



普通高等学校计算机教育
“十二五”规划教材

卓越工程师培养计划推荐教材
——软件开发类

Java Web 开发与实践

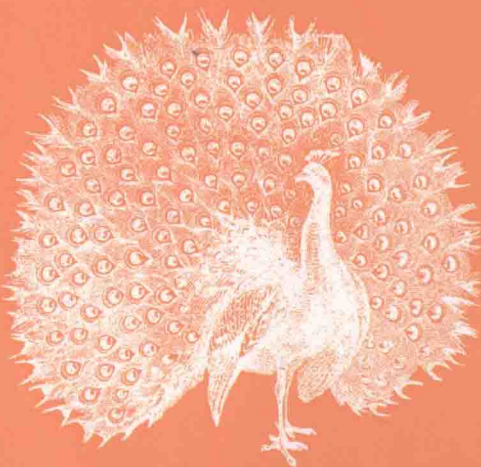
■ 高翔 李志浩 主编 ■ 靳冰 康晓宇 副主编



大容量 DVD

书中所有实例源代码
多媒体视频教学

- 全程 PPT 课件，方便教师授课教学
- 8 小时多媒体视频教学，一线开发人员讲解，轻松学会 Java Web 技术
- 115 个综合实例，14 个实验，在实战中掌握 Java Web 开发
- 企业实际综合案例 + 课程设计，轻松掌握项目开发全过程



 人民邮电出版社
POSTS & TELECOM PRESS



普通高等学校计算机教育
“十二五”规划教材

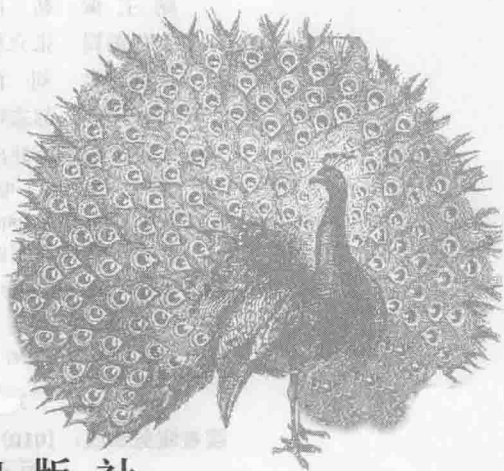
卓越工程师培

计算机应用
软件开发

Java Web 开发与实践



■ 高翔 李志浩 主编 ■ 靳冰 康晓宇 副主编



人民邮电出版社

北京

图书在版编目 (CIP) 数据

Java Web开发与实践 / 高翔, 李志浩主编. — 北京: 人民邮电出版社, 2014.7
普通高等学校计算机教育“十二五”规划教材
ISBN 978-7-115-35803-5

I. ①J… II. ①高… ②李… III. ①JAVA语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2014)第117510号

内 容 提 要

本书作为 Java Web 课程的教材, 系统、全面地介绍了有关 Java Web 程序开发所涉及各类知识。全书共分 18 章, 内容包括 Web 应用开发概述、HTML 与 CSS 网页开发基础、JavaScript 脚本语言、搭建 Java Web 开发环境、JSP 基本语法、JSP 的内置对象、JavaBean 技术、Servlet 技术、数据库应用开发、EL (表达式语言)、JSTL 核心标签库、Ajax 技术、Struts 2 框架技术、Hibernate 技术、Spring 技术、综合案例——基于 Struts 2 +Hibernate+Spring 的网络商城、课程设计——基于 Struts 2 的博客网站、课程设计——基于 Servlet 的图书馆管理系统。书中的每章内容都与实例紧密结合, 有助于学生理解知识、应用知识, 达到学以致用目的。

本书附有配套 DVD 光盘, 光盘中提供本书所有实例、综合实例、实验、综合案例和课程设计的源代码, 及教学录像。其中, 源代码全部经过精心测试, 能够在 Windows XP、Windows Server 2003、Windows 7 系下编译和运行。PPT 课件可从人民邮电出版社教学服务与资源网 (www.ptpdu.com.cn) 上免费下载。

本书可作为应用型本科计算机专业、软件学院、高职软件及相关专业的教材, 同时也适合 Java Web 爱好者以及初、中级的 Java Web 程序开发人员参考使用。

-
- ◆ 主 编 高 翔 李志浩
 - 副 主 编 靳 冰 康晓宇
 - 责任编辑 张立科
 - 执行编辑 刘 博
 - 责任印制 彭志环 杨林杰

 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京中新伟业印刷有限公司印刷

 - ◆ 开本: 787×1092 1/16
印张: 28 2014 年 7 月第 1 版
字数: 736 千字 2014 年 7 月北京第 1 次印刷
-

定价: 59.80 元 (附光盘)

读者服务热线: (010) 81055256 印装质量热线: (010) 81055316

反盗版热线: (010) 81055315

前言

Java 是当前程序开发中最流行的编程语言之一,应用它可以开发桌面应用、网站程序、手机和电子设备程序等,特别在 Web 程序开发方面,Java 跨平台的优越性体现得更加淋漓尽致。近年来,Java Web 框架技术的层出不穷,更体现了 Java 在 Web 方面强大的生命力,从其诞生以来就受到了广大程序开发人员的追捧。目前,无论是高校的计算机专业还是 IT 培训学校,都将 Java Web 作为教学内容之一,这对培养学生的计算机应用能力具有非常重要的意义。

在当前的教育体系下,实例教学是计算机语言教学最有效的方法之一。本书将 Java Web 的知识和实例有机结合起来,一方面,跟踪 Java Web 的发展,适应市场需求,精心选择内容,突出重点,强调实用,使知识讲解全面、系统;另一方面,设计典型的实例,将实例融入到知识讲解中,使知识与实例相辅相成,既有利于学生学习知识,又有利于指导学生实践。另外,本书在每一章的后面还提供了习题和实验,方便读者及时验证自己的学习成果(包括理论知识和动手实践能力)。

本书作为教材使用时,课堂教学建议 48~56 学时,实验教学建议 24~29 学时。各章主要内容和学时建议分配如下,老师可以根据实际教学情况进行调整。

章节	主要内容	课堂学时	实验学时
第 1 章	Web 应用开发概述,包括网络程序开发体系结构、Web 简介、Web 开发技术	2	0
第 2 章	HTML 与 CSS 网页开发基础,包括 HTML 标记语言、CSS 样式表、CSS 3 的新特征、综合实例——应用 DIV+CSS 布局许愿墙主界面	3	2
第 3 章	JavaScript 脚本语言,包括了解 JavaScript、JavaScript 语言基础、流程控制语句、函数、事件处理、常用对象、DOM 技术、综合实例——将数字字符串格式化为指定长度	4	3
第 4 章	搭建 Java Web 开发环境,包括 JSP 概述、JDK 的安装与配置、Tomcat 的安装与配置、Eclipse 开发工具的安装与使用、综合实例——使用 Eclipse 开发一个 Java Web 网站	2	2
第 5 章	JSP 基本语法,包括 JSP 页面的基本构成、指令标识、脚本标识、注释、动作标识、综合实例——包含需要传递参数的文件	3	2
第 6 章	JSP 的内置对象,包括内置对象概述、request 请求对象、response 响应对象、out 输出对象、session 会话对象、application 应用对象、其他内置对象、综合实例——应用 session 实现用户登录	3	2
第 7 章	JavaBean 技术,包括 JavaBean 技术简介、JavaBean 的应用、综合实例——应用 JavaBean 解决中文乱码	2	2
第 8 章	Servlet 技术,包括 Servlet 基础、Servlet 开发、Servlet 过滤器、Servlet 监听器、综合实例——应用监听器统计在线用户	3	3
第 9 章	数据库应用开发,包括 JDBC 简介、JDBC API、连接数据库、JDBC 操作数据库、综合实例——分页查询	4	2

续表

章节	主要内容	课堂学时	实验学时
第 10 章	EL (表达式语言), 包括 EL (表达式语言) 概述、与低版本的环境兼容——禁用 EL、保留的关键字、EL 的运算符及优先级、EL 的隐含对象、综合实例——通过 EL 显示投票结果	2	1
第 11 章	JSTL 核心标签库, 包括 JSTL 标签库简介、JSTL 的下载与配置、表达式标签、URL 相关标签、流程控制标签、循环标签、综合实例——JSTL 在电子商城中的应用	3	1
第 12 章	Ajax 技术, 包括 Ajax 简介、使用 XMLHttpRequest 对象、传统 Ajax 的工作流程、应用 jQuery 实现 Ajax、综合实例——多级联动下拉列表	3	2
第 13 章	Struts 2 框架技术, 包括 Struts 2 框架概述、Struts 2 入门、Action 对象、Struts 2 的配置文件、Struts 2 的标签库、Struts 2 的开发模式、Struts 2 的拦截器、综合实例——利用 Struts 2 实现简单的投票器	4	2
第 14 章	Hibernate 技术, 包括初识 Hibernate、Hibernate 入门、Hibernate 数据持久化、使用 Hibernate 的缓存、HQL (Hibernate 查询语言)、综合实例——修改员工信息	3	2
第 15 章	Spring 技术, 包括 Spring 概述、依赖注入、AOP 概述、Spring 的切入点、Aspect 对 AOP 的支持、Spring 持久化、综合实例——整合 Spring 与 Hibernate 向表中添加信息	3	3
第 16 章	综合案例——基于 Struts 2+Hibernate+Spring 的网络商城, 包括需求分析、系统设计、系统开发及运行环境、数据库与数据表设计、系统文件夹组织结构、搭建项目环境、公共类设计、登录模块设计、前台商品信息查询模块设计、购物车模块设计、后台商品管理模块设计、后台订单管理模块设计、网站编译与发布	4	0
第 17 章	课程设计——基于 Struts 2 的博客网站, 包括课程设计目的、功能描述、总体设计、数据库设计、实现过程、调试运行、课程设计总结	4	0
第 18 章	课程设计——基于 Servlet 的图书馆管理系统, 包括课程设计目的、功能描述、总体设计、数据库设计、实现过程、调试运行、课程设计总结	4	0

除主编和副主编外, 参加本书编写的还有高贤强、张兵、史召峰。

由于编者水平有限, 书中难免存在疏漏和不足之处, 敬请广大读者批评指正, 使本书得以改进和完善。

编者
2013 年 12 月

目 录

第 1 章 Web 应用开发概述	1	知识点提炼	32
1.1 网络程序开发体系结构	1	习题	32
1.1.1 C/S 结构介绍	1	实验: 编写用户注册表单	32
1.1.2 B/S 结构介绍	2	实验目的	32
1.1.3 两种体系结构的比较	2	实验内容	33
1.2 Web 简介	3	实验步骤	33
1.2.1 什么是 Web	3	第 3 章 JavaScript 脚本语言	34
1.2.2 Web 的工作原理	3	3.1 了解 JavaScript	34
1.2.3 Web 的发展历程	4	3.1.1 什么是 JavaScript	34
1.3 Web 开发技术	4	3.1.2 JavaScript 的主要特征	34
1.3.1 客户端应用技术	5	3.2 JavaScript 语言基础	35
1.3.2 服务器端应用技术	7	3.2.1 JavaScript 的语法	35
知识点提炼	8	3.2.2 JavaScript 中的关键字	36
习题	8	3.2.3 JavaScript 的数据类型	36
第 2 章 HTML 与 CSS 网页		3.2.4 变量的定义及使用	38
开发基础	9	3.2.5 运算符的应用	39
2.1 HTML	9	3.3 流程控制语句	41
2.1.1 创建第一个 HTML 文件	9	3.3.1 if 条件判断语句	42
2.1.2 HTML 文档结构	11	3.3.2 switch 多分支语句	44
2.1.3 HTML 常用标记	11	3.3.3 for 循环语句	46
2.1.4 表格标记	14	3.3.4 while 循环语句	47
2.1.5 HTML 表单标记	15	3.3.5 do...while 循环语句	47
2.1.6 超链接与图片标记	19	3.3.6 break 与 continue 语句	48
2.2 CSS 样式表	21	3.4 函数	49
2.2.1 CSS 规则	21	3.4.1 函数的定义	49
2.2.2 CSS 选择器	21	3.4.2 函数的调用	50
2.2.3 在页面中包含 CSS	23	3.5 事件处理	51
2.3 CSS 3 的新特征	25	3.5.1 什么是事件处理程序	51
2.3.1 模块与模块化结构	25	3.5.2 JavaScript 常用事件	51
2.3.2 一个简单的 CSS 3 实例	25	3.5.3 事件处理程序的调用	52
2.4 综合实例——应用 DIV+CSS 布局		3.6 常用对象	53
许愿墙主界面	27	3.6.1 Window 对象	53

3.6.2 String 对象.....	56	4.4.3 安装 Eclipse 中文语言包.....	83
3.6.3 Date 对象.....	60	4.4.4 Eclipse 工作台.....	83
3.7 DOM 技术.....	62	4.5 综合实例——使用 Eclipse	
3.7.1 DOM 的分层结构.....	62	开发一个 Java Web 网站.....	84
3.7.2 遍历文档.....	63	知识点提炼.....	88
3.7.3 获取文档中的指定元素.....	64	习题.....	89
3.7.4 操作文档.....	64	实验：创建并发布一个 Java Web 网站.....	89
3.8 综合实例——将数字字符串格式化		实验目的.....	89
为指定长度.....	66	实验内容.....	89
知识点提炼.....	67	实验步骤.....	89
习题.....	68	第 5 章 JSP 基本语法.....	91
实验：验证用户注册信息的合法性.....	68	5.1 JSP 页面的基本构成.....	91
实验目的.....	68	5.2 指令标识.....	92
实验内容.....	68	5.2.1 page 指令.....	93
实验步骤.....	68	5.2.2 include 指令.....	95
第 4 章 搭建 Java Web		5.2.3 taglib 指令.....	96
开发环境.....	71	5.3 脚本标识.....	97
4.1 JSP 概述.....	71	5.3.1 JSP 表达式 (Expression).....	97
4.1.1 Java 的体系结构.....	71	5.3.2 声明标识 (Declaration).....	97
4.1.2 JSP 技术特征.....	72	5.3.3 代码片段.....	98
4.1.3 JSP 页面的执行过程.....	73	5.4 注释.....	99
4.1.4 在 JSP 中应用 MVC 架构.....	74	5.4.1 HTML 中的注释.....	99
4.1.5 JSP 开发及运行环境.....	74	5.4.2 隐藏注释.....	100
4.2 JDK 的安装与配置.....	76	5.4.3 动态注释.....	100
4.2.1 JDK 的下载与安装.....	76	5.4.4 代码片段中的注释.....	100
4.2.2 在 Windows 系统下配置		5.5 动作标识.....	102
和测试 JDK.....	77	5.5.1 操作 JavaBean 的动作标识.....	102
4.3 Tomcat 的安装与配置.....	78	5.5.2 包含外部文件的动作	
4.3.1 下载和安装 Tomcat 服务器.....	78	标识<jsp:include>.....	106
4.3.2 Tomcat 的目录结构.....	81	5.5.3 执行请求转发的动作	
4.3.3 修改 Tomcat 的默认端口.....	81	标识<jsp:forward>.....	107
4.3.4 部署 Web 应用.....	81	5.5.4 设置参数的子动作	
4.4 Eclipse 开发工具的安装与使用.....	82	标识<jsp:param>.....	108
4.4.1 Eclipse 的下载与安装.....	82	5.6 综合实例——包含需要	
4.4.2 启动 Eclipse.....	82	传递参数的文件.....	109

知识点提炼	111	6.7.2 获取页面上下文的	
习题	111	pageContext 对象	130
实验: 动态添加下拉列表的列表项	111	6.7.3 获取 web.xml 配置信息的	
实验目的	111	config 对象	130
实验内容	111	6.7.4 获取异常信息的 exception 对象	130
实验步骤	112	6.8 综合实例——应用 session 实现	
第 6 章 JSP 的内置对象	113	用户登录	131
6.1 内置对象概述	113	知识点提炼	133
6.2 request 请求对象	114	习题	134
6.2.1 获取访问请求参数	114	实验: 带验证码的用户登录	134
6.2.2 获取表单提交的信息	115	实验目的	134
6.2.3 解决中文乱码	115	实验内容	134
6.2.4 通过 request 对象进行数据传递	116	实验步骤	134
6.2.5 获取客户端信息	117	第 7 章 JavaBean 技术	136
6.2.6 获取 cookie	118	7.1 JavaBean 技术简介	136
6.2.7 显示国际化信息	120	7.1.1 JavaBean 概述	136
6.3 response 响应对象	120	7.1.2 JavaBean 种类	137
6.3.1 实现重定向页面	120	7.2 JavaBean 的应用	138
6.3.2 处理 HTTP 文件头	121	7.2.1 获取 JavaBean 属性	138
6.3.3 设置输出缓冲	123	7.2.2 对 JavaBean 属性赋值	140
6.4 out 输出对象	123	7.2.3 如何在 JSP 页面中	
6.4.1 向客户端输出数据	123	应用 JavaBean	141
6.4.2 管理相应缓冲区	124	7.3 综合实例——应用 JavaBean	
6.5 session 会话对象	125	解决中文乱码	143
6.5.1 创建及获取客户的会话	125	知识点提炼	145
6.5.2 从会话中移除指定的对象	125	习题	146
6.5.3 设置 session 的有效时间	126	实验: 转换输入文本中的回车和空格	146
6.5.4 销毁 session	126	实验目的	146
6.6 application 应用对象	126	实验内容	146
6.6.1 访问应用程序初始化参数	126	实验步骤	146
6.6.2 应用程序环境属性管理	127	第 8 章 Servlet 技术	149
6.6.3 应用 application 实现		8.1 Servlet 基础	149
网页计数器	128	8.1.1 Servlet 体系结构	149
6.7 其他内置对象	129	8.1.2 Servlet 技术特点	150
6.7.1 应答与请求的 page 对象	129	8.1.3 Servlet 与 JSP 的区别	151

8.2 Servlet 开发.....	151	9.3 连接数据库.....	179
8.2.1 创建 Servlet.....	152	9.3.1 加载 JDBC 驱动程序.....	180
8.2.2 Servlet 配置.....	155	9.3.2 创建数据库连接.....	180
8.2.3 在 Servlet 中实现页面转发.....	156	9.3.3 执行 SQL 语句.....	181
8.2.4 Servlet 处理表单数据.....	156	9.3.4 获得查询结果.....	181
8.3 Servlet 过滤器.....	158	9.3.5 关闭连接.....	182
8.3.1 什么是过滤器.....	158	9.4 JDBC 操作数据库.....	182
8.3.2 过滤器的核心对象.....	159	9.4.1 添加数据.....	182
8.3.3 过滤器创建与配置.....	160	9.4.2 查询数据.....	185
8.4 Servlet 监听器.....	163	9.4.3 修改数据.....	188
8.4.1 Servlet 监听器简介.....	163	9.4.4 删除数据.....	190
8.4.2 Servlet 监听器的原理.....	163	9.4.5 批处理.....	190
8.4.3 Servlet 上下文监听.....	164	9.4.6 调用存储过程.....	193
8.4.4 HTTP 会话监听.....	165	9.5 综合实例——分页查询.....	195
8.4.5 Servlet 请求监听.....	166	知识点提炼.....	199
8.5 综合实例——应用监听器统计		习题.....	200
在线用户.....	166	实验：实现批量删除数据.....	200
知识点提炼.....	169	实验目的.....	200
习题.....	169	实验内容.....	200
实验：编写一个字符编码过滤器.....	169	实验步骤.....	200
实验目的.....	169	第 10 章 EL (表达式语言).....	204
实验内容.....	169	10.1 EL (表达式语言) 概述.....	204
实验步骤.....	170	10.1.1 EL 的基本语法.....	204
第 9 章 数据库应用开发.....	173	10.1.2 EL 的特点.....	205
9.1 JDBC 简介.....	173	10.2 与低版本的环境兼容——禁用 EL.....	205
9.1.1 JDBC 技术介绍.....	173	10.2.1 使用反斜杠“\”符号.....	205
9.1.2 JDBC 驱动程序.....	174	10.2.2 使用 page 指令.....	206
9.2 JDBC API.....	174	10.2.3 在 web.xml 文件中配置	
9.2.1 Driver 接口.....	175	<el-ignored>元素.....	206
9.2.2 Connection 接口.....	175	10.3 保留的关键字.....	206
9.2.3 DriverManager 类.....	176	10.4 EL 的运算符及优先级.....	207
9.2.4 Statement 接口.....	177	10.4.1 通过 EL 访问数据.....	207
9.2.5 PreparedStatement 接口.....	177	10.4.2 在 EL 中进行算术运算.....	209
9.2.6 CallableStatement 接口.....	178	10.4.3 在 EL 中判断对象是否为空.....	209
9.2.7 ResultSet 接口.....	179	10.4.4 在 EL 中进行逻辑关系运算.....	210

10.4.5 在 EL 中进行条件运算.....	211	11.6.2 <c:forTokens>迭代标签.....	241
10.5 EL 的隐含对象.....	212	11.7 综合实例——JSTL 在电子商城中的	
10.5.1 页面上下文对象.....	212	应用.....	242
10.5.2 访问作用域范围的隐含对象.....	213	知识点提炼.....	245
10.5.3 访问环境信息的隐含对象.....	214	习题.....	245
10.6 综合实例——通过 EL 显示		实验：显示数据库中的图书信息.....	245
投票结果.....	216	实验目的.....	245
知识点提炼.....	219	实验内容.....	245
习题.....	219	实验步骤.....	246
实验：应用 EL 访问 JavaBean 属性.....	220	第 12 章 Ajax 技术.....	248
实验目的.....	220	12.1 Ajax 简介.....	248
实验内容.....	220	12.1.1 什么是 Ajax.....	248
实验步骤.....	220	12.1.2 Ajax 开发模式与传统开发模式	
第 11 章 JSTL 核心标签库.....	222	的比较.....	248
11.1 JSTL 标签库简介.....	222	12.1.3 Ajax 的优点.....	249
11.2 JSTL 的下载与配置.....	224	12.2 使用 XMLHttpRequest 对象.....	250
11.2.1 下载 JSTL 标签库.....	224	12.2.1 初始化 XMLHttpRequest 对象.....	250
11.2.2 配置 JSTL.....	224	12.2.2 XMLHttpRequest 对象的	
11.3 表达式标签.....	227	常用方法.....	251
11.3.1 <c:out>输出标签.....	227	12.2.3 XMLHttpRequest 对象的	
11.3.2 <c:set>变量设置标签.....	228	常用属性.....	252
11.3.3 <c:remove>变量移除标签.....	230	12.3 传统 Ajax 的工作流程.....	253
11.3.4 <c:catch>捕获异常标签.....	231	12.3.1 发送请求.....	253
11.4 URL 相关标签.....	232	12.3.2 处理服务器响应.....	254
11.4.1 <c:import>导入标签.....	232	12.3.3 一个完整的实例——检测用户名	
11.4.2 <c:url>动态生成 URL 标签.....	233	是否唯一.....	256
11.4.3 <c:redirect>重定向标签.....	234	12.4 应用 jQuery 实现 Ajax.....	258
11.4.4 <c:param>传递参数标签.....	235	12.4.1 jQuery 简介.....	258
11.5 流程控制标签.....	235	12.4.2 应用 load()方法发送请求.....	260
11.5.1 <c:if>条件判断标签.....	235	12.4.3 发送 GET 和 POST 请求.....	261
11.5.2 <c:choose>条件选择标签.....	236	12.4.4 服务器返回的数据格式.....	265
11.5.3 <c:when>条件测试标签.....	237	12.4.5 使用\$.ajax()方法.....	268
11.5.4 <c:otherwise>其他条件标签.....	238	12.5 综合实例——多级联动下拉列表.....	270
11.6 循环标签.....	239	知识点提炼.....	276
11.6.1 <c:forEach>循环标签.....	239	习题.....	276

实验：实时显示公告信息	276	13.7.2 拦截器 API	300
实验目的	276	13.7.3 使用拦截器	301
实验内容	276	13.8 综合实例——利用 Struts 2	
实验步骤	277	实现简单的投票器	302
第 13 章 Struts 2 框架技术	279	知识点提炼	303
13.1 Struts 2 框架概述	279	习题	303
13.1.1 理解 MVC 的原理	279	实验：Struts 2 标签下的用户注册	304
13.1.2 Struts 2 框架的产生	280	实验目的	304
13.1.3 Struts 2 的结构体系	280	实验内容	304
13.2 Struts 2 入门	281	实验步骤	304
13.2.1 获取与配置 Struts 2	281	第 14 章 Hibernate 技术	306
13.2.2 创建第一个 Struts 2 程序	282	14.1 初识 Hibernate	306
13.3 Action 对象	284	14.1.1 ORM 原理	306
13.3.1 认识 Action 对象	284	14.1.2 Hibernate 简介	306
13.3.2 请求参数的注入原理	285	14.2 Hibernate 入门	307
13.3.3 Action 的基本流程	285	14.2.1 获取 Hibernate	307
13.3.4 动态 Action	285	14.2.2 Hibernate 配置文件	307
13.3.5 应用动态 Action	286	14.2.3 编写持久化类	309
13.4 Struts 2 的配置文件	288	14.2.4 Hibernate 映射	310
13.4.1 Struts 2 的配置文件类型	288	14.2.5 Hibernate 主键策略	311
13.4.2 配置 Struts 2 包	288	14.3 Hibernate 数据持久化	311
13.4.3 配置名称空间	289	14.3.1 Hibernate 实例状态	312
13.4.4 Action 的相关配置	289	14.3.2 Hibernate 初始化类	312
13.4.5 使用通配符简化配置	291	14.3.3 保存数据	314
13.4.6 配置返回结果	291	14.3.4 查询数据	316
13.5 Struts 2 的标签库	292	14.3.5 删除数据	316
13.5.1 应用数据标签	292	14.3.6 修改数据	317
13.5.2 应用控制标签	294	14.3.7 延迟加载	318
13.5.3 应用表单标签	296	14.4 使用 Hibernate 的缓存	319
13.6 Struts 2 的开发模式	297	14.4.1 使用一级缓存	319
13.6.1 实现与 Servlet API 的交互	297	14.4.2 配置和使用二级缓存	320
13.6.2 Domain Model (域模型)	297	14.5 HQL (Hibernate 查询语言)	321
13.6.3 Model Driven (驱动模型)	298	14.5.1 HQL 概述	321
13.7 Struts 2 的拦截器	299	14.5.2 查询实体对象	322
13.7.1 拦截器概述	299	14.5.3 条件查询	323

14.5.4	HQL 参数绑定机制.....	323	15.5.3	DefaultPointcutAdvisor 切入点 配置器.....	343
14.5.5	排序查询.....	324	15.5.4	NameMatchMethodPointcutAdvisor 切入点配置器.....	344
14.5.6	应用聚合函数.....	324	15.6	Spring 持久化.....	344
14.5.7	分组方法.....	324	15.6.1	DAO 模式.....	344
14.5.8	联合查询.....	325	15.6.2	Spring 的 DAO 理念.....	345
14.6	综合实例——修改员工信息.....	325	15.6.3	事务管理.....	347
	知识点提炼.....	327	15.6.4	应用 JdbcTemplate 操作数据库.....	350
	习题.....	328	15.6.5	与 Hibernate 整合.....	351
	实验：员工信息的模糊查询.....	328	15.7	综合实例——整合 Spring 与 Hibernate 向表中添加信息.....	353
	实验目的.....	328		知识点提炼.....	354
	实验内容.....	328		习题.....	354
	实验步骤.....	328		实验：用 AOP 实现用户注册.....	354
				实验目的.....	354
				实验内容.....	355
				实验步骤.....	355
第 15 章	Spring 技术.....	330	第 16 章	综合案例——基于 Struts+Hibernate+Spring 的网络商城.....	357
15.1	Spring 概述.....	330	16.1	需求分析.....	357
15.1.1	Spring 组成.....	330	16.2	系统设计.....	357
15.1.2	获取 Spring.....	331	16.2.1	系统目标.....	357
15.1.3	配置 Spring.....	331	16.2.2	系统功能结构.....	358
15.1.4	使用 BeanFactory 管理 Bean.....	332	16.2.3	系统业务流程图.....	358
15.1.5	应用 ApplicationContext.....	333	16.3	系统开发及运行环境.....	359
15.2	依赖注入.....	334	16.4	数据库与数据表设计.....	359
15.2.1	控制反转与依赖注入.....	334	16.4.1	E-R 图设计.....	359
15.2.2	配置 Bean.....	335	16.4.2	创建数据库及数据表.....	360
15.2.3	Setter 注入.....	335	16.5	系统文件夹组织结构.....	362
15.2.4	引用其他 Bean.....	336	16.6	搭建项目环境.....	362
15.3	AOP 概述.....	337	16.6.1	配置 Struts 2.....	363
15.3.1	AOP 术语.....	338	16.6.2	配置 Hibernate.....	365
15.3.2	AOP 的简单实现.....	339	16.6.3	配置 Spring.....	366
15.4	Spring 的切入点.....	340			
15.4.1	静态切入点与动态切入点.....	340			
15.4.2	深入静态切入点.....	341			
15.4.3	深入切入点底层.....	341			
15.4.4	Spring 中的其他切入点.....	342			
15.5	Aspect 对 AOP 的支持.....	342			
15.5.1	Aspect 概述.....	342			
15.5.2	Spring 中的 Aspect.....	343			

16.6.4 配置 web.xml.....	367	17.3.3 系统流程图.....	399
16.7 公共类设计.....	368	17.4 数据库设计.....	400
16.7.1 Hibernate 的 Session 初始化类.....	368	17.4.1 实体 E-R 图.....	400
16.7.2 泛型工具类.....	369	17.4.2 数据表设计.....	400
16.7.3 数据持久化类.....	370	17.5 实现过程.....	401
16.7.4 分页设计.....	372	17.5.1 公共模块设计.....	401
16.8 登录模块设计.....	375	17.5.2 主页面设计.....	405
16.9 前台商品信息查询模块设计.....	377	17.5.3 用户管理模块设计.....	407
16.9.1 实现商品类别分级查询.....	377	17.5.4 文章模块设计.....	411
16.9.2 实现商品搜索.....	378	17.6 调试运行.....	415
16.10 购物车模块设计.....	379	17.7 课程设计总结.....	416
16.10.1 实现购物车的基本功能.....	380		
16.10.2 实现订单的相关功能.....	383	第 18 章 课程设计——基于 Servlet	
16.11 后台商品管理模块设计.....	385	的图书馆管理系统.....	417
16.11.1 查询商品信息.....	385	18.1 课程设计目的.....	417
16.11.2 修改商品信息.....	387	18.2 功能描述.....	418
16.11.3 删除商品信息.....	388	18.3 总体设计.....	418
16.11.4 添加商品信息.....	388	18.3.1 构建开发环境.....	418
16.12 后台订单管理模块设计.....	388	18.3.2 网站功能结构.....	418
16.12.1 实现后台订单查询.....	389	18.3.3 系统流程图.....	419
16.12.2 实现后台订单状态管理.....	390	18.4 数据库设计.....	419
16.13 网站编译与发布.....	392	18.4.1 实体图.....	420
16.13.1 网站编译.....	392	18.4.2 数据表设计.....	421
16.13.2 网站发布.....	392	18.5 实现过程.....	423
第 17 章 课程设计——基于 Struts 2		18.5.1 系统登录设计.....	423
的博客网站.....	397	18.5.2 主界面设计.....	425
17.1 课程设计目的.....	397	18.5.3 图书借阅设计.....	427
17.2 功能描述.....	397	18.5.4 图书续借设计.....	431
17.3 总体设计.....	398	18.5.5 图书归还设计.....	434
17.3.1 构建开发环境.....	398	18.6 调试运行.....	435
17.3.2 网站功能结构.....	398	18.7 课程设计总结.....	436

第 1 章

Web 应用开发概述

本章要点:

- 什么是 C/S 结构和 B/S 结构
- C/S 结构和 B/S 结构的比较
- 什么是 Web
- Web 的工作原理
- Web 的发展历程
- Web 开发技术

随着网络技术的迅猛发展,国内外的信息化建设已经进入以 Web 应用为核心的阶段。作为即将进入 Web 应用开发阵营的准程序员,首先需要对网络程序开发的体系结构、Web 以及 Web 开发技术有所了解。本章将对网络程序开发体系结构、什么是 Web、Web 的工作原理、Web 的发展历程和 Web 开发技术进行介绍。

1.1 网络程序开发体系结构

随着网络技术的不断发展,单机的软件程序开始难以满足网络用户的需要,为此,各种各样的网络程序开发体系结构应运而生。其中,运用最多的网络程序开发体系结构可以分为两种:一种是基于浏览器/服务器的 B/S 结构,另一种是基于客户端/服务器的 C/S 结构。下面进行详细介绍。

1.1.1 C/S 结构介绍

C/S 是 Client/Server 的缩写,即客户端/服务器结构。在这种结构中,服务器通常采用高性能的 PC 或工作站,并采用大型数据库系统(如 Oracle 或 SQL Server),客户端则需要安装专用的客户端软件,如图 1-1 所示。这种结构可以充分利用两端硬件环境的优势,将任务合理分配到客户端和服务器,从而降低了系统的通信开销。在 2000 年以前,C/S 结构占据网络程序开发领域的主流。

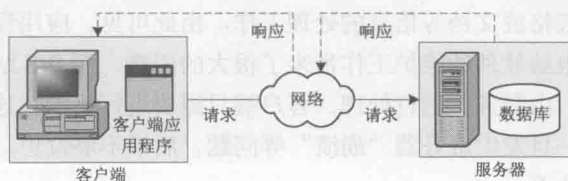


图 1-1 C/S 体系结构

1.1.2 B/S 结构介绍

B/S 是 Browser/Server 的缩写,即浏览器/服务器结构。在这种结构中,客户端不需要开发任何用户界面,而统一采用如 IE 和火狐等浏览器,通过 Web 浏览器向 Web 服务器发送请求,由 Web 服务器进行处理,并将处理结果逐级传回客户端,如图 1-2 所示。这种结构利用不断成熟和普及的浏览器技术,实现原来需要复杂专用软件才能实现的强大功能,从而节约了开发成本,是一种全新的软件体系结构。这种体系结构已经成为当今应用软件的首选体系结构。

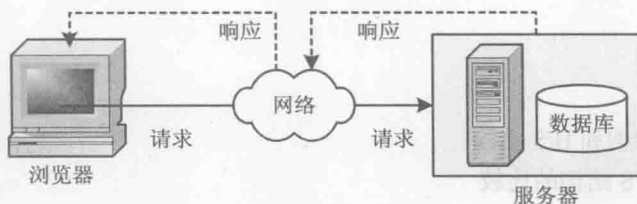


图 1-2 B/S 体系结构



说明

B/S 由美国 Microsoft 公司研发, C/S 最早由美国 Borland 公司研发。

1.1.3 两种体系结构的比较

C/S 结构和 B/S 结构是当今世界网络程序开发体系结构的两大主流。目前,这两种体系结构都有自己的市场份额和客户群。但是,这两种体系结构又各有各的优点和缺点。下面将从以下 3 个方面进行比较说明。

1. 开发和维护成本

C/S 结构的开发和维护成本都比 B/S 高。采用 C/S 结构时,对于不同的客户端,要开发不同的程序,而且软件的安装、调试和升级均需要在所有的客户机上进行。例如,如果一个企业共有 10 个客户站点使用一套 C/S 结构的软件,则这 10 个客户站点都需要安装客户端程序。当这套软件进行了哪怕很微小的改动,系统维护员都必须将客户端原有的软件卸载,再安装新的版本并进行配置,最可怕的是客户端的维护工作必须不折不扣地进行 10 次。若某个客户端忘记进行这样的更新,则该客户端将会因软件版本不一致而无法工作。而采用 B/S 结构则不必在客户端进行安装及维护。如果我们将前面企业的 C/S 结构的软件换成 B/S 结构的,这样在软件升级后,系统维护员只需要将服务器的软件升级到最新版本,而其他客户端只要重新登录系统就可以使用最新版本的软件了。

2. 客户端负载

C/S 结构的客户端不仅负责与用户的交互、收集用户信息,而且还需要完成通过网络向服务器请求对数据库、电子表格或文档等信息的处理工作。由此可见,应用程序的功能越复杂,客户端程序也就越庞大,这也给软件的维护工作带来了很大的困难。而 B/S 结构的客户端把事务处理逻辑部分交给了服务器,由服务器进行处理,客户端只需要进行显示,这样将使应用程序服务器的运行数据负荷较重,一旦发生服务器“崩溃”等问题,后果不堪设想。因此,许多单位都备有数据库存储服务器,以防万一。

3. 安全性

C/S 结构适用于专人使用的系统, 可以通过严格的管理派发软件, 达到保证系统安全的目的, 这样的软件相对来说安全性比较高。而对于 B/S 结构的软件, 由于使用的人数较多且不固定, 因此相对来说安全性就会低些。

由此可见, B/S 相对于 C/S 具有更多的优势, 现今大量的应用程序开始转移到应用 B/S 结构, 许多软件公司也争相开发 B/S 版的软件, 也就是 Web 应用程序。随着 Internet 的发展, 基于 HTTP 和 HTML 标准的 Web 应用呈几何数量级的增长, 而这些 Web 应用又由各种 Web 技术所开发。

1.2 Web 简介

Web 是 WWW (World Wide Web) 的简称, 引申为“环球网”, 在不同的领域有不同的含义。针对普通的用户, Web 仅仅是一种环境——互联网的使用环境; 而针对网站制作或设计者, 它是一系列技术的总称 (包括网站的页面布局、后台程序、美工、数据库领域等)。下面将对什么是 Web 和 Web 的工作原理进行详细介绍。

1.2.1 什么是 Web

Web 的本意是网和网状物, 现在被广泛译作网络、万维网或互联网等技术领域。它是一种基于超文本方式工作的信息系统。作为一个能够处理文字、图像、声音和视频等多媒体信息的综合系统, 它提供了丰富的信息资源, 这些信息资源通常表现为以下 3 种形式。

- 超文本 (hypertext)

超文本是一种全局性的信息结构, 它将文档中的不同部分通过关键字建立链接, 使信息得以用交互方式搜索。

- 超媒体 (hypermedia)

超媒体是超文本和多媒体在信息浏览环境下的结合。有了超媒体, 用户不仅能从一个文本跳到另一个文本, 而且可以显示图像, 播放动画、音频和视频等。

- 超文本传送协议 (HTTP)

超文本传送协议是超文本在互联网上的传送协议。

1.2.2 Web 的工作原理

在 Web 中, 信息资源将以 Web 页面的形式分别存放在各个 Web 服务器上, 用户可以通过浏览器选择并浏览所需的信息。Web 的具体工作流程如图 1-3 所示。

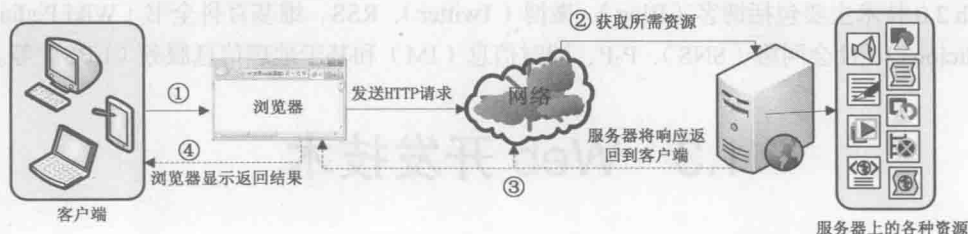


图 1-3 Web 的工作流程图

从图 1-3 中可以看出, Web 的工作流程大致可以分为以下 4 个步骤。

- (1) 用户在浏览器中输入 URL 地址 (即统一资源定位符), 或者通过超链接方式链接到一个网页或者网络资源后, 浏览器将该信息转换成标准的 HTTP 请求发送给 Web 服务器。
- (2) 当 Web 服务器接收到 HTTP 请求后, 根据请求内容查找所需的信息资源。
- (3) 找到相应资源后, Web 服务器将该部分资源通过标准的 HTTP 响应发送回浏览器。
- (4) 浏览器将经服务器转换后的 HTML 代码显示给客户端用户。

1.2.3 Web 的发展历程

自从 1989 年 Tim Berners-Lee (蒂姆·伯纳斯·李) 发明 World Wide Web 以来, Web 主要经历了 3 个阶段, 分别是静态文档阶段 (指代 Web 1.0)、动态网页阶段 (指代 Web 1.5) 和 Web 2.0 阶段。下面将对这 3 个阶段进行介绍。

1. 静态文档阶段

处理静态文档阶段的 Web, 主要是用于静态 Web 页面的浏览。用户通过客户端的 Web 浏览器, 可以访问 Internet 上的各个 Web 站点。在每个 Web 站点上保存着提前编写好的 HTML 格式的 Web 页, 以及各 Web 页之间可以实现跳转的超文本链接。通常情况下, 这些 Web 页都是通过 HTML 编写的。由于受低版本 HTML 和旧式浏览器的制约, Web 页面只能包括单纯的文本内容, 浏览器也只能显示呆板的文字信息, 不过这已经基本满足了建立 Web 站点的初衷, 实现了信息资源共享。

随着互联网技术的不断发展以及网上信息呈几何级数的增加, 人们逐渐发现手工编写包含所有信息和内容的页面对人力和物力都是一种极大的浪费, 而且几乎变得难以实现。另外, 这样的页面也无法实现各种动态的交互功能, 这就促使 Web 技术进入了发展的第二阶段——动态网页阶段。

2. 动态网页阶段

为了克服静态页面的不足, 人们将传统单机环境下的编程技术与 Web 技术相结合, 从而形成新的网络编程技术。网络编程技术通过在传统的静态页面中加入各种程序和逻辑控制, 从而实现动态和个性化的交流与互动。我们将这种使用网络编程技术创建的页面称为动态页面, 其后缀通常是 .jsp、.php 和 .asp 等, 而静态页面的后缀通常是 .htm、.html 和 .shtml 等。



这里说的动态网页, 与网页上的各种动画、滚动字幕等视觉上的动态效果没有直接关系, 动态网页也可以是纯文字内容的, 这些只是网页具体内容的表现形式。无论网页是否具有动态效果, 采用动态网络编程技术生成的网页都称为动态网页。

3. Web 2.0 阶段

随着互联网技术的不断发展, 又提出了一种新的互联网模式——Web 2.0。这种模式更加以用户为中心, 通过网络应用 (Web Applications) 促进网络上人与人之间的信息交换和协同合作。

Web 2.0 技术主要包括博客 (Blog)、微博 (Twitter)、RSS、维基百科全书 (Wiki Pedia)、网摘 (Delicious)、社会网络 (SNS)、P2P、即时信息 (IM) 和基于地理信息服务 (LBS) 等。

1.3 Web 开发技术

Web 是一种典型的分布式应用架构。Web 应用中的每一次信息交换都要涉及客户端和服务