

高等院校“十三五”应用技能培养规划教材
• 移动应用开发系列

JSP 编程技术

徐天凤 李桂珍 郭洪荣 主 编
徐嵩松 侯小毛 刘 源 刘儒香 副主编

- 本书各章采用基础知识—实例讲解—案例实训的编写模式
- 内容全面，选取经典实用案例，紧密结合内容呈现清晰讲解
- 免费赠送电子课件、程序源代码、习题答案及扩展综合案例的完整操作讲解

清华大学出版社



高等院校“十三五”应用技能培养规划教材·移动应用开发系列

JSP 编程技术

徐天凤 李桂珍 郭洪荣 主 编
徐嵩松 侯小毛 刘 源 刘儒香 副主编



清华大学出版社
北 京

内 容 简 介

JSP 是一种动态网页技术标准, 由 Sun 公司开发, 可以运行在 Windows、UNIX、Linux 操作平台上。本书共分为 12 章, 内容包括: JSP 运行环境的安装与配置、JSP 的语法、JSP 的内置对象、JavaBean 技术、Servlet 技术、JSP Servlet 的 MVC 模式、表达式语言、JSP 与 JDBC、JSP 中的文件操作、JSP 的 XML 文件处理、JSP 与 MySQL 数据库操作, 最后一章以网上书店系统的案例讲解了 JSP 编程的实际应用。

本书案例丰富, 配合知识要点讲解, 语言通俗, 讲解关系层层递进, 可读性强。通过学习本书内容, 读者能够轻易并牢固地掌握 JSP 的相关实用技能。本书既可以作为应用型高等院校或高职高专院校计算机及相关专业的教材, 又可作为计算机编程从业人员的专业指导用书。

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。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

JSP 编程技术/徐天凤, 李桂珍, 郭洪荣主编. —北京: 清华大学出版社, 2018

(高等院校“十三五”应用技能培养规划教材·移动应用开发系列)

ISBN 978-7-302-49290-0

I. ①J… II. ①徐… ②李… ③郭… III. ①JAVA 语言—网页制作工具—程序设计—高等学校—教材 IV. ①TP312.8 ②TP393.092

中国版本图书馆 CIP 数据核字(2018)第 001416 号

责任编辑: 汤涌涛

装帧设计: 杨玉兰

责任校对: 王明明

责任印制: 沈 露

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62791865

印 装 者: 三河市金元印装有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 20.75 字 数: 500 千字

版 次: 2018 年 5 月第 1 版 印 次: 2018 年 5 月第 1 次印刷

印 数: 1~2000

定 价: 49.80 元

产品编号: 077540-01

前 言

JSP 是由 Sun 公司开发的一种动态网页技术标准,是基于 Java Servlet 以及整个 Java 体系的 Web 开发技术,利用这一技术可以建立面向对象、安全、跨平台的动态网站。随着 Java 技术的不断提升, JSP 也在逐步发展, JSP 能够将页面设计与后台代码分离,提高了工作效率。目前, JSP 已成为动态网站开发不可缺少的开发工具。

本书共分为 12 章,具体内容介绍如下。

第 1 章: JSP 基本概述。主要介绍 JSP 的定义、特点、工作流程、组成元素以及相关的安装与配置。

第 2 章: JSP 基础语法知识。主要介绍 JSP 的基本语法(注释、声明、代码段、表达式)、JSP 程序开发模式、调试处理、JSP 指令标记和动作标记的使用。

第 3 章: JSP 的内置对象。主要介绍 JSP 中的 application、request、response、pageContext、session、out、config Servlet、page JSP、exception 9 个内置组件的应用。

第 4 章: JavaBean 技术。主要介绍 JavaBean 的定义、工具、规范、属性、事件应用。

第 5 章: Servlet 技术。主要介绍 Servlet 的定义、特点、生命周期、类的方法、跳转的使用、异步处理。

第 6 章: JSP Servlet 的 MVC 模式。主要介绍 JSP Servlet 的 MVC 模式,包括模型的生命周期与视图更新、注册登录、与数据库的连接以及文件操作。

第 7 章: 表达式语言。主要介绍 EL 表达式的定义及特点,以及 EL 表达式的语法、运算规则、内置对象。

第 8 章: JSP 与 JDBC。主要介绍 JDBC 的定义、产品组件、建立 JDBC 连接、JDBC 包。

第 9 章: JSP 中的文件操作。主要介绍 JSP 的文件操作,包括 File 类、字节流读/写文件、RandomAccessFile 类、文件的上传或下载。

第 10 章: JSP 的 XML 文件处理。主要介绍 XML 的定义、用途、基本语法、命名规则、元素的定义、XML 的解析方法(DOM、SAX、DOM4j)。

第 11 章: JSP 与 MySQL 数据库操作。主要介绍 MySQL 数据库的安装、配置、基础操作以及连接 JSP 的基本操作。

第 12 章: 网上书店系统设计。主要利用 JSP 开发一个网上书店系统,从用户登录、用户选书,到提交订单的每个功能模块的代码都进行了全面详细的分析。

本书由徐天凤、李桂珍、郭洪荣担任主编,由徐嵩松、侯小毛、刘源、刘儒香担任副主编。其中,徐天凤编写第 1、2、4、5 章;李桂珍编写第 3、8、9 章;侯小毛编写第 6 章;郭洪荣、刘源、刘儒香编写第 7、11 章;徐嵩松编写第 10、12 章。本书内容根据易学、易懂、易掌握的原则,结合 JSP 知识体系,由浅入深、循序渐进地进行讲解。同时还将 JSP 知识与案例有机地结合起来,使知识与案例相辅相成。

由于作者水平有限,书中难免有疏漏和不妥之处,敬请业内专家、同行以及广大读者提出宝贵意见,以便今后不断改进。

编 者

目 录

第 1 章 JSP 基本概述.....1	
1.1 了解 JSP 技术.....2	
1.1.1 什么是 JSP2	
1.1.2 JSP 的特点与工作流程3	
1.1.3 JSP 与类似语言技术的比较4	
1.1.4 JSP 页面的组成5	
1.1.5 JSP 页面中的元素6	
1.2 JSP 的安装与配置.....6	
1.2.1 JDK 的安装与配置6	
1.2.2 Tomcat 的安装与启动8	
1.2.3 Eclipse 的安装与使用.....9	
1.3 案例：编写 HelloWorld.jsp 文件并试运行.....11	
本章小结.....12	
习题.....12	
第 2 章 JSP 基础语法知识14	
2.1 JSP 语法注释声明.....15	
2.1.1 语法注释15	
2.1.2 声明19	
2.1.3 代码段21	
2.1.4 表达式22	
2.2 JSP 程序开发模式.....23	
2.2.1 单纯的 JSP 编程24	
2.2.2 JSP+JavaBean 编程.....24	
2.2.3 JSP+JavaBean+Servlet 编程25	
2.2.4 MVC 模式26	
2.2.5 运行 JSP 时常见的出错信息及处理26	
2.3 JSP 的指令.....27	
2.3.1 page 指令.....27	
2.3.2 include 指令.....33	
2.4 JSP 的动作.....33	
2.4.1 <jsp:include>动作标记34	
2.4.2 <jsp:param>动作标记36	
2.4.3 <jsp:forward>动作标记.....38	
2.4.4 <jsp:plugin>动作标记39	
2.4.5 <jsp:useBean>动作标记42	
2.4.6 <jsp:setProperty>动作标记44	
2.4.7 <jsp:getProperty>动作标记45	
2.5 案例：JSP 指令标记.....47	
本章小结.....48	
习题.....48	
第 3 章 JSP 的内置对象50	
3.1 application 对象.....51	
3.1.1 查找 Servlet 有关的属性信息.....51	
3.1.2 管理应用程序属性.....52	
3.2 out 对象.....53	
3.2.1 向客户端输出数据.....53	
3.2.2 管理输出缓冲区.....53	
3.3 request 对象.....54	
3.3.1 获取客户信息.....55	
3.3.2 获取请求参数.....56	
3.3.3 获取查询字符串.....58	
3.3.4 在作用域中管理属性.....59	
3.3.5 获取 Cookie60	
3.3.6 访问安全信息.....62	
3.3.7 访问国际化信息.....62	
3.4 response 对象.....62	
3.4.1 动态设置响应的类型.....63	
3.4.2 重定向网页.....64	
3.4.3 设置页面自动刷新以及定时跳转.....65	
3.4.4 配置缓冲区.....66	
3.5 session 对象.....67	
3.5.1 创建及获取客户会话属性.....68	
3.5.2 从会话中移除指定的对象.....69	
3.5.3 设置会话时限.....70	



3.6 其他内置对象.....	71
3.6.1 pageContext 对象.....	71
3.6.2 page 对象.....	73
3.6.3 config 对象.....	74
3.7 案例：显示字符串长度.....	74
本章小结.....	75
习题.....	75
第 4 章 JavaBean 技术.....	77
4.1 JavaBean 的定义与规范.....	78
4.1.1 什么是 JavaBean.....	78
4.1.2 JavaBean 工具.....	78
4.1.3 JavaBean 规范.....	79
4.2 JavaBean 的属性与事件.....	80
4.2.1 JavaBean 的属性.....	80
4.2.2 JavaBean 的 Scope 属性.....	83
4.2.3 JavaBean 事件.....	85
4.3 案例：JavaBean 实现用户登录界面 ...	90
本章小结.....	94
习题.....	94
第 5 章 Servlet 技术.....	96
5.1 Servlet 概述.....	97
5.1.1 Servlet 的定义和特点.....	97
5.1.2 Servlet 的生命周期.....	98
5.1.3 Servlet 的类和方法.....	99
5.2 Servlet 的跳转与使用.....	102
5.2.1 客户端跳转.....	102
5.2.2 服务器跳转.....	103
5.2.3 获取客户端信息.....	104
5.2.4 过滤器.....	107
5.2.5 监听器.....	111
5.3 异步处理.....	115
5.3.1 什么是 AsyncContext.....	115
5.3.2 模拟服务器推送.....	117
5.4 案例：通过表单向 Servlet 提交	
数据.....	121
本章小结.....	123
习题.....	123

第 6 章 JSP Servlet 的 MVC 模式.....	125
6.1 模型的生命周期与视图更新.....	126
6.1.1 MVC 的定义.....	126
6.1.2 request 周期的 JavaBean.....	127
6.1.3 session 周期的 JavaBean.....	128
6.1.4 application 周期的 JavaBean... ..	129
6.2 MVC 模式与注册登录.....	130
6.2.1 JavaBean 与 Servlet 管理.....	130
6.2.2 配置文件管理.....	131
6.2.3 数据库设计与连接.....	132
6.2.4 注册.....	132
6.2.5 登录与验证.....	136
6.3 MVC 模式与数据库操作.....	141
6.3.1 JavaBean 与 Servlet 管理.....	141
6.3.2 配置文件与数据库连接.....	141
6.3.3 MVC 设计细节.....	142
6.4 MVC 模式与文件操作.....	148
6.4.1 模型(JavaBean).....	148
6.4.2 控制器(Servlet).....	149
6.4.3 视图(JSP 页面).....	150
6.5 案例：计算三角形与梯形的面积.....	151
本章小结.....	154
习题.....	154
第 7 章 表达式语言.....	156
7.1 EL 表达式的语法.....	157
7.1.1 EL 简介.....	157
7.1.2 运算符.....	158
7.1.3 常量与变量.....	164
7.1.4 保留字.....	167
7.2 EL 数据访问.....	168
7.2.1 对象的作用域.....	169
7.2.2 访问 JavaBean.....	171
7.2.3 访问集合.....	172
7.3 其他内置对象.....	173
7.3.1 param 和 paramValues 对象... ..	174
7.3.2 cookie 对象.....	176
7.3.3 initParam 对象.....	177
7.4 案例：EL 表达式的运算应用.....	178

本章小结	179	第 10 章 JSP 的 XML 文件处理	228
习题	180	10.1 认识 XML	229
第 8 章 JSP 与 JDBC	181	10.1.1 XML 概述	229
8.1 认识 JDBC	182	10.1.2 XML 的基本语法	230
8.1.1 JDBC 的定义与产品组件	182	10.1.3 JDK 中的 XML API	233
8.1.2 建立 JDBC 连接	183	10.2 XML 解析模型	234
8.1.3 利用 JDBC 发送 SQL 语句	187	10.2.1 DOM 解析	234
8.1.4 JDBC API 技术记录集接口	189	10.2.2 SAX 解析	236
8.2 JDBC 的包	195	10.2.3 DOM4j 解析	238
8.2.1 RowSet 包	195	10.3 XML 与 Java 类映射 JAXB	240
8.2.2 CachedRowSet 包	198	10.3.1 什么是 XML 与 Java 类 映射	240
8.3 案例: 填充 CachedRowSet 对象 记录集	201	10.3.2 Java 对象转化成 XML	242
本章小结	202	10.3.3 XML 转化为 Java 对象	243
习题	202	10.4 案例: 复杂的映射	245
第 9 章 JSP 中的文件操作	204	本章小结	248
9.1 File 类	205	习题	248
9.1.1 获取文件的属性	205	第 11 章 JSP 与 MySQL 数据库 操作	250
9.1.2 创建目录的基本操作	206	11.1 认识 MySQL 数据库	251
9.1.3 删除文件和目录	208	11.1.1 MySQL 数据库的基础 概念	251
9.2 使用字节流读/写文件	208	11.1.2 安装 MySQL 数据库	251
9.2.1 FileInputStream 类和 FileOutputStream 类	209	11.1.3 配置 MySQL 数据库	255
9.2.2 BufferedInputStream 类和 BufferedOutputStream 类	210	11.1.4 启动 MySQL 数据库	256
9.3 使用字符流读/写文件	212	11.1.5 登录 MySQL 数据库	257
9.3.1 FileReader 类和 FileWriter 类	212	11.2 MySQL 数据库的基本操作	259
9.3.2 BufferedReader 类和 BufferedWriter 类	213	11.2.1 创建数据库	259
9.4 RandomAccessFile 类	215	11.2.2 删除数据库	259
9.5 文件上传和下载	218	11.2.3 创建数据表	260
9.5.1 文件上传	219	11.2.4 修改数据表	266
9.5.2 文件下载	223	11.2.5 删除数据表	274
9.6 案例: 利用 JSP 表单调用文件	225	11.2.6 插入数据	276
本章小结	226	11.2.7 更新数据	282
习题	226	11.2.8 删除数据	283
		11.3 JSP 连接 MySQL	284
		11.3.1 JSP 连接 MySQL 的方法	284
		11.3.2 MySQL 数据库最基本的 DB	



操作	285	12.2 选书	308
11.3.3 调用对 DB 操作的方法	290	12.2.1 选书 JavaBean	308
11.3.4 JSP 数据分页显示	291	12.2.2 选书 JSP	309
11.4 案例：制作旅游景区网站留言本	295	12.3 订单提交及查询	317
本章小结	300	12.3.1 订单提交 Java Bean	317
习题	300	12.3.2 订单提交 JSP	320
第 12 章 网上书店系统设计	302	本章小结	321
12.1 网上书店系统会员登录	303	习题	321
12.1.1 会员登录 JavaBean	303	参考文献	323
12.1.2 会员登录 HTML 与 JSP	306		

第

1

章

JSP 基本概述

本章要点

1. JSP 的组成元素。
2. JSP 的运行原理。

学习目标

1. 了解 JSP 的特点、工作流程。
2. 掌握 JSP 的页面元素。
3. 了解 JSP 的运行环境。
4. 掌握 JDK 的安装与配置。
5. 掌握 Tomcat 的安装与启动。
6. 掌握 Eclipse 的安装与使用。

1.1 了解 JSP 技术

JSP 是一种简化的 Servlet 设计，可以调用强大的 Java 类库，并可以与其他相关的一些技术(Servlet、JavaBean、EJB)联合工作。下面介绍什么是 JSP 以及它有哪些特点。

1.1.1 什么是 JSP

JSP(Java Server Pages)的中文含义是 Java 服务器端语言。其核心技术是 Java 技术，以 Servlet 的形式接受用户的访问和处理数据，在服务器端 JSP 文件会被编译为类文件，其扩展名为.class。JSP 是由 Sun Microsystems 公司倡导、多家公司参与一起建立的一种动态网页技术标准，其在动态网页的创建中有着强大而特殊的功能，是一种实现普通静态 HTML 和动态 HTML 混合编码的技术。在 Sun 正式发布 Java Server Pages 之后，这种新的 Web 应用开发技术很快便引起了人们的关注。Java Server Pages 为创建高度动态的 Web 应用提供了一个独特的开发环境。

Java Server Pages(以下简称为 JSP)是 Java 平台上用于编写包含诸如 HTML、DHTML、XHTML 和 XML 等含有动态生成内容的 Web 页面的应用程序的技术。JSP 技术的功能强大，使用灵活，为创建显示动态 Web 内容的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、Web 应用服务器、浏览器和开发工具共同工作。

许多由 CGI 程序生成的页面大部分仍旧是静态 HTML，动态内容只在页面中有限的几个部分出现。但是包括 Servlet 在内的大多数 CGI 技术及其变种，总是通过程序生成整个页面。例如，下面就是一个简单的 JSP 页面：

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD><TITLE>欢迎访问</TITLE></HEAD>
<BODY>
<H1>欢迎</H1>
<SMALL>欢迎,
<!-- 首次访问的用户名字为"New User" -->
<!--
    out.println 用来输出内容
-->
<% out.println(Utils.getUserNameFromCookie(request)); %>
要设置账号信息, 请点击:
<A HREF="Account-Settings.html">这里</A></SMALL>
<P>
页面的其余内容:
</BODY></HTML>
```

1.1.2 JSP 的特点与工作流程

我们可以将 JSP 看作 Java Servlet 的一种扩展, 在使用时 JSP 必须被编译为 Servlet, 也就是 Java 类, 然后才能调用执行, Servlet 所产生的 Web 页面不能包含在 HTML 标签中。JSP 的应用特点如下。

1. 实现跨平台操作

JSP 技术的最大特点是其编写的代码与设计平台完全无关, 用户可以将任何平台上编写的 JSP 页面拿来在任何 Web 服务器或 Web 应用服务器上运行, 然后通过任何 Web 浏览器访问。除此之外, JSP 还可以在任何平台上建立服务器组件, 在任何服务器上运行程序。从 JSP 的这个特点可以看出, 应用程序开发者只要在自己选用的任意平台上编写 Web 页面, 就可以将编写好的页面放在任意服务器上运行, 当需要对页面进行修改时, 也无须考虑开发运行平台。JSP 页面的内置脚本语言是基于 Java 编程语言的, 因此, JSP 页面都要被编译为 Servlet。

2. 可重复使用组件

JSP 页面依赖于可重用的、跨平台的组件来执行应用程序中所要求的更为复杂的处理。基于组件的方法的特点是: 能够提高总体开发过程的效率, 使得各种组织在他们现有优点的基础上得到更好的优化处理。开发人员能够共享并且交换执行普通操作的组件, 这些组件除了可以将网页的设计与逻辑程序的设计分离以节约开发时间, 还可以充分利用 Java 以及其他脚本语言的跨平台的能力及其灵活性。

3. 标记简化的语言

标准的 JSP 标记可以访问和实例化 JavaBean 组件, 并且可以设置或检索组件属性, 以及下载 Applet, 执行用其他方法更难以编码和耗时的功能。JSP 技术可以将许多功能封装起来, 在进行 Web 页面开发时, 利用这些封装的功能就可以方便地使用与 JSP 相关的 XML 标记进行动态内容的生成。

4. 实现应用程序与页面显示的分离化

应用程序与页面显示的分离化可以使 Web 页面的设计者和管理人员能够互不影响地编辑和使用 JSP 页面, 而不影响其内容的生成。Web 页面的开发人员可以利用 HTML 或 XML 标记来设计和格式化最终页面, 而利用 JSP 标记或 Scriptlet 来生成页面上的动态内容。生成的内容被封装在标记和 JavaBean 组件中, 并将它们捆绑在 Scriptlet 中, 使得所有的脚本程序都运行在服务器端。

通常, 在服务器端由 JSP 引擎解释 JSP 标记和 Scriptlet, 生成所请求的内容, 同时, 将结果以 HTML 或者 XML 页面的形式发送回浏览器。这样做, 不但可以对程序代码进行保密, 又可以保证任何基于 HTML 的 Web 浏览器的跨平台使用。与 Servlet 相比, JSP 可以提供所有 Servlet 的功能, 比使用 println 编写和修改 HTML 更方便。

提示

JSP 还可以更明确地进行分工, Web 页面的设计人员编写 HTML 时, 只需要留出地方让 Servlet 程序员插入动态部分就可以了。

在编写 JSP 程序时，要了解它的执行顺序，JSP 的执行流程如图 1-1 所示。首先，客户端向 Web 服务器提出请求，然后 JSP 引擎负责将页面转化为 Servlet，此 Servlet 经过虚拟机编译生成类文件，然后再把类文件加载到内存中执行。最后，由服务器将处理结果返回给客户端。



图 1-1 JSP 的执行流程

提示

JSP 页面代码会被编译成 Servlet 代码，执行效率没有 Servlet 快，但并不是每一次都需要编译 JSP 页面。当 JSP 第一次被编译成类文件后，重复调用该 JSP 页面时，若 JSP 引擎发现该 JSP 页面没有被改动过，那么会直接使用编译后的类文件而不会再次编译成新的 Servlet。当然，如果页面被修改后，则需要重新加载和编译。

1.1.3 JSP 与类似语言技术的比较

现在，最常用的动态网页设计语言有 ASP(Active Server Pages)、JSP(Java Server Pages) 和 PHP (Hypertext Preprocessor)。

1. ASP

ASP 采用脚本语言 VBScript(JavaScript)作为自己的开发语言。ASP 是 Microsoft 开发的动态网页语言，也继承了微软产品的一贯传统，只能在微软的服务器产品 IIS(Internet Information Server)上执行。ASP 是 Web 服务器端的开发环境，可以产生和执行动态的、交互的、高效的 Web 服务应用程序。

其技术特点主要有以下几个方面：

- (1) 与浏览器无关(Browser Independence)，客户端只要使用可执行 HTML 码的浏览器，即可浏览 Active Server Pages 所设计的网页内容。Active Server Pages 所使用的脚本语言 (VBScript、JScript)均在 Web 服务器端执行，客户端的浏览器不需要执行这些脚本语言。
- (2) Active Server Pages 能与任何 ActiveX Scripting 语言兼容。除了可使用 VBScript 或 JScript 语言设计外，还可以通过 plug-in 方式，使用由第三方提供的其他脚本语言，如 REXX、Perl、Tcl 等。脚本引擎是处理脚本程序的 COM(Component Object Model)对象。
- (3) 使用 VBScript、JScript 等简单易懂的脚本语言，结合 HTML 代码，即可快速地编写出网站的应用程序。可使用服务器端的脚本来产生客户端的脚本。
- (4) 使用普通的文本编辑器，如 Windows 的记事本，即可进行程序设计，无须编译，容易编写，可在服务器端直接执行。

2. PHP

PHP 是跨平台的服务器端的嵌入式脚本语言。它几乎都要借用 C、Java 和 Perl 语言的语法，同时结合 PHP 自己的特性，使得 Web 开发者能够快速写出动态页面。PHP 的特点

是：支持绝大多数数据库，并且其源码是完全公开的。PHP 可在 Windows、Unix、Linux 的 Web 服务器上正常执行，还支持 IIS、Apache 等一般的 Web 服务器，用户更换平台时，无需变换 PHP 代码。PHP 与 MySQL 是目前绝佳的组合。用户还可以自己编写外围的函数间接存取数据库，通过这样的途径，在更换使用的数据库时，可以轻松地修改编码以适应这样的变化。

提示

PHP LIB 就是最常用的可以提供一般事务需要的一系列基库。但 PHP 提供的数据库接口支持彼此不够统一。

3. JSP

JSP 同 PHP 类似，几乎可以在所有平台上执行，如 Windows、Linux、UNIX。Web 服务器 Apache 已经能够支持 JSP，而 Apache 广泛应用在 Windows、UNIX 和 Linux 上，因此 JSP 有更广泛的执行平台。虽然现在 Windows 操作系统占了很大的市场份额，但是在服务器方面 UNIX 的优势仍然很大，而新崛起的 Linux 更是来势不小。从一个平台移植到另外一个平台时，JSP 和 JavaBean 甚至不用重新编译，因为 Java 字节码都是标准的，与平台无关。

ASP、PHP、JSP 三者都是面向 Web 服务器的技术，客户端浏览器不需要任何附加的软件支持。普通的 HTML 页面只依赖于 Web 服务器，但 ASP、PHP、JSP 页面需要附加的语言引擎分析和执行程序代码。程序代码的执行结果被重新嵌入 HTML 代码中，然后一起发送给浏览器。三者都提供了在 HTML 代码中混合某种程序代码、由语言引擎解释执行程序代码的能力。JSP 代码被编译成 Servlet 并由 Java 虚拟机解释执行，这种编译操作仅在对 JSP 页面的第一次请求时发生。在 ASP、PHP、JSP 环境下，HTML 代码主要负责描述信息的显示样式，而程序代码则用来描述处理逻辑。

1.1.4 JSP 页面的组成

在 HTML 页面文件中加入 Java 程序段和 JSP 标签，即可构成一个 JSP 页文件，JSP 页面由 5 种元素组合而成。

- (1) 普通的 HTML 标记符。
- (2) JSP 标签，如指令标签、动作标签。
- (3) 变量和方法的声明。
- (4) Java 程序段。
- (5) Java 表达式。

当服务器上的 JSP 页面被第一次请求执行时，服务器上的 JSP 引擎首先将 JSP 页面文件转译成 Java 文件，再将 Java 文件编译，生成字节码文件，然后通过执行字节码文件响应客户的请求，这个字节码文件的任务如下。

- (1) 把 JSP 页面中普通的 HTML 标记符号交给客户的浏览器执行并显示。
- (2) JSP 标签、数据和方法声明、Java 程序段由服务器负责执行，将需要显示的结果发送给客户的浏览器。
- (3) Java 表达式由服务器负责计算，并将结果转化为字符串，然后交给客户的浏览器负责显示。

1.1.5 JSP 页面中的元素

Java Server Pages(JSP)能够分离页面的静态 HTML 和动态部分。HTML 可以用任何通常使用的 Web 制作工具编写,编写方式也和原来的一样;动态部分的代码放入特殊标记之内,大部分以“<%”开始,以“%>”结束。例如,下面是一个 JSP 页面的片断,如果用 `http://host/test.jsp?title=Core+Web+Programming` 这个 URL 打开该页面,则结果显示“Thanks for ordering Core Web Programming”。

test.jsp 源程序如下:

```
Thanks for ordering
<I><%= request.getParameter("title") %></I>
```

JSP 页面文件通常以 .jsp 为扩展名,而且可以安装到任何能够存放普通 Web 页面的地方。虽然从代码编写来看, JSP 页面更像普通 Web 页面而不像 Servlet,但实际上, JSP 最终会被转换成正规的 Servlet,静态 HTML 直接输出到和 Servlet service 方法关联的输出流。

JSP 到 Servlet 的转换过程一般在出现第一次页面请求时进行。因此,如果希望第一个用户不会由于 JSP 页面转换成 Servlet 而等待太长的时间,并且希望确保 Servlet 已经正确地编译并装载,你可以在安装 JSP 页面之后自己请求这个页面,这样 JSP 页面就转换成 Servlet 了。

另外也请注意,许多 Web 服务器允许定义别名,所以一个看起来指向 HTML 文件的 URL 实际上可能指向 Servlet 或 JSP 页面。

除了普通 HTML 代码之外,嵌入 JSP 页面的其他成分主要有三种:脚本元素(Scripting Element)、指令(Directive)和动作(Action)。脚本元素用来嵌入 Java 代码,这些 Java 代码将成为转换得到的 Servlet 的一部分;JSP 指令用来从整体上控制 Servlet 的结构;动作用来引入现有的组件或者控制 JSP 引擎的行为。为了简化脚本元素, JSP 定义了一组可以直接使用的变量(预定义变量)。

1.2 JSP 的安装与配置

使用 JSP 开发程序,需要具备对应的运行环境:Web 浏览器、Web 服务器、JDK 开发工具包、数据库(MySQL、SQL Server 等)。下面以 Windows 操作系统为平台介绍 JSP 的安装与配置。

1.2.1 JDK 的安装与配置

JDK 包含运行 Java 程序必需的 Java 运行环境(Java Runtime Environment, JRE)及开发过程中常用的库文件。在使用 JSP 开发网站前,要先安装 JDK 组件。

1. JDK 的安装

进入 Oracle 公司网站下载 JDK 1.8.0 安装程序包,其网址为 `http://www.oracle.com/technetwork/java/javase/downloads/jdk-8u111-windows-i586.exe`,下载后的操作步骤如下。

(1) 双击下载的 `jdk-8u111-windows-i586.exe` 文件,系统将自动启动 Windows Installer,

开始安装过程，随后，系统弹出许可证协议对话框，单击“接受”按钮，弹出如图 1-2 所示的安装向导。

(2)单击“下一步”按钮，选择默认配置，如图 1-3 所示。



图 1-2 安装向导



图 1-3 定制安装

(3) 系统开始复制新文件，如图 1-4 所示。

(4) 进入目标文件夹界面，选择默认安装路径，单击“下一步”按钮，如图 1-5 所示。



图 1-4 复制文件



图 1-5 设置安装路径

(5) 进入安装状态，滚动条会显示安装的进度，如图 1-6 所示。

(6) 最后单击“关闭”按钮，完成 JDK 的安装，如图 1-7 所示。



图 1-6 安装 JDK 状态



图 1-7 完成 JDK 安装



2. 配置

安装好 JDK 软件之后, 要对 Windows 操作系统的环境变量进行设置, 设置环境变量的步骤如下。

(1) 右键单击“计算机”, 在弹出的快捷菜单中选择“属性”菜单命令, 系统将自动打开“系统属性”对话框, 在该对话框中单击“高级”选项卡中的“环境变量”按钮, 如图 1-8 所示。

(2) 打开“新建用户变量”对话框, 在“变量名”文本框中输入 JAVA_HOME, 并配置变量值, 单击“确定”按钮, 如图 1-9 所示。再编辑变量 PATH, 变量值为%JAVA_HOME%\bin, 单击“确定”按钮, 如图 1-10 所示。

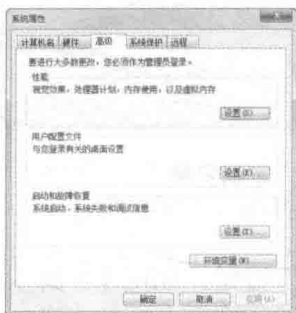


图 1-8 “高级”选项卡



图 1-9 新建系统变量 JAVA_HOME



图 1-10 编辑系统变量 PATH

(3) 返回到“环境变量”对话框, 可以看到配置的变量, 单击“确定”按钮, 如图 1-11 所示。

(4) 设置完成系统变量后, 在 DOS 模式或命令行模式下, 输入命令“javac”, 出现如图 1-12 所示窗口, 表示 JDK 软件安装成功。



图 1-11 “环境变量”对话框

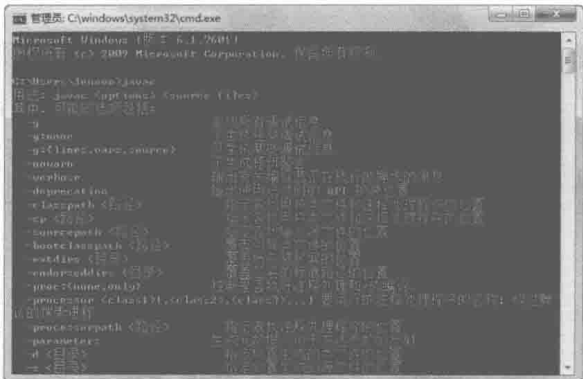


图 1-12 JDK 安装成功

1.2.2 Tomcat 的安装与启动

Tomcat 是由 JavaSoft 和 Apache 开发团队共同合作推出的产品, 它完全支持 Servlet 和 JSP, 并且可以免费使用。

由于 JSP 程序是需要在服务器中运行的,因此还需要运行网页的 Apache 服务器。下面来安装 Apache 服务器 Tomcat。

- (1) 进入 Tomcat 官方网站 <http://tomcat.apache.org>, 下载 Tomcat 7.0, 如图 1-13 所示。
- (2) 下载后解压文件, 如图 1-14 所示。

Binary Distributions

- Core:
 - zip (pgp, md5, sha1)
 - tar.gz (pgp, md5, sha1)
 - 32-bit Windows zip (pgp, md5, sha1)
 - 64-bit Windows zip (pgp, md5, sha1)
 - 32-bit/64-bit Windows Service Installer (pgp, md5, sha1)

图 1-13 Tomcat 7.0 下载页面

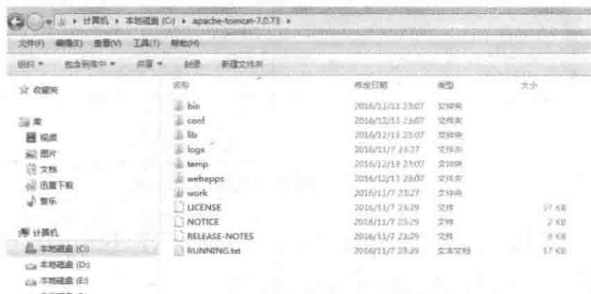


图 1-14 解压文件

(3) 进入 Tomcat 安装目录下的 bin 子目录, 可以看到 startup.bat 和 shutdown.bat 文件。双击 startup.bat 启动 Tomcat 服务器, 将产生如图 1-15 所示的输出信息。

(4) 在浏览器中输入 <http://localhost:8080>, 出现如图 1-16 所示情况则表示 Tomcat 安装成功。

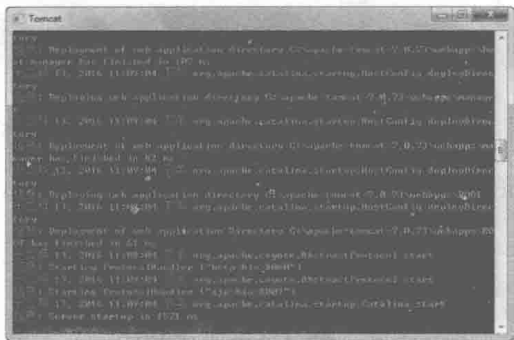


图 1-15 启动 Tomcat 服务

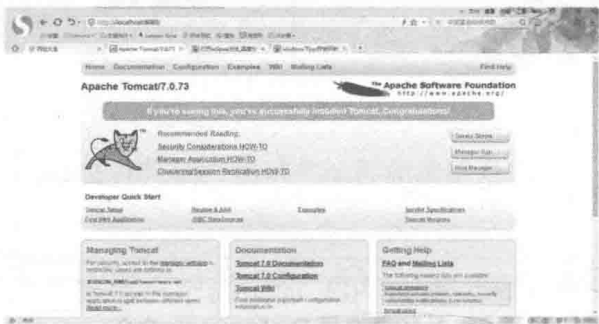


图 1-16 测试 Tomcat 服务

提示

- ① 直到看到提示信息“Server startup in 1571 ms”输出, 表示 Tomcat 启动完毕。否则可能出现错误, 将无法启动。这时, 需要关闭 Tomcat 服务器, 可以关闭这个 CMD 窗口, 也可以双击运行 shutdown.bat。
- ② Tomcat 服务器默认占用 8080 端口, 如果 Tomcat 要使用的端口已经被占用, 则 Tomcat 服务器将无法启动。

1.2.3 Eclipse 的安装与使用

Eclipse 提供了大量的 Java 工具集, 如 CCS/JS/HTML/XML 编辑器, 可以帮助创建 EJB 和 Struts 项目的向导、编辑 Hibernate 配置文件和执行 MySQL 语句的工具等。

(1) 进入 Eclipse 官方网站 <https://www.eclipse.org/downloads/>, 下载并解压 Eclipse 文件, 双击 eclipse.exe 应用程序, 如图 1-17 所示。

(2) 进入 Workspace Launcher 对话框, 设置路径, 单击 OK 按钮, 如图 1-18 所示。