

徐小平 夏保芹 迟增晓 编著

JSP程序设计 实训与案例教程

计算机科学与技术专业实践系列教材

清华大学出版社



014057137

计算机科学与技术专业实践系列教材

TP312JA-43
316

JSP程序设计 实训与案例教程

徐小平 夏保芹 迟增晓 编著



TP312JA-43
316

清华大学出版社



北航

C1742173

内 容 简 介

本书旨在培养和提高学生 Java Web 技术的工程开发和创新实践能力。

本书理论联系实际,引进以项目案例为驱动的教学模式,系统地介绍 JSP 应用开发中的各种相关技术。全书共分 10 章,内容包括 JSP 概述、Web 客户端编程、JSP 基础知识、JSP 内置对象、JSP 数据库应用开发、网上选课系统项目实训、JavaBean 技术、Servlet 技术、自定义标签和 AJAX 技术等。通过本书的学习,可以为后续课程(如 Java Web 高级开发技术、SSH 框架技术等)打下良好基础。

本书重视对学生工程实践能力的培养和训练,“项目”和“案例”理念贯穿始终。部分章节安排了综合案例,有助于学生理解知识、应用知识,达到学以致用。

本书可作为普通高等院校计算机、信息管理等专业“JSP 程序设计”、“Java Web 开发”、“动态网站制作”等相关课程的教材,也可以作为 Java Web 编程爱好者和开发人员的参考用书。

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 程序设计实训与案例教程/徐小平,夏保芹,迟增晓编著. --北京:清华大学出版社,2014

计算机科学与技术专业实践系列教材

ISBN 978-7-302-36883-0

I. ①J… II. ①徐… ②夏… ③迟… III. ①JAVA 语言—网页制作工具—高等学校—教材
IV. ①TP312 ②TP393.092

中国版本图书馆 CIP 数据核字(2014)第 131399 号

责任编辑:白立军

封面设计:傅瑞学

责任校对:白 蕾

责任印制:何 芊

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>, 010-62795954

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:23.25

字 数:539 千字

版 次:2014 年 8 月第 1 版

印 次:2014 年 8 月第 1 次印刷

印 数:1~2000

定 价:39.50 元

产品编号:057166-01

前 言

JSP 是由 Sun 公司在 1999 年 6 月推出的一种动态网页技术,在目前流行的 Web 开发技术中占有很大的比重。它基于 Java 语言,具备 Java 技术简单易用、完全面向对象、与平台无关、安全可靠等特点,可以在 Servlet 和 JavaBean 的支持下,完成功能强大的动态网站设计,成为大、中型网站开发的首选。

本书围绕 JSP 开发循序渐进地介绍各种相关技术,旨在培养和提高学生 Java Web 技术的工程开发和创新实践能力。

本书内容丰富,理论联系实际,引进以项目案例为驱动的教学模式,系统介绍 JSP 应用开发中的各种相关技术。全书共分 10 章,各章主要内容如下。

第 1 章 JSP 概述,主要介绍 Web 的基础知识、JSP 的开发工具及运行环境、JSP 的工作原理和 JSP 项目的目录结构等。

第 2 章 Web 客户端编程,主要介绍 HTML 标记、CSS 样式表、JavaScript 脚本,最后通过一个案例介绍静态网页的设计方法。

第 3 章 JSP 基础知识,主要介绍 JSP 页面元素的组成、JSP 脚本的使用、常用的指令元素和动作元素,最后对 JSP 开发中用到的 Java 知识做简要回顾。

第 4 章 JSP 内置对象,主要介绍 request、response、session 和 application 等常用 JSP 内置对象的用法,并通过一个网上模拟考试系统案例介绍在 Java Web 开发中各主要内置对象的综合应用。

第 5 章 JSP 数据库应用开发,主要介绍如何通过 JDBC 访问 SQL Server 2005 数据库,如何对数据库进行增、删、改、查等基本操作,并给出一个数据库综合访问案例。同时还介绍在数据库开发中的其他重要技术,如存储过程、事务处理、连接池、分页显示技术等。

第 6 章网上选课系统项目实训,以网上选课系统为例,介绍基于 JSP 进行 Web 应用系统开发的全过程。

第 7 章 JavaBean 技术,主要介绍 JavaBean 的基本概念、JavaBean 的创建和简单应用。通过一个数据库综合访问案例介绍基于 JSP+JavaBean 的 Web 开发方法。另外,介绍第三方组件 jspSmartUpload 实现文件上传和下载。

第 8 章 Servlet 技术,主要介绍 Servlet 的基本概念、Servlet 的常用类及接口、Servlet 的编写和访问、Servlet 的简单应用、Servlet 的过滤器和监听器。通过学生选课系统案例介绍基于 JSP+Servlet+JavaBean 的 Web 应用开发过程。

第 9 章自定义标签,主要介绍 JSP 标签的基本概念、如何开发使用自定义标签,并以实例说明如何使用自定义标签及 JSTL 常用标签。

第 10 章 AJAX 技术,主要介绍什么是 AJAX、核心对象 XMLHttpRequest 的使用、如何进行 AJAX 简单程序的开发,最后通过一个实例讲述 AJAX 程序的开发流程。

通过本书的学习,可以为后续课程(如 Java Web 高级开发技术、SSH 框架技术等)打下

良好基础。本书重视对学生工程实践能力的培养和训练,“项目”和“案例”理念贯穿始终。部分章节安排了综合案例,有助于学生理解知识、应用知识,达到学以致用。

本书主要特点如下。

(1) 知识体系结构合理。本书各章安排强调整体性和系统性,知识表达强调层次性和有序性,便于读者学习和理解。

(2) 注重理论与实践相结合。本书所有知识都结合具体实例进行介绍,力求突出动手能力,使读者在实践基础上加深对理论的认识及掌握。

(3) 突出案例的典型性和针对性。以案例为线索组织知识,围绕知识设计案例,将系统性的案例融入全书的知识介绍中。

(4) 综合案例实用性强。详细介绍从需求分析、数据库设计到系统设计、编码实现的整个流程。读者通过案例不仅能掌握书中知识点的综合应用,更能学会 Web 系统开发的一般流程。

(5) 注重启发式、引导式教学。案例项目介绍完后,都设计了若干思考题、课后习题和改进建议,引导学生在掌握现有知识的基础上继续探究,不断提高。

本书理论和实践相结合,内容翔实,实例丰富,图文并茂,体系完整,可作为普通高等院校计算机、信息管理等专业“JSP 程序设计”、“Java Web 开发”、“动态网站制作”等相关课程的教材,也可以作为 Java Web 编程爱好者和开发人员的参考用书。

本书在编写过程中,得到很多同行老师的帮助,除本书最后列出的参考文献外,还参考了一些博客、论坛等,在此一并表示感谢。

本书由徐小平、夏保芹、迟增晓共同完成。其中第 1 章、第 4~8 章由徐小平编写,第 2~3 章由夏保芹编写,第 9~10 章由迟增晓编写。另外,夏保芹还参与了第 4~6 章部分代码的测试。全书由徐小平统稿。

为便于教学,本书配有多媒体课件、课后习题答案、所有调试通过的源代码,读者可以从清华大学出版社网站(www.tup.com.cn)下载。

感谢读者选择使用本书。由于作者水平有限,书中难免存在错误、疏漏之处,欢迎读者对本书结构、内容提出批评和修改建议,我们将不胜感激。

作者

2014 年 5 月

目 录

第 1 章 JSP 概述	1
1.1 Web 简介	1
1.1.1 什么是 Web	1
1.1.2 静态网页与动态网页	2
1.1.3 Web 动态网页技术	2
1.1.4 Web 服务器介绍	4
1.1.5 Java Web 常用开发技术	5
1.2 JSP 开发工具及运行环境	7
1.2.1 JDK 的安装与配置	7
1.2.2 Tomcat 的安装与启动	9
1.2.3 Dreamweaver 开发工具	14
1.2.4 MyEclipse 开发工具	14
1.3 第一个 JSP 程序	20
1.3.1 认识第一个 JSP 程序	20
1.3.2 JSP 的工作原理	21
1.3.3 使用 MyEclipse 进行 Web 开发	23
1.3.4 Web 应用的目录结构	25
1.4 本章小结	26
习题	26
第 2 章 Web 客户端编程	27
2.1 HTML 语言	27
2.1.1 HTML 文档结构	27
2.1.2 文字与段落标记	29
2.1.3 多媒体标记和超链接标记	32
2.1.4 列表标记	34
2.1.5 表格标记	36
2.1.6 表单标记	40
2.1.7 框架标记	44
2.2 CSS	47
2.2.1 CSS 组成	47
2.2.2 CSS 样式的 4 种应用方式	49
2.2.3 CSS 样式的继承性及优先级	50
2.2.4 CSS 常用的样式属性	51

2.2.5	DIV+CSS 布局方法	52
2.3	JavaScript 脚本语言	54
2.4	静态网页设计综合案例	57
2.4.1	案例介绍	57
2.4.2	案例实现	59
2.5	本章小结	61
	习题	62
第3章	JSP 基础知识	63
3.1	JSP 页面组成	63
3.2	JSP 中常用的脚本元素	66
3.2.1	JSP 中变量和方法的声明	66
3.2.2	JSP 中的表达式	66
3.2.3	JSP 中的 Java 脚本	67
3.3	JSP 常用指令	68
3.3.1	page 指令	69
3.3.2	include 指令	71
3.3.3	taglib 指令	73
3.4	JSP 常用动作	73
3.4.1	<jsp:include> 动作	73
3.4.2	<jsp:forward> 动作	75
3.4.3	<jsp:useBean> 动作	78
3.4.4	<jsp:setProperty> 动作	79
3.4.5	<jsp:getProperty> 动作	80
3.5	Java 知识回顾	80
3.5.1	Java 编程基础	80
3.5.2	Java 程序结构	81
3.5.3	类与接口、包的概念	83
3.5.4	Java 常用类库	84
3.5.5	Java 异常处理	87
3.6	本章小结	87
	习题	88
第4章	JSP 内置对象	89
4.1	JSP 内置对象概述	89
4.2	request 对象	90
4.2.1	request 对象的常用方法	90
4.2.2	request 对象的主要应用	91
4.2.3	request 对象其他方法应用	98

4.3	response 对象	99
4.3.1	response 对象的常用方法	99
4.3.2	response 对象的主要应用	99
4.4	session 对象	105
4.4.1	session 对象的相关知识	105
4.4.2	session 对象的主要应用	106
4.5	application 对象	111
4.5.1	application 对象的常用方法	112
4.5.2	application 对象的主要应用	112
4.6	JSP 其他内置对象	113
4.7	JSP 内置对象综合应用	113
4.7.1	网上模拟考试系统案例介绍	114
4.7.2	网上模拟考试系统案例实现	114
4.7.3	网上模拟考试系统案例总结	124
4.8	本章小结	125
	习题	125
第 5 章	JSP 数据库应用开发	127
5.1	关系数据库简介	127
5.1.1	关系数据库基本概念	127
5.1.2	常见关系数据库管理系统	128
5.1.3	SQL 语言简介	128
5.1.4	本章实例数据库介绍	130
5.2	JDBC 技术介绍	132
5.2.1	JDBC 概述	132
5.2.2	JDBC 常用类和接口介绍	133
5.2.3	JDBC 数据库访问步骤	134
5.3	连接 SQL Server 数据库	135
5.3.1	DriverManager 类	135
5.3.2	使用 JDBC-ODBC 桥连接数据库	136
5.3.3	使用专用 JDBC 驱动连接数据库	141
5.4	访问 SQL Server 数据库	143
5.4.1	Connection 接口	143
5.4.2	Statement 接口及应用	144
5.4.3	PreparedStatement 接口及应用	150
5.4.4	ResultSet 接口及应用	152
5.4.5	数据库访问综合案例	157
5.5	其他重要技术及典型应用	170
5.5.1	存储过程	170

5.5.2	事务处理	174
5.5.3	连接池技术	175
5.5.4	数据分页显示技术	177
5.6	常见数据库连接	181
5.6.1	连接 Oracle 数据库	181
5.6.2	连接 Access 数据库	183
5.6.3	连接 MySQL 数据库	185
5.6.4	连接 Excel 表格	186
5.7	本章小结	188
	习题	188
第 6 章	网上选课系统项目实训	190
6.1	系统需求分析	190
6.2	系统功能模块设计	190
6.3	系统数据库设计	192
6.3.1	数据库概念结构设计	192
6.3.2	数据库逻辑结构设计	192
6.4	系统实现	194
6.4.1	系统文件目录介绍	194
6.4.2	系统登录页面的实现	195
6.4.3	管理员模块的实现	198
6.4.4	学生用户模块的实现	213
6.4.5	教师用户模块的实现	223
6.5	系统完善和改进	228
6.6	本章小结	229
	习题	229
第 7 章	JavaBean 技术	230
7.1	JavaBean 简介	230
7.1.1	JavaBean 基本概念	230
7.1.2	JavaBean 规范	231
7.1.3	JavaBean 的种类	231
7.2	JavaBean 的创建和调用	232
7.2.1	JavaBean 的创建	232
7.2.2	JavaBean 的调用	237
7.3	JSP+JavaBean 综合开发案例	245
7.3.1	案例介绍	245
7.3.2	案例实现	245
7.4	文件上传与下载组件	251

102	7.4.1	概述	251
102	7.4.2	jspSmartUpload 组件介绍	252
102	7.4.3	文件上传	253
102	7.4.4	文件下载	255
102	7.5	Java Web 开发模式介绍	257
102	7.6	本章小结	259
102		习题	259
102		第 8 章 Servlet 技术	260
102	8.1	Servlet 概述	260
102	8.1.1	什么是 Servlet	260
102	8.1.2	Servlet 的生命周期	261
102	8.1.3	认识第一个 Servlet	262
102	8.2	Servlet 编程常用类和接口	263
102	8.2.1	Servlet 常用类和接口简介	263
102	8.2.2	Servlet 接口	264
102	8.2.3	GenericServlet 抽象类	264
102	8.2.4	HttpServlet 类	265
102	8.2.5	ServletRequest 接口与 ServletResponse 接口	266
102	8.2.6	HttpServletRequest 接口	266
102	8.2.7	HttpServletResponse 接口	267
102	8.2.8	Servlet 其他常用接口	267
102	8.3	Servlet 的创建和配置	269
102	8.3.1	Servlet 创建方式综述	269
102	8.3.2	创建一个 Servlet	271
102	8.3.3	Servlet 的配置和调用	271
102	8.4	Servlet 简单应用	275
102	8.5	Servlet 过滤器	283
102	8.5.1	Servlet 过滤器简介	283
102	8.5.2	Servlet 过滤器接口	284
102	8.5.3	Servlet 过滤器的创建和应用	284
102	8.6	Servlet 监听器	288
102	8.6.1	Servlet 监听器介绍	288
102	8.6.2	Servlet 监听器应用	289
102	8.7	JSP+Servlet 综合应用案例	290
102	8.7.1	案例介绍	290
102	8.7.2	案例实现	290
102	8.8	本章小结	299
102		习题	299

第 9 章 自定义标签	301
9.1 自定义标签简介	301
9.1.1 自定义标签的基本概念.....	301
9.1.2 自定义标签的特点.....	301
9.1.3 自定义标签的接口与类的层次结构.....	302
9.2 自定义标签的开发	305
9.2.1 自定义标签的开发流程.....	305
9.2.2 自定义标签的处理过程.....	308
9.2.3 HelloWorld 标签的开发	309
9.3 JSTL 标准标签库	313
9.3.1 JSTL 简介	313
9.3.2 JSTL 的下载及安装	314
9.3.3 JSTL 核心标签库	314
9.4 自定义标签的使用实例	322
9.4.1 自定义标签显示时间.....	322
9.4.2 使用 JSTL 标签库实现图书查询功能	325
9.5 本章小结	334
习题.....	334
第 10 章 AJAX 技术	335
10.1 什么是 AJAX	335
10.1.1 AJAX 定义.....	335
10.1.2 为什么要用 AJAX	335
10.1.3 当前 AJAX 的典型应用	337
10.1.4 AJAX 的缺陷.....	337
10.1.5 AJAX 简单实例	337
10.2 AJAX 开发基础.....	339
10.2.1 XMLHttpRequest 核心对象	339
10.2.2 XMLHttpRequest 对象的五步使用法	343
10.2.3 AJAX 读取 XML	346
10.2.4 AJAX 与数据库.....	348
10.3 AJAX 应用案例.....	351
10.3.1 案例介绍.....	351
10.3.2 案例实现.....	351
10.4 AJAX 框架简介.....	358
10.5 本章小结.....	359
习题.....	360
参考文献	361

第 1 章 JSP 概述

本章学习目标

- 了解 Web 开发的基础知识。
- 掌握 JDK 的安装与配置。
- 掌握 Tomcat 的安装与启动。
- 了解 Tomcat 的目录结构。
- 了解 MyEclipse 开发工具。
- 熟悉 JSP 程序的编辑与运行。

本章首先向读者介绍 Web 的基础知识,包括 Web 的概念、静态网页与动态网页、Web 动态网页开发技术、Web 服务器、Java Web 开发技术等。接着介绍 JSP 的开发工具及运行环境,包括 JDK 的安装与配置、Tomcat 服务器的安装与启动、Dreamweaver 开发工具、MyEclipse 集成开发环境的配置等。最后通过一个简单的 JSP 程序的编写和运行,介绍 JSP 的工作原理、JSP 项目的目录结构,以及如何在 MyEclipse 集成环境中开发 Java Web 项目。

1.1 Web 简介

1.1.1 什么是 Web

Web 的全称为 World Wide Web,简称为 WWW,中文译为“万维网”,它是 Internet 上的一种服务。

Web 技术是 Internet 的核心技术之一,它的主要功能是信息发布和信息处理。它可以把网络上处于不同地域的计算机内的各种类型的信息,如文本、图像、声音、动画等有机地组织在一起,并允许 Web 用户通过超文本传输协议(HyperText Transfer Protocol,HTTP)从一台 Web 服务器转到另一台 Web 服务器上检索信息。

现在 Web 应用已渗透到几乎所有信息领域,如教育、科技、新闻、娱乐、广告、电子商务、电子政务和企业 ERP 等,成为 Internet 上最受欢迎的应用之一。Web 的成功主要取决于以下 3 个要素。

(1) 万维网联盟(World Wide Web Consortium,W3C)制定了一套标准的、易于人们掌握的超文本标记语言(Hypertext Markup Language,HTML)。

(2) 使用统一资源定位器(Uniform Resource Locator,URL)来描述 Internet 上资源的位置和访问方式。

(3) 提供一种超文本传输协议(HTTP)来解决如何访问 Internet 上的资源。

人们只要通过 Web 浏览器,就可以轻松地访问 Web 服务器上的资源。这些资源以网

页的形式呈现给用户。网页一般又称为 HTML 文件,是一种可以在 WWW 上传输、能被浏览器翻译并显示出来的文件。网页是构成网站的基本元素,是承载各种网站应用的平台。根据其生成方式,可以将网页分为静态网页和动态网页两种。

1.1.2 静态网页与动态网页

1. 静态网页

在网站设计中,纯粹 HTML 格式的网页通常称为“静态网页”,早期的网站一般都是由静态网页构成的。在静态网页中可以出现各种动态效果,如 GIF 格式的动画、Flash、滚动字幕等,这些“动态效果”只是视觉上的,与下面将要介绍的动态网页是不同的概念。

静态网页的特点简要归纳如下。

(1) 每个静态网页都有一个固定的 URL,网页 URL 以 .htm、.html、.shtml 等常见形式为后缀,而且不含有“?”。

(2) 网页内容一经发布到网站服务器,无论是否有用户访问,每个静态网页的内容都保存在网站服务器上。也就是说,静态网页是实实在在保存在服务器上的文件,每个网页都是一个独立的文件。

(3) 静态网页的内容相对稳定,因此容易被搜索引擎检索。

(4) 静态网页没有数据库的支持,在网站制作和维护方面工作量较大,因此当网站信息量很大时完全依靠静态网页制作比较困难。

(5) 静态网页的交互性较差,在功能方面有较大的限制。

(6) 由于不需通过数据库工作,所以静态网页的访问速度比较快。

2. 动态网页

动态网页是指在接到用户访问要求后动态生成的页面,页面内容会随着访问时间和访问者的不同而发生变化,一般运行在服务器端。动态网页是与静态网页相对应的,网页 URL 通常以 .asp、.jsp、.php、.cgi 等形式为后缀。

动态网页的特点简要归纳如下。

(1) 动态网页以数据库技术为基础,可以大大降低网站维护的工作量。

(2) 采用动态网页技术的网站可以实现更多的功能,如用户注册、用户登录、在线调查、用户管理和订单管理等。

(3) 动态网页实际上并不是独立存在于服务器上的网页文件,只有当用户请求时服务器才返回一个完整的网页。

(4) 由于搜索引擎一般不可能从网站的数据库中访问全部网页,因此采用动态网页的网站在进行搜索引擎推广时需要做一定的技术处理才能适应搜索引擎的要求。

静态网页和动态网页各有特点,网站采用动态网页还是静态网页主要取决于网站的功能需求和网站内容的多少。如果网站功能比较简单,内容更新量不是很大,采用纯静态网页的方式会更简单,反之一般要采用动态网页技术来实现。

1.1.3 Web 动态网页技术

常见的 Web 动态网页技术有以下几种。

1. CGI 技术

公共网关接口(Common Gateway Interface, CGI)是第一种使服务器能根据运行时的具体情况动态生成 HTML 页面的技术。

1993 年, CGI 1.0 的标准草案由美国国家超级计算机应用中心(National Center for Supercomputing Applications, NCSA)提出。1995 年开始制定 CGI 1.1 标准。CGI 技术允许服务器端的应用程序根据客户端请求动态生成返回网页,使客户端和服务端的动态信息交换成为可能,推动了 Web 应用的发展。

虽然 CGI 技术成熟而且功能强大,但由于编程困难、效率低下、修改复杂等缺陷,所以有逐渐被新技术取代的趋势。

2. ASP 技术

动态服务器页面(Active Server Pages, ASP)是由微软公司在 1996 年年底推出的一种允许用户将 HTML 或 XML 标记与 VBScript 或 JavaScript 脚本代码相结合生成动态网页的技术,是一个 Web 服务器端的开发环境,利用它可以创建和运行动态、交互、高性能的 Web 应用程序。ASP 网页文件的后缀是 .asp。

ASP 简单易学、安装使用方便、开发工具强大。其主要特点如下。

(1) 使用 VBScript、JavaScript 等简单易懂的脚本语言,结合 HTML 代码即可快速进行网站开发。

(2) 无须编译,容易编写和测试,可以在服务器端直接执行。

(3) 使用普通的文本编辑器,如 Windows 记事本即可进行编辑设计。

(4) 与浏览器无关,用户只需要使用能解释 HTML 代码的浏览器,即可浏览用 ASP 设计的网页内容。

(5) ASP 提供了一些内置对象,通过这些对象可以使服务器端脚本功能更强大。

(6) ASP 可以使用服务器端 ActiveX 组件来执行各种各样的任务,例如,存取数据库、收发 E-mail 或访问文件系统等,并且组件具有无限可扩充性。

(7) 可以使用服务器端的脚本来产生客户端脚本,防止 ASP 程序代码被窃取。

2001 年,微软公司推出 ASP 的升级版 ASP.NET,保留了 ASP 的最大优点并全力使其扩大化。ASP.NET 作为一种全新的技术架构,几乎完全基于组件和模块化,使编程变得更加简单,成为创建动态网站和 Web 应用的首选技术之一。

但 ASP 只能运行于微软公司的服务器产品之上,所以在安全性、稳定性、跨平台性等方面还有待继续提升。

3. PHP 技术

超文本预处理器(Hypertext Preprocessor, PHP)是一种跨平台的服务器端的嵌入式脚本语言,由创始人 Rasmus Lerdorf 在 1994 年提出,1995 年发布第一个公开版本。PHP 大量地借用 C、Java 和 Perl 语言的语法,并耦合 PHP 自己的特性,使 Web 开发者能够快速开发出 Web 应用程序。PHP 开放源码,是完全免费的,并且可以在多种 Web 服务器平台上运行。PHP 网页文件的后缀是 .php。

PHP 与 HTML 语言具有非常好的兼容性,可以直接在脚本代码中加入 HTML 标记,或者在 HTML 标记中加入脚本代码从而更好地实现页面控制。它支持目前绝大多数数据库,如 Microsoft SQL Server、MySQL、Sybase、Oracle 等。特别地,PHP 与 MySQL 是一种

绝佳组合。另外,PHP 提供了类和对象,支持面向对象编程。

4. JSP 技术

Java 服务器页面(Java Server Pages,JSP)是由 Sun 公司在 1999 年 6 月推出的一种动态网页技术,它完全解决了 ASP 和 PHP 的脚本级执行的缺点。JSP 是基于 Java 语言的,具备 Java 技术简单易用、完全面向对象、与平台无关、安全可靠等特点,可以在 Servlet 和 JavaBean 的支持下,完成功能强大的动态网站程序的开发。JSP 网页文件的后缀是 .jsp。

JSP 的主要特点如下。

(1) 具有很好的跨平台特性,一次编写,到处运行。

(2) 将内容的生成和显示分离。可以使用 HTML 或 XML 标记设计页面,使用 JSP 标记等生成页面上的动态内容。

(3) 强调可重用的组件。绝大多数 JSP 页面依赖于可重用、跨平台的组件,如 JavaBean、企业 JavaBean 等来执行复杂的应用处理。开发人员可以共享和交换组件,提高了 Web 应用的开发效率。

(4) 多样化和功能强大的开发工具支持。

1.1.4 Web 服务器介绍

Web 服务器是运行和发布 Web 应用的大容器,主要功能是提供网上信息浏览服务。只有将开发的 Web 项目部署到 Web 服务器中,网络中的所有用户才能通过浏览器进行访问。

Web 服务器的工作原理并不复杂,一般可分为如下 4 个步骤。

(1) 连接:在 Web 服务器和浏览器之间建立一种连接关系。

(2) 请求:Web 浏览器向已建立连接的服务器发出各种请求。

(3) 应答:Web 服务器将接收到的请求进行处理后,将处理结果返回给 Web 浏览器进行显示。

(4) 关闭连接:在应答结束以后,将 Web 服务器和浏览器之间的连接断开。

下面介绍几种常见的 Web 服务器。

1. IIS

Internet 信息服务(Internet Information Services,IIS)是微软公司的一款 Web 服务器产品,提供对 ASP 的良好支持,只要在 Windows 平台上安装 IIS,就可以运行 ASP 应用程序。

IIS 提供了一个图形界面的管理工具,称为 Internet 服务管理器,用于监视、配置和控制 Internet 服务。IIS 还提供了 Internet 服务器扩展,作为扩展 Web 服务器功能的编程接口。另外,它也提供一个 Internet 数据库连接器,可以实现对数据库的查询和更新。

IIS 作为一种 Web 服务组件,可以用于网页浏览、文件传输、新闻服务和邮件发送等各方面,使得在互联网或局域网上发布信息成为一件很容易的事情。

2. Apache

Apache 是由 Apache 软件基金组织提供的一款开放源码的 Web 服务器产品,是最流行的 Web 服务器端产品之一。由于其跨平台性和安全性,市场占有率一直较高。世界上很多著名的网站都是 Apache 的产物。它的成功之处主要在于源代码开放、有一支开放的开发

队伍、支持跨平台的应用(可以运行在几乎所有的 UNIX、Windows、Linux 系统平台)以及它的可移植性等方面。

Apache 的主要特点是简单、对静态页面处理效率非常高、性能稳定,并且可以作为代理服务器使用。

3. Tomcat

Tomcat 是 Apache 软件基金组织 Jakarta 项目中的一个核心项目,由 Apache、Sun 和其他一些公司及个人共同开发而成。

Tomcat 和 IIS 等 Web 服务器一样,同样具有处理 HTML 页面的能力。另外,它还是一个 Servlet 和 JSP 容器,独立的 Servlet 容器是 Tomcat 的默认模式。不过, Tomcat 处理静态 HTML 页面的能力不如 Apache 服务器。

Tomcat 服务器是一个免费的开放源代码的 Web 应用服务器,属于轻量级应用服务器。由于 Tomcat 技术先进、性能稳定、扩展性好、运行时占用的系统资源少,因而在中小型系统和并发访问用户不是很多的场合下被普遍使用,成为目前比较流行的 Web 应用服务器,是开发和调试 JSP 程序的首选。目前最新版本是 8.0。

4. JBoss

JBoss 是全世界开发者共同努力的成果,是一个基于 J2EE(Java 2 Platform Enterprise Edition, Java 2 平台企业版)的开放源代码的重量级的应用服务器。

2004 年 6 月, JBoss 应用服务器通过 Sun 公司的 J2EE 认证。从此 JBoss 一直紧跟最新的 J2EE 规范,而且在某些技术领域引领 J2EE 规范的开发。2006 年, JBoss 公司被 Redhat 公司收购。 JBoss 应用服务器具有许多优秀的特质,无论是学习还是应用, JBoss 都为用户提供了一个非常优秀的平台。但 JBoss 核心服务不包括支持 Servlet/JSP 的 Web 容器,一般与 Tomcat 等绑定使用。

5. WebSphere

WebSphere 是一种功能完善、开放的 Web 应用程序服务器,是 IBM 电子商务应用框架的核心部分,它基于 Java 应用环境,能够帮助客户在 Web 上创建自己的业务或将自己的业务扩展到 Web 上,为客户提供了一个可靠、灵活、可扩展、跨平台的集成解决方案,用于建立、部署和管理 Internet 和 Intranet Web 应用程序。

6. WebLogic

WebLogic 是美国 BEA 公司出品的一种多功能、基于 Java Enterprise 标准的 Web 应用服务器,确切地说是一个基于 J2EE 架构的中间件,用于开发、集成、部署和管理大型分布式 Web 应用、网络应用和数据库应用。目前 BEA 公司已被 Oracle 公司收购。

由于 WebLogic 应用服务器拥有处理关键 Web 应用问题所需的性能、可扩展性、高可用性、对开放标准的遵从性、多层架构、支持基于组件的开发等特性,为企业构建自己的应用提供了坚实的基础,因而成为基于 Internet 的企业进行开发、部署的最佳选择。

以上介绍了 Web 开发中几款常用的服务器产品,用户可以根据自己学习或开发的需要进行选择。本书选用 Tomcat 作为 JSP 开发和运行的服务器。

1.1.5 Java Web 常用开发技术

进行 Java Web 项目的开发一般需要掌握如下几种技术。

1. HTML

HTML 是 Web 开发的基础,是一种标记语言,用来定义如何在页面上显示信息。

2. CSS

层叠样式表(Cascading Style Sheets,CSS)是一种不需要编译,直接由浏览器解释执行的标记语言,主要用来对 HTML 页面的风格,如布局、字体、颜色、背景和其他效果等进行更加精确的控制。

3. JavaScript

JavaScript 是一种由浏览器解释的基于对象的脚本语言,不需要编译,可以直接嵌入 HTML 文档中,可有效提高网页的交互性能。

4. Servlet 技术

Servlet 是 Java 平台上的 CGI 技术,运行在服务器端,主要用于处理 HTTP 请求,并将处理结果返回给浏览器生成动态 Web 页面。Servlet 具有与平台无关、功能强大、安全、可扩展和灵活等优点。

5. JSP 技术

JSP 是建立在 Servlet 基础上的一种静态与动态网页混合编码的开发技术,具有与平台无关、可搭配重用组件、分离静态与动态内容等特点。

6. JavaBean 技术

JavaBean 是一些可移植、可重用、能组装到应用程序中的 Java 类,是 Java 组件技术的核心。JavaBean 的结构必须满足一定的规范。通过应用 JavaBean,可以很好地将业务逻辑与前台显示代码分离,提高代码的重用性、可读性和系统的可维护性。

7. JDBC 技术

Java 数据库连接(Java Database Connectivity,JDBC)技术是一种通过 Java 语言访问数据库的应用程序接口技术。它制定了统一的数据库连接和访问标准,使数据库开发人员可以通过 Java 应用程序编程接口(Application Programming Interface,API)访问各种数据库。

8. XML

可扩展标记语言(eXtensible Markup Language,XML)实际上是一种定义语言的语言。它允许用户定义无穷的标记来描述文档中的任何数据元素,将文档内容组织成丰富完整的信息体系。XML 具有以下特点:数据格式便于存储、可扩展、高度结构化以及方便的网络传输,这些特点为创建开放、高效、可扩展、个性化的 Web 应用提供一个崭新的起点。

9. AJAX 技术

异步 JavaScript 和 XML(Asynchronous JavaScript+XML,AJAX)是一种用于创建快速动态网页的技术。通过在后台与服务器进行少量数据交换,AJAX 可以使网页实现异步更新。AJAX 的核心是 JavaScript 对象 XmlHttpRequest。该对象在 IE 5 中首次引入,是一种支持异步请求的技术。

除了以上常见的开发技术外,还有企业 JavaBean(Enterprise JavaBean,EJB)、框架技术(如 Struts、Spring)等。功能强大的 Java Web 项目会应用以上所有技术,它们之间联合使用、协作开发,可以大大提高开发效率和系统性能。