JSP 页面有修改的痕迹,就会直接执行编译后的类文件而无须重新编译 Servlet。因此,除了第一次的编译会花比较久的时间外,之后的 JSP 和 Servlet 的执行速度几乎相同。当然,如果 JSP 引擎检查到 JSP 页面被修改过,则需要重新进行编译执行。

1.4 JSP 的技术前景

目前,在国内 PHP 与 ASP 应用较为广泛。在国内外,JSP 已经是比较流行的一种技术,尤其是电子商务类的网站,多采用 JSP。三者中 JSP 应该是未来电子商务网站发展的趋势。世界上一些大型的电子商务解决方案提供商都采用 JSP/Servlet 方案,比较出名的如 IBM 的 E-business,它的核心是采用 JSP/Servlet 的 WebSphere。它们都是通过 CGI 来提供支持的。

1.5 本章小结

JSP 是 Sun 公司于 1999 年推出的一种 Web 开发技术。JSP 网页由 HTML 标记、JSP 标签和 Java 程序片段组成。本章为读者介绍 JSP 的基本特点和作用,通过一个 JSP 的例子,使读者有了感性认识。最后,介绍了 JSP 的执行过程,这对读者以后的学习具有指导性意义。



习 题 1

一、填空题

1. JSP 推出的时间是_____年。
2. JSP 网页包含三种元素，它们是_____、_____和_____。
3. JSP 的全称是_____。
4. JSP 的优点之一是可以使得_____和_____相分离。
5. JSP 本质上是对_____的扩展。

二、判断题

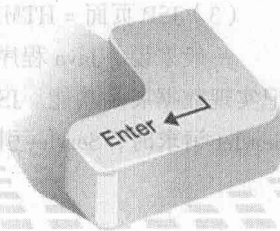
1. JSP 和 Java 一样具有平台独立性。
2. JSP 在运行之前必须转换成 Servlet。
3. JSP 网页中一定包含 JSP 程序。
4. JSP 第一次执行的速度要比第二次执行的速度慢。
5. 运行 JSP 程序只需要浏览器即可。

三、简答题

1. JSP 出自哪家公司？
2. 列举 JSP 的主要优点。
3. 简述 JSP 的执行过程。

第 2 章

配置 JSP 运行环境



在第 1 章中提到，JSP 是一种 Web 网页开发技术，为了熟悉这种技术，必须了解 JSP 技术支持什么样的体系结构，需要运行在什么样的需要平台上，需要通过什么协议实现通信等。

我们开发自己的 JSP 程序，首先配置好 JSP 运行环境，选择好 JSP 开发工具。现今支持 JSP 应用程序的服务器非常多，配置方法也各不相同，因此我们选取了最有代表性的 JSP 平台作为本书中所有 JSP 应用程序的编译、调试和运行平台。本书选用 Apache Tomcat 6.0 作为 JSP 的 Web 服务器平台，JDK 使用的是 1.5 版本。

主要内容

JSP 技术支持的架构体系；JSP 程序的运行环境和开发环境；J2SDK 软件和 Tomcat 软件的安装与配置；JSP 页面的执行流程。

学习目标

理论知识 明确 JSP 技术支持的体系结构、JSP 程序的运行环境和 JSP 页面的执行流程。

能力形成 能够安装和配置 J2SDK 开发包和 Tomcat 服务器，正确部署和运行 JSP 的页面程序。

2.1 JSP 技术支持的架构模型

JSP 技术开发的程序架构只能是 C/S 结构或 B/S 结构。下面是 JSP 技术支持的三层体系结构模型，如图 2-1 所示。

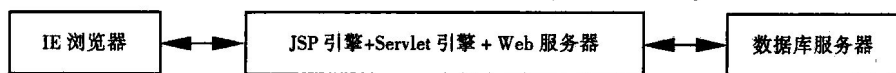


图 2-1 JSP 技术支持的三层模型

通常情况下，JSP 页面由展现用户界面的 HTML 标记和数据计算两部分组成，因此数据展示代码和数据计算代码可能会放在同一个 JSP 页面上，它们都会部署在 Web 服务器端。



JSP 页面有如下三种形式：

- (1) JSP 页面 = HTML 标记+Java 程序片
- (2) JSP 页面 = HTML 标记+Servlet 模块
- (3) JSP 页面 = HTML 标记+bean

一般来说，Java 程序片、Servlet 模块实现逻辑计算功能，bean 实现数据处理功能，HTML 标记实现数据展示功能。JSP 页面中的 Java 程序片最终被 JSP 引擎转译为 Servlet 模块，当客户发出 Servlet 请求时，Servlet 引擎将这些 Servlet 模块载入内存运行，以处理客户请求。

2.2 应用程序服务器

上面介绍了程序的体系结构，下面将讨论程序主要运行的平台，即应用程序服务器的组成和作用。JSP 技术支持的应用程序服务器一般由 JSP 引擎、Servlet 引擎和 Web 服务器组成。JSP 引擎将 JSP 页面转译并编译为相应的字节码文件（Servlet 应用模块），Servlet 引擎将客户的请求传递给相应的 Servlet 应用模块；Web 服务器接受客户的请求，并把处理的结果返回给客户。

Web 服务器是一种请求/响应模式的服务器。即由客户机向服务器提出请求，服务器接到请求后，进行处理，并将处理的结果返回给客户机（响应）。客户机与服务器间的通信协议是遵循 HTTP 协议。

1. 请求/响应模式

请求/响应模式如图 2-2 所示。

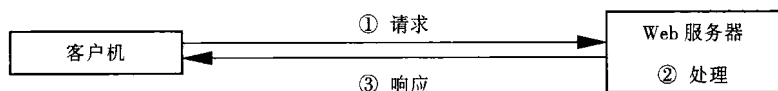


图 2-2 Web 服务器的请求/响应模式

2. 流行的 Web 服务器

Web 服务器有很多，流行的 Web 服务器有 Weblogic、WebSphere 和 Tomcat 三种。这三种服务器都带有 JSP 引擎和 Servlet 引擎。

JSP 引擎和 Servlet 引擎都是系统模块（为应用程序提供服务的模块），也属于 Servlet 模块，它们随着 Web 服务器启动载入内存，随着 Web 服务器的关闭而释放。Servlet 模块分两类：一类是应用 Servlet，它是 JSP 页面转译并编译的结果，即应用程序员编写的 Servlet 程序；另一类是系统 Servlet，如 JSP 引擎和 Servlet 引擎。

JSP 引擎的作用是当客户向服务器发出 JSP 页面请求时，将 JSP 页面转译为 Servlet 源代码，然后调用 javac 命令，把 Servlet 源代码编译为相应的字节码，并保存在相应目录中。

Servlet 引擎的作用是管理和加载应用 Servlet 模块。当客户向相应的应用 Servlet 发出请求时，Servlet 引擎把应用 Servlet 模块载入虚拟机运行，由应用 Servlet 处理客户请求，将处理结果返回客户。

2.3 JSP 运行环境

整个 JSP 运行环境主要包括 JSP 执行环境和 JSP 开发环境。JSP 属于 Java 家族，因此它的运

行首先离不开 Sun 公司的 J2SDK；另外作为应用服务器，这里以典型的安装软件 Tomcat 为例加以介绍；接下来对 JSP 程序的开发工具 Jcreator、Editplus 等软件进行简要介绍。

J2SDK 全称 Java 2 Software Development Kit，它是由 Sun MicroSystem 公司提供的 Java 开发工具。J2SDK 在 Java 1.1 以前称作 JDK，Java 1.2 版本以后改名为 J2SDK，但习惯上仍有很多人称 JDK。J2SDK 是学习 Java 开发（包括 JSP 开发）的初始环境，其他开发环境均构建在此环境之上。

J2SDK 分成标准版（J2SE）、企业版（J2EE）和微型版（J2ME）。其中标准版可以从 Sun 官方网站 <http://java.sun.com> 免费下载，目前的最新版本为 Java 1.5。本书将使用 Windows 操作系统环境下的标准版 jdk-1_5_0_04-windows-i586-p。

Tomcat 平台原先是 Sun 公司借助 Apache 开发团队的影响来开发服务器端的一种 Java 技术，完全符合 Servlet 2.2 和 JSP 1.1 规范。Tomcat 一度作为 Java 爱好者个人使用而非面向商业领域的平台，但由于 Tomcat 出色的性能表现和不断的功能更新，已经有不少开发者把它用于商业用途了。由于 Tomcat 是基于 Java 的，因此它的正常运行也离不开 JDK。

访问 Apache 官方网址 <http://jakarta.apache.com> 可以下载到最新版本的 Tomcat，目前的版本为 apache-tomcat-6.0.13.exe，它是基于 JDK 1.5 及更高版本设计的，也是本书使用的版本。

支持 JSP 开发的工具有很多，较为知名的有 Borland JBuilder、IBM WebSphere、Dreamweaver MX 等，但这些工具大都为企业应用而设计，对系统的条件都有严格的要求，例如 JBuilder 8.0 推荐系统内存为 1 GB，至少也需要 256 MB。如果你的机器达不到这样的要求，最好还是不要用它。如果你的硬件条件满足了，面对这些大型工具复杂的 IDE 界面，你会感到学习 JSP 的门槛又高了一层。

JCreator 是 XINOX 软件公司推出的一款轻型 Java 开发工具，完全基于 C++ 设计，运行效率高、速度快，对系统要求远低于上述 Java IDE 工具。用来学习基于 Java 的开发，JCreator 是一个很好的选择。但是 JCreator 对 JSP 这种 Web 动态网页却支持不够。JCreator 可以从 <http://www.jcreator.com> 下载得到，其运行界面如图 2-3 所示。

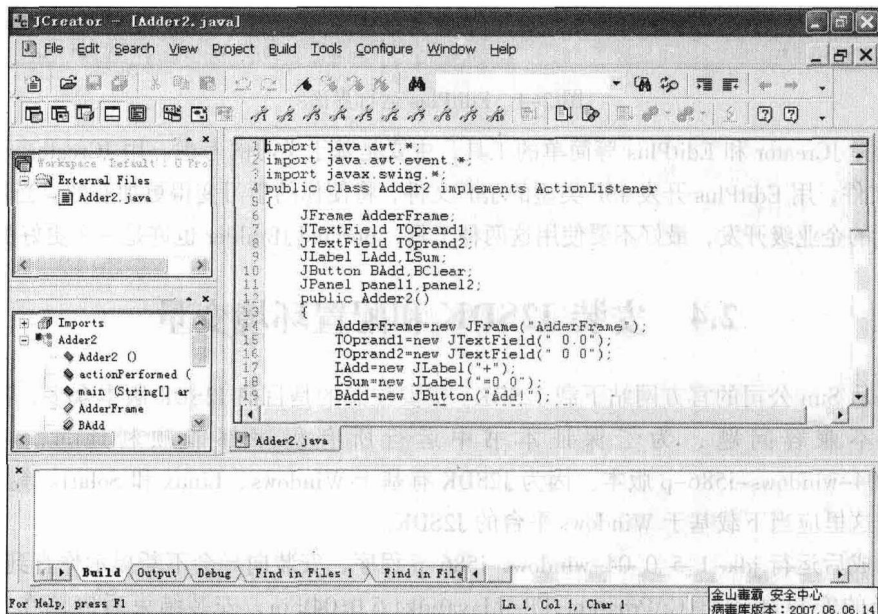


图 2-3 JCreator 运行界面



使用 JCreator 主要是考虑到在 JSP 开发时需要用到一些 Java Class 文件（例如 Javabeans、Servlet、实用的 Java 类等），这些 Java Class 文件使用 JCreator 来进行开发、编译、调试等操作都很方便。

其他类似于 JCreator 的开发工具，例如 JPad、JRun 等，也是很好的选择，读者都可以尝试用一下。

EditPlus 是一款出色的编辑工具，对语法有高彩显示的功能，对于 JSP 初学者而言，最好能使用这样的编辑工具，可以真正快速地熟悉 JSP。登录 <http://www.editplus.com> 就可以下载到 EditPlus，其运行界面如图 2-4 所示。可以看到 EditPlus 还具有文件管理的功能，如果你的应用程序中存在大量 JSP 或 Java 文件，这将是一项非常有用的功能。

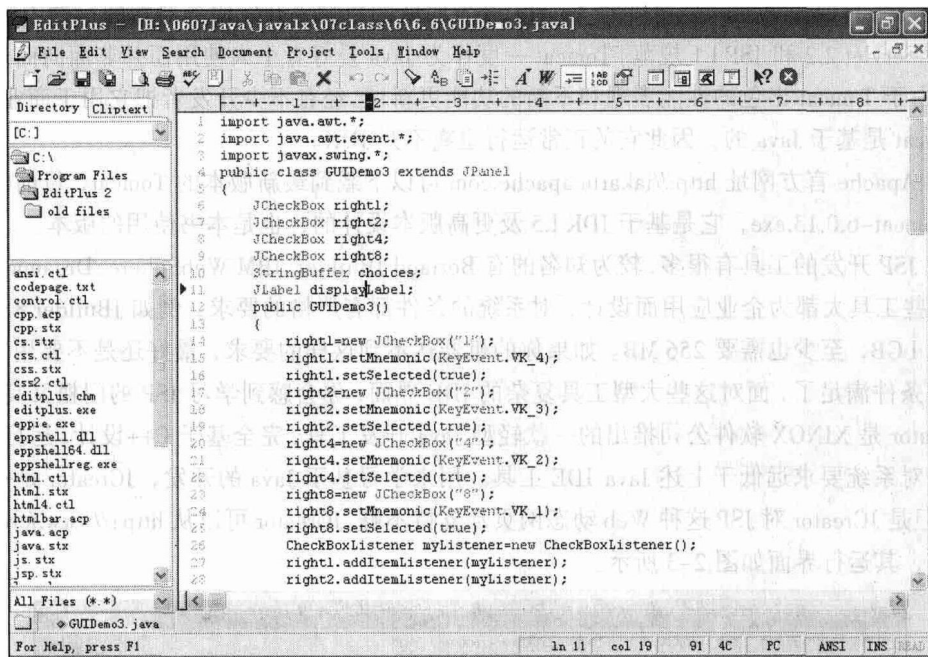


图 2-4 EditPlus 运行界面

这里选择 JCreator 和 EditPlus 等简单的工具，主要是基于学习的考虑。用 JCreator 开发 Java 类型的 bean 文件，用 EditPlus 开发 JSP 类型的 JSP 文件，将使你的学习变得更加有效。当然，如果进行基于 Java 的企业级开发，最好不要使用这两种工具，Borland JBuilder 也许是一个更好的选择。

2.4 安装 J2SDK 和配置环境变量

首先是从 Sun 公司的官方网站下载 J2SDK。需要注意的是目前 J2SDK 版本众多，各版本之间还存在着不兼容问题，为了保证本书中运行所有程序都能顺利通过，建议下载 `jdk-1_5_0_04-windows-i586-p` 版本。因为 J2SDK 有基于 Windows、Linux 和 Solaris 等操作系统的不同版本，这里应当下载基于 Windows 平台的 J2SDK。

下载完成后运行 `jdk-1_5_0_04-windows-i586-p` 程序，安装向导会不断提示你直到安装完成，如果你选择的安装目录是 `C:\Program Files\Java\jdk1.5.0_04\bin`，安装结束后应该在 `C:\Program Files\Java\jdk1.5.0_04\bin` 目录下出现如图 2-5 所示文件。

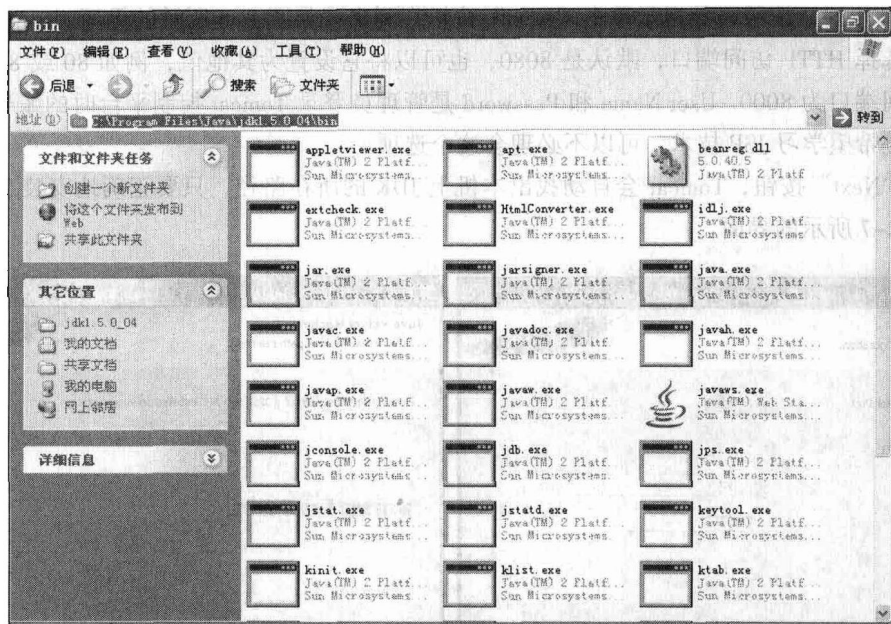


图 2-5 Java 安装目录下的文件

在 C:\Program Files\Java\jdk1.5.0_04\bin 目录中包含了 javac.exe、java.exe 等许多可执行文件。Javac.exe 就是 Java 编译器，用来把 Java 文件编译成字节码。在 MS-DOS 环境下，可以使用“javac Java 文件名”的形式来编译一个 Java 文件。

Java.exe 是用来运行编译后的.class 文件，被称为 Java 直译器。如果需要运行 jar 文件（一种封装程序，把所有相关的文件封装在一起），则需要 -jar 命令。例如，在 MS-DOS 命令行中输入“java -jar C:\JDK \ demo \ jfc \ Notepad \ Notepad.jar”，这样就可以运行 Notepad.jar 文件。

命令符是运行 java 程序的最基本的方法，它多少显得有些不便，只能通过输入字母来执行命令。但是这种运行方法有助于了解 J2SDK。在安装 Jcreator 以后就可以直接通过单击“编译”或“执行”按钮来执行你的命令，这将会方便很多。

安装完 J2SDK 后，应该配置环境变量，其中配置环境变量目的有三个：第一，让操作系统自动查找编译器、解释器的所在路径；第二，设置程序编译和执行时需要的类路径；第三，Tomcat 服务器安装时需要知道虚拟机的所在路径。

配置环境变量的步骤：

右击“我的电脑”图标，在快捷菜单中选择“属性”选项，弹出“系统属性”对话框，选择“高级”标签，并单击“环境变量”按钮，在“系统变量”对话框中单击“新建”按钮，建立 path 和 classpath 环境变量，若已经存在这两个环境变量，可设置环境变量的值，在环境变量值中添加下面的内容即可。

```
path=.;C:\j2sdk1.5.2\bin
classpath=.;C:\j2sdk1.5.2\lib
```

2.5 安装 Tomcat

下载完 apache-tomcat-6.0.13.exe（也可以是 Tomcat 5.0 以上的其他版本），运行 japache-