

从基础到应用



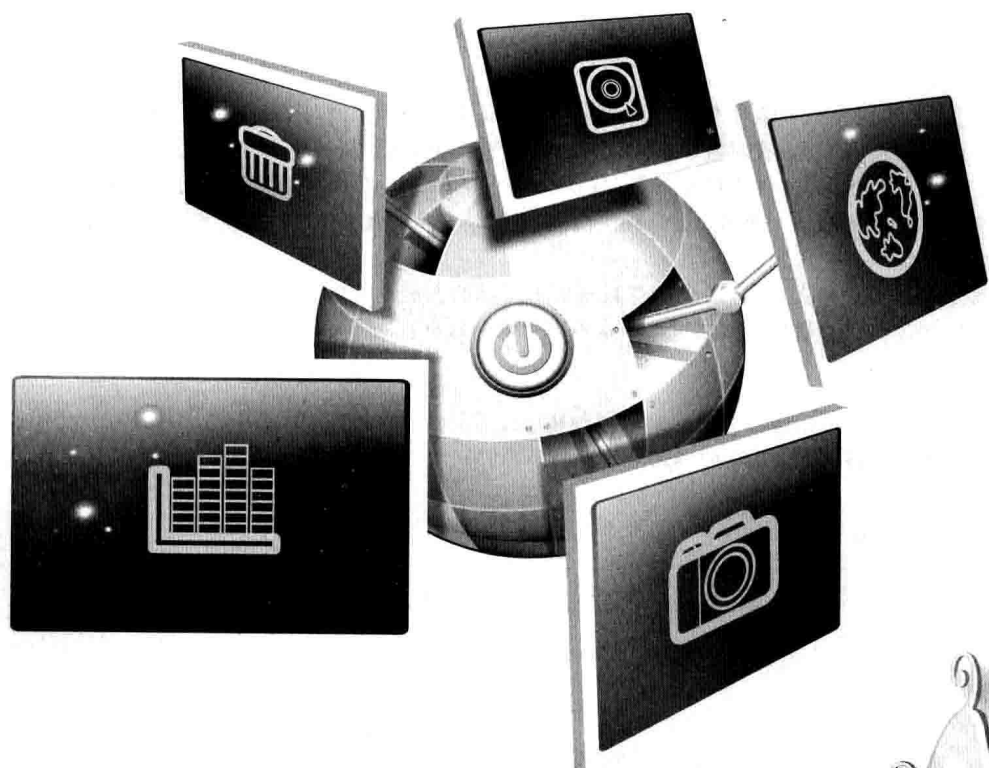
孙宇霞 编著
孙晓芳



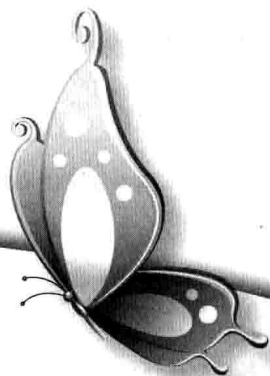
Java Web编程 从基础到应用

清华大学出版社





孙宇霞 编著
孙晓芳



Java Web编程 从基础到应用

清华大学出版社
北京

内 容 简 介

本书由浅入深、循序渐进地介绍了学习 Java Web 程序开发必备的知识和技能。主要内容包括构建 Java Web 开发环境、JSP 页面元素、JSP 内置对象、JavaBean、数据库访问技术、Servlet 技术、使用 EL 表达式、JSTL 标准标签、JSP 操作 XML、文件上传与下载、使用 Ajax 技术、Struts 2 框架的具体应用、Hibernate 框架应用和 Spring 框架应用,最后以一个 Struts 2+Hibernate+Spring 三大框架整合的实例,综合地介绍了 Java Web 在实际开发中的应用。

本书适合 Java Web 初学者快速入门,也适合已有 Java Web 基础的人员完善自己的 Java Web 知识体系。另外,对于大中专院校和培训班的学生,本书更是一本不可多得的教材。

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

Java Web 编程从基础到应用/孙宇霞等编著. —北京:清华大学出版社,2014

从基础到应用

ISBN 978-7-302-32881-0

I. ①J… II. ①孙… III. ①JAVA 语言-程序设计-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 136391 号

责任编辑:夏兆彦

封面设计:胡文航

责任校对:胡伟民

责任印制:刘海龙

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

印 装 者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:32.5 字 数:812 千字
(附光盘 1 张)

版 次:2014 年 3 月第 1 版

印 次:2014 年 3 月第 1 次印刷

印 数:1~4000

定 价:59.00 元

产品编号:045968-01

FOREWORD

前言

Java 是 Sun 公司推出的能够跨越多平台的、可移植性最高的编程语言。Java 技术具有卓越的通用性、高效性、平台移植性和安全性，广泛应用于个人 PC、数据中心、游戏控制台、科学超级计算机、移动电话和互联网，同时拥有全球最大的开发者专业社群。从而使其成为应用范围最广泛的开发语言，特别是在 Web 程序开发方面。

在市场上可以看到大量 Java Web 方面的书籍，这些书籍写的都很不错。可是，很多读者看过之后虽然对知识点都了解了，但是依然不会动手操作，想要自己独立开发一个项目很难。作者研究其中原因发现，大多书籍比较重视知识点的介绍而忽略了实践练习的重要性。本书从初学者的角度出发，循序渐进地讲解使用 Java 语言和开源框架进行 Web 程序开发应该掌握的技术，其中主要包括 JSP/Servlet 技术、常用的 Web 开发框架和 Web 开发实用技术等，这些知识是每一个程序员都应该具备的。

本书内容

第 1 章：构建 Java Web 应用程序。主要介绍了 Java Web 应用技术和服务器的安装；重点介绍了 JDK 的安装和配置以及 Tomcat 与 MyEclipse 的集成。

第 2 章：JSP 页面元素。主要介绍了 JSP 的基本元素和页面指令，重点讲解了脚本和各种动作元素的使用，并且使用本章所学知识完成一个求圆面积的项目案例。

第 3 章：JSP 内置对象。主要介绍 JSP 九大内置对象及其用法以及它们的作用，之后应用 JSP 内置对象做出了一个简单的在线考试系统。

第 4 章：JavaBean 编程技术。主要讲述了 JavaBean 的构成和 JavaBean 中不同类型属性的使用；重点讲述了 JavaBean 在 JSP 页面中的应用以及 JavaBean 不同作用域的应用。

第 5 章：数据库访问技术。主要介绍了 JDBC 的概念、用法以及其驱动，还介绍了连接池连接数据库；重点介绍了在 JSP 中访问数据库的方法，以及连接数据库的常用接口。最后做了一个会员的增删改查系统来巩固本章知识。

第 6 章：Servlet 技术。主要介绍了 Servlet 概念以及配置，介绍了 Servlet 基本类和常用接口；重点介绍了 Servlet 的会话跟踪技术以及过滤器和监听器。最后使用该技术完成了对用户的索引。

第7章：使用 EL 表达式。主要介绍了 EL 表达式语言的概念、变量以及运算符；重点介绍了 EL 表达式的隐含变量和 EL 函数。

第8章：JSTL 标签库。简单介绍了关于核心标签的库的变量、数字和日期时间的格式化；重点讲述了通过 I18N 实现 Web 应用和国际化操作、通过 SQL 标签完成对数据库的操作以及 XML 标签库完成对 XML 的操作；还介绍了通过 JSTL 函数标签库完成常用的函数操作。

第9章：JSP 操作 XML。主要介绍了 XML 文档中的元素定义属性以及解析 XML 文件的方法。最后使用 XML 文件存储数据完成了学生信息管理。

第10章：文件上传与下载。主要介绍了使用 Common-FileUpload 组件实现文件上传以及使用输入输出流实现文件下载。

第11章：应用 Ajax 技术。简单介绍了 Ajax 开发模式的优势；重点讲述了 XMLHttpRequest 对象、通过 Ajax 向服务器端发送请求以及 Ajax 向服务器端发送相应。

第12章：Struts 2 框架。主要介绍了 Struts 2 的配置、各类型标签的使用和输入校验机制；还介绍了文件上传与下载的实现。

第13章：Hibernate 技术。主要介绍了 Session 接口、Hibernate 的持久化以及 Hibernate 的映射文件、映射关系以及 HQL 语言、Hibernate 缓存；最后使用 Hibernate 技术实现了学生信息管理系统。

第14章：Spring 框架。主要介绍了 Spring 框架概述，以及两个非常重要的概念 IoC 和 AOP。最后结合 Struts 2 框架实现了用户注册功能。

第15章：交友社区项目。本章通过一个 Web 项目讲述 Struts 2、Hibernate 和 Spring 三大框架的整合，并且详细讲述了各个模块的实现过程和运行结果。

本书特色

本书采用大量的实例进行讲解，力求通过实际操作使读者轻松掌握 Java Web 技术的使用。本书每章后面都有精选的习题供读者巩固所学知识，每个章节末尾都有实践疑难解答，在这个模块中收集了每章中的重点、难点以及易出错点以问答的形式来告诉读者，使读者更轻松地掌握这些知识。本书难度适中，内容由浅而深、实用性强、覆盖面广、条理清晰。

- 知识全面：本书全面地介绍了 Java Web 的各知识点，具有很强的系统性。
- 示例典型、应用广泛：作者精心挑选了大量示例程序，它们都是根据作者在实际开发中的经验总结而来，涵盖了在实际开发中出现的各种问题。而且有些程序可以直接在项目中使用，无须二次开发。
- 快速掌握：注重技术原理和实际应用的高度融合，通过循序渐进的内容组织，帮助读者快速掌握和应用 Java Web 技术。
- 随书光盘：本书为实例配备了视频教学文件，读者可以通过视频文件更加直观地学习 Java Web 相关知识。
- 网站技术支持：读者在学习或者工作的过程中，如果遇到实际问题，可以直接登录 www.itzcn.com 与我们取得联系，作者会在第一时间内给予帮助。

读者对象

本书具有知识全面、实例精彩、指导性强的特点，力求以全面的知识及丰富的实例来指导读者透彻地学习 Java Web 方面的知识。本书可以作为 JSP 的入门书籍，也可以帮助中级读者提高技能，对高级读者也有一定的启发意义。

本书适合以下人员阅读学习：

- ☐ 有志于从事软件开发的初学者
- ☐ 高等院校计算机相关专业的老师和学生
- ☐ 准备从事软件开发的求职者
- ☐ JSP 初学者
- ☐ MVC 框架编程人员

除了封面署名人员之外，参与本书编写的人员还有马海军、李海庆、陶丽、王咏梅、康显丽、郝军启、朱俊成、宋强、孙洪叶、袁江涛、张东平、吴鹏、王新伟、刘青凤、汤莉、冀明、王超英、王丹花、闫琰、张丽莉、李卫平、王慧、牛红惠、丁国庆、黄锦刚、李旒、王中行、李志国等。在编写过程中难免会有漏洞，欢迎读者通过我们的网站 www.itzcn.com 与我们联系，帮助我们改正提高。

CONTENTS

目 录

第 1 章 构建 Java Web 应用程序	1
1.1 初识 Java Web	1
1.2 Java Web 开发环境概述	2
1.2.1 开发工具包 JDK	2
1.2.2 Web 服务器	3
1.2.3 数据库	4
1.2.4 Web 浏览器	4
1.3 JDK	4
1.3.1 JDK 的下载和安装	5
1.3.2 配置 JDK	8
1.4 安装配置 Tomcat 服务器	9
1.4.1 Tomcat 的下载和安装	9
1.4.2 配置 Tomcat 应用程序服务器	10
1.5 使用 MyEclipse 开发 Web 应用	12
1.5.1 创建项目	13
1.5.2 创建 JSP 文件	14
1.5.3 发布项目到 Tomcat 并运行	15
1.6 习题	16
1.7 实践疑难解答	17
1.7.1 在网页设计中 JSP 和 HTML 有什么区别	17
1.7.2 启动 Tomcat 出现 java.net.BindException 异常	17
第 2 章 JSP 页面元素	19
2.1 JSP 指令元素	19
2.1.1 page 指令	20
2.1.2 include 指令	21
2.1.3 taglib 指令	22
2.2 JSP 脚本元素	23
2.2.1 声明	23
2.2.2 表达式	24
2.2.3 脚本程序	24
2.3 JSP 动作元素	25
2.3.1 <jsp:include>动作	26
2.3.2 <jsp:useBean>动作	27
2.3.3 <jsp:forward>动作	29
2.3.4 <jsp:setProperty>动作和<jsp:getProperty>动作	32

2.3.5	<jsp:param>动作	36	第 4 章	JavaBean 编程技术	75
2.3.6	<jsp:plugin>动作和<jsp:fallback> 动作	37	4.1	JavaBean 概述	75
2.3.7	JSP 注释	40	4.1.1	JavaBean 技术介绍	75
2.4	项目案例: 求圆的周长和面积	41	4.1.2	JavaBean 的分类	76
2.5	习题	42	4.1.3	JavaBean 规范	77
2.6	实践疑难解答	43	4.2	JavaBean 的属性	78
2.6.1	被 include 引入的页面有什么要求 或规范	43	4.2.1	Simple 属性	78
2.6.2	如何获取跳转页面的参数	44	4.2.2	Indexed 属性	79
2.6.3	JSP 中动作元素 include 和 forward 的差别	45	4.2.3	Bound 属性	80
			4.2.4	Constrained 属性	82
第 3 章	JSP 内置对象	47	4.3	更加对象化的 JSP 开发	84
3.1	request 对象	47	4.3.1	创建 JavaBean	84
3.1.1	request 对象方法	47	4.3.2	创建注册页面	85
3.1.2	获取客户端请求参数	48	4.3.3	创建用户信息显示页面	86
3.1.3	获取 HTTP 请求头	52	4.3.4	运行结果	87
3.2	response 对象	54	4.4	JavaBean 与 JSP 的完美结合	88
3.2.1	response 对象方法	54	4.4.1	创建 JavaBean	88
3.2.2	重定向	54	4.4.2	创建学生档案录入页面	89
3.3	out 对象	56	4.4.3	在 JSP 页面中使用 useBean 标签	90
3.4	session 对象	58	4.4.4	运行结果	91
3.4.1	session 对象的方法	58	4.5	JavaBean 的作用域	92
3.4.2	session 对象的 ID	58	4.5.1	JavaBean 的作用域简介	92
3.4.3	session 对象的应用	60	4.5.2	Page 作用域	92
3.5	application 对象	62	4.5.3	Request 作用域	94
3.6	page 与 config 对象	64	4.5.4	Session 作用域	97
3.7	pageContext 对象	65	4.5.5	Application 作用域	98
3.8	exception 对象	67	4.6	项目案例: 对用户登录信息进行检测	99
3.9	综合案例: 在线考试系统	69	4.7	习题	103
3.10	习题	71	4.8	实践疑难解答	104
3.11	实践疑难解答	72	4.8.1	JavaBean 出现找不到属性值的 问题	104
3.11.1	JSP 内置对象为什么可以直接 使用	72	4.8.2	为 JavaBean 属性赋值时出现 500 错误	105
3.11.2	JSP 中的 request 对象可以设置多 个属性	73	第 5 章	数据库访问技术	106
3.11.3	JSP 中的内置对象的作用域是什 么意思	73	5.1	JDBC 概述	106
			5.1.1	什么是 JDBC	106
			5.1.2	JDBC 的 API 介绍	107

5.1.3 JDBC 驱动程序	107	的区别	170
5.2 JDBC 连接数据库常用类	108	6.8.2 关于 Servlet 过滤器的问题	170
5.2.1 管理驱动程序类 DriverManager	108	6.8.3 Servlet 容器调用自定义 Servlet 的问题	171
5.2.2 Connection 接口	110		
5.3 JDBC-ODBC 桥连 Access 数据库	111	第 7 章 使用 EL 表达式	172
5.4 使用连接池连接数据库	114	7.1 EL 简介	172
5.5 JDBC 访问数据库的常用接口	116	7.2 EL 语法	173
5.5.1 Statement 接口	116	7.2.1 EL 变量	173
5.5.2 PreparedStatement 接口	117	7.2.2 EL 运算符	174
5.5.3 CallableStatement 接口	119	7.2.3 运算符的优先级	178
5.5.4 ResultSet 接口	120	7.2.4 EL 表达式中的保留字	178
5.6 在 JSP 页面中直接访问数据库	121	7.2.5 自动转换类型	178
5.7 项目案例: 会员增删改查	124	7.3 EL 隐含对象	179
5.8 习题	132	7.3.1 与范围有关的隐含对象	180
5.9 实践疑难解答	133	7.3.2 与输入有关的隐含对象	180
5.9.1 关于 Connection 接口的问题	133	7.3.3 访问环境信息的隐含对象	183
5.9.2 JDBC 连接数据库传递参数 的问题	134	7.4 EL 函数	185
5.9.3 不同的数据库需要不同的驱动程 序吗	135	7.5 项目案例: 使用 EL 表达式显示留言 信息	187
第 6 章 Servlet 技术	136	7.6 习题	188
6.1 Servlet 概述	136	7.7 实践疑难解答	190
6.2 Servlet 的配置	138	7.7.1 EL 函数报错	190
6.2.1 Servlet 的名字及类的配置	138	7.7.2 EL 表达式能否直接引用 javaBean 中的方法	190
6.2.2 Servlet 初始化参数的配置	139	7.7.3 如何使用 EL 表达式获取 参数值	191
6.3 Java Servlet API	140		
6.4 会话跟踪技术	150	第 8 章 JSTL 标签库	192
6.4.1 三种会话机制	150	8.1 JSTL 标签库简介	192
6.4.2 HTTP 会话	151	8.2 核心标签库	193
6.5 Servlet 过滤器和监听器	157	8.2.1 一般用途的标签	193
6.5.1 过滤器	157	8.2.2 流程控制标签	199
6.5.2 监听器	161	8.2.3 循环控制标签	202
6.6 项目案例: 根据条件查询用户	166	8.2.4 URL 相关标签	206
6.7 习题	169	8.3 I18N 标签库	209
6.8 实践疑难解答	170	8.3.1 数字格式化标签	209
6.8.1 forward 和 response.sendRedirect		8.3.2 日期时间格式化标签	212

8.3.3 Web 应用和消息的国际化 标签.....	215
8.4 SQL 标签库.....	220
8.5 XML 标签库.....	226
8.6 函数标签库.....	229
8.7 项目案例：保留登录信息.....	232
8.8 习题.....	234
8.9 实践疑难解答.....	236
8.9.1 JSTL 标签库异常.....	236
8.9.2 JSTL 格式化输出下午时间却 显示为上午.....	237
第 9 章 JSP 操作 XML	238
9.1 XML 概述.....	238
9.1.1 XML 文档结构.....	238
9.1.2 XML 语法特点.....	240
9.1.3 XML 元素属性.....	240
9.1.4 字符和实体引用.....	241
9.2 XML 的两种语法规则.....	242
9.2.1 文档类型定义.....	242
9.2.2 元素属性的定义.....	243
9.2.3 Schema 文档规则.....	246
9.3 DOM 解析 XML 文档.....	247
9.3.1 DOM 接口.....	247
9.3.2 DOM 读取 XML 文件.....	249
9.4 SAX 解析 XML 文档.....	251
9.4.1 SAX 常用接口.....	252
9.4.2 SAX 读取 XML 文件.....	253
9.5 DOM4J 解析 XML 文档.....	258
9.5.1 DOM4J 介绍.....	258
9.5.2 使用 DOM4J 创建 XML 文档.....	259
9.5.3 使用 DOM4J 解析 XML 文件.....	261
9.6 项目案例：学生信息管理.....	265
9.7 习题.....	270
9.8 实践疑难解答.....	271
9.8.1 DOM4J 怎样获取节点属性.....	271
9.8.2 DOM4J 生成的 XML 文件没 有格式.....	272
9.8.3 DOM4J 中文乱码问题.....	273
第 10 章 文件的上传与下载	274
10.1 文件上传.....	274
10.1.1 Common-FileUpload 组件 的下载.....	274
10.1.2 Common-FileUpload 组件 核心.....	275
10.1.3 简单文件上传.....	277
10.1.4 限制文件上传类型.....	280
10.1.5 读取表单信息.....	282
10.2 文件下载.....	285
10.2.1 文件类 (File).....	285
10.2.2 文件字节输入流.....	286
10.2.3 字节输出流.....	286
10.3 项目案例：音乐收藏夹.....	288
10.4 习题.....	292
10.5 实践疑难解答.....	293
10.5.1 使用 Common-fileUpload 上传获取文本框值的问题.....	293
10.5.2 在文件上传的时候报错.....	294
第 11 章 应用 Ajax 技术	295
11.1 Ajax 简介.....	295
11.1.1 Ajax 开发模式与传统开发模式 的比较.....	295
11.1.2 Ajax 技术的特点.....	296
11.1.3 Ajax 需要注意的几个问题.....	297
11.1.4 Ajax 使用的技术.....	298
11.2 使用 XMLHttpRequest 对象.....	299
11.2.1 初始化 XMLHttpRequest 对象.....	300
11.2.2 XMLHttpRequest 对象的常用 方法.....	301
11.2.3 XMLHttpRequest 对象的常用 属性.....	303
11.3 发送请求与处理响应.....	304
11.3.1 发送请求.....	304
11.3.2 处理服务器响应.....	306
11.4 Ajax 重构.....	311
11.5 项目案例：验证用户名是否被注册.....	316

11.6 习题	318	12.9.1 在 Struts 2 中实现文件上传	371
11.7 实践疑难解答	320	12.9.2 Struts 2 实现文件下载	375
11.7.1 XMLHttpRequest 状态为 0 的问题	320	12.10 项目案例: 对客户信息进行校验	378
11.7.2 Ajax 路径问题	320	12.11 习题	382
第 12 章 Struts 2 框架	323	12.12 实践疑难解答	383
12.1 Struts 2 简介	323	12.12.1 package 配置的命名空间问题	383
12.1.1 MVC 设计模式	323	12.12.2 Struts 2 中迭代 List 时的迭代次数问题	384
12.1.2 MVC 的优点	325	第 13 章 Hibernate 技术	385
12.1.3 Struts 2 的结构体系	326	13.1 ORM 简介	385
12.2 配置 Struts 2 运行环境	327	13.1.1 ORM 的概念	385
12.3 Struts 2 配置文件	328	13.1.2 ORM 优缺点	386
12.3.1 web.xml 配置	328	13.2 Hibernate 简介	386
12.3.2 struts.xml 配置	330	13.2.1 Hibernate 的发展	386
12.4 Action 配置	334	13.2.2 Hibernate 的下载与安装	387
12.4.1 Action 的实现类	334	13.3 简单的 Hibernate 程序	388
12.4.2 Action 配置	336	13.3.1 创建项目	388
12.4.3 动态方法调用	337	13.3.2 创建持久化类	388
12.4.4 使用 method 属性	341	13.3.3 Hibernate 映射文件	389
12.5 Result 配置	343	13.3.4 Hibernate 配置文件	390
12.5.1 Result 映射	343	13.3.5 运行程序	391
12.5.2 Result 类型	344	13.4 Session 接口	393
12.6 Struts 2 的拦截器	349	13.4.1 Configuration	393
12.6.1 拦截器概述	349	13.4.2 SessionFactory	394
12.6.2 配置拦截器	350	13.4.3 Session	394
12.6.3 使用拦截器	352	13.4.4 Transaction	395
12.6.4 拦截器应用	353	13.4.5 Query 和 Criteria	395
12.7 Struts 2 标签库	356	13.5 Hibernate 数据持久化	396
12.7.1 Struts 2 标签库概述	356	13.5.1 Hibernate 实例状态	396
12.7.2 表单标签	357	13.5.2 Hibernate 的增删改查操作	397
12.7.3 非表单标签	360	13.5.3 加载对象	398
12.7.4 控制标签	362	13.6 Hibernate 的缓存	399
12.7.5 数据标签	365	13.6.1 一级缓存	399
12.8 输入校验	368	13.6.2 二级缓存	400
12.8.1 手动输入校验	368	13.7 映射文件配置	402
12.8.2 定义校验规则	368	13.7.1 <hibernate-mapping>元素	402
12.8.3 校验文件的配置风格	369	13.7.2 <class>元素	403
12.9 文件上传与下载	370		

13.7.3	<id>元素	404	14.2.6	引用其他 Bean	446
13.7.4	<property>元素	407	14.2.7	集合类型属性	447
13.8	映射关系	408	14.3	AOP	449
13.8.1	映射单向多对一关联	408	14.3.1	AOP 简介	449
13.8.2	基于外键单向一对一关联	410	14.3.2	通知 (advice)	451
13.8.3	基于主键单向一对一关联	411	14.3.3	切入点	459
13.8.4	映射双向一对一关联	412	14.4	自动代理	466
13.8.5	映射双向一对多关联	414	14.4.1	使用 BeanNameAutoProxy- Creator	466
13.8.6	映射单向多对多关联	415	14.4.2	使用 DefaultAdvisorAutoProxy- Creator	468
13.8.7	映射双向多对多关联	417	14.5	项目案例: 整合 Struts 2 实现用户 注册	468
13.9	HQL 语言	418	14.6	习题	472
13.9.1	HQL 查询语言	418	14.7	实践疑难解答	473
13.9.2	连接查询	420	14.7.1	用了 Spring 之后能否使用 new 创 建对象	473
13.9.3	子查询	423	14.7.2	IOC 和 AOP 有什么作用, 为什么 要用它们	473
13.9.4	参数绑定机制	423			
13.9.5	分页查询	424			
13.10	学生信息管理系统	425			
13.10.1	功能概述	426			
13.10.2	Hibernate 映射文件	427			
13.10.3	Hibernate 配置文件	428			
13.10.4	系统实现	428			
13.11	习题	434			
13.12	实践疑难解答	435			
13.12.1	uniqueResult 的使用问题	435			
13.12.2	update 更新操作	436			
第 14 章	Spring 框架	437	第 15 章	交友社区项目	474
14.1	Spring 概述	437	15.1	系统概述	474
14.1.1	Spring 体系结构	437	15.1.1	需求分析	474
14.1.2	Spring 的下载与安装	439	15.1.2	系统用例图	475
14.1.3	使用 BeanFactory 管理 Bean	439	15.1.3	系统设计	476
14.1.4	ApplicationContext 应用	440	15.2	数据库设计	476
14.2	Spring 依赖注入	441	15.3	搭建 Hibernate+Struts2+Spring 开发 环境	477
14.2.1	依赖注入和控制反转	441	15.3.1	整合 Struts2	478
14.2.2	IOC 的类型	442	15.3.2	加载 Hibernate	479
14.2.3	Bean 的配置	443	15.3.3	加载 Spring	480
14.2.4	Setter 注入	444	15.4	系统实现	482
14.2.5	构造器注入	445	15.4.1	个人中心	482
			15.4.2	好友管理	490
			15.4.3	个性空间	495
			习题答案		503

第1章

构建 Java Web 应用程序

应用程序有两种模式：C/S、B/S。C/S 是客户端/服务器端应用程序，也就是说这种应用程序一般独立运行。而 B/S 就是浏览器端/服务器端应用程序，这种应用程序一般借助 IE 等浏览器来运行。Web 应用程序一般是 B/S 模式。目前，国内外信息化建设已经进入基于 Web 应用为核心的阶段，Java 作为应用于网络的流行语言，其主要用途就是开发 Web 应用程序。本章将介绍 Java Web 的基本知识以及开发环境的相关配置。

本章学习要点：

- 了解 Java Web 应用技术
- 熟练掌握 JDK 的安装和配置
- 了解 Tomcat 服务器的安装
- 熟练掌握 Tomcat 与 MyEclipse 的集成

1.1 初识 Java Web

Java Web 是用 Java 技术来解决相关 Web 互联网领域的技术总和，一个 Web 应用程序包括 Web 客户端和 Web 服务器端两部分。

1. Web 客户端

Web 客户端通常是指用户机上的浏览器，如微软的 IE 浏览器或火狐浏览器等。客户端不需要开发任何用户界面，而统一采用浏览器即可。

2. Web 服务器端

Web 服务器是一台或多台可运行 Web 应用程序的计算机，通常在浏览器中输入的网络地址，即 Web 服务器的地址。当用户在浏览器的地址栏中输入网站地址并按回车键后，请求即被发送到 Web 服务器。服务器接收到请求后，会返回给用户带来请求资源的响应消息。Java 在服务器端的应用非常丰富，如 Servlet、JSP 和第三方框架等。

B/S 中客户端与服务器端采用请求/响应模式进行交互，其工作流程如图 1-1 所示。

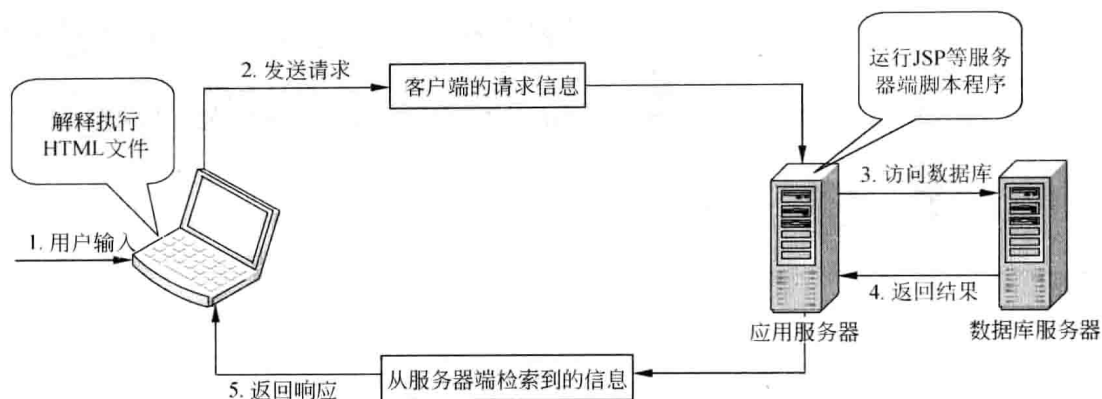


图 1-1 Web 应用程序的工作流程

1.2 Java Web 开发环境概述

在搭建 Java Web 开发环境时,首先需要安装开发工具包 JDK、Web 服务器和数据库等。为了提高开发效率,通常还需要安装 IDE (集成开发环境) 工具。本节将对 Java Web 开发环境进行简单的介绍。

1.2.1 开发工具包 JDK

JDK (Java Development Kit) 是一种用于构建在 Java 平台上发布的应用程序、Applet 和组件的开发环境,即编写 Java 程序必须使用 JDK,它提供了编译 Java 程序和运行 Java 程序的环境。

JDK 是一切 Java 应用程序的基础,所有的 Java 应用程序是构建在 JDK 之上的。JDK 中包括了完整的 JRE (Java Runtime Environment, Java 运行环境),还包括了用于产品环境的各种库类,以及给开发者使用的补充库,如国际化的库、IDL 库。JDK 中还包括各种例子程序,用以展示 Java API 中的各部分。

从初学者角度来看,采用 JDK 开发 Java 程序能够快速理解程序中各部分代码之间的关系,从而有利于理解 Java 面向对象的设计思想。JDK 的另一个显著特点是随着 Java (J2EE、J2SE 以及 J2ME) 版本的升级而升级,但它的缺点也非常明显,从事大规模企业级 Java 应用开发非常困难,不能进行复杂的 Java 软件开发,也不利于团体协同开发。

Java 开发工具包 (JDK) 是 Java 程序员最初使用的开发环境。JDK 由一个标准类库和一组测试及建立文档的 Java 实用程序组成。其核心 Java API 是一些预定义类库,开发人员需要用这些类来访问 Java 语言的功能。Java API 包括一些重要的语言结构以及基本图形、网络和文件 I/O。一般来说,Java API 的非 I/O 部分对于运行 Java 的所有平台是相同的,而 I/O 部分则仅在通用 Java 环境中实现。作为 JDK 实用程序,工具库中有以下 7 种主要的可执行程序:

- ❑ **javac** Java 编译器，将 Java 源代码转换成字节码。
- ❑ **java** Java 解释器，直接从类文件执行 Java 应用程序字节代码。
- ❑ **appletviewer** 小程序浏览器，一种执行 HTML 文件上的 Java 小程序的 Java 浏览器。
- ❑ **javadoc** 根据 Java 源代码及说明语句生成 HTML 文档。
- ❑ **jdb** Java 调试器，可以逐行执行程序、设置断点和检查变量。
- ❑ **javah** 产生可以调用 Java 过程的 C 过程，或建立能被 Java 程序调用的 C 过程的头文件。
- ❑ **javap** Java 反汇编器，显示编译类文件中的可访问功能和数据，同时显示字节代码含义。

1.2.2 Web 服务器

Web 服务器是在网络中为实现信息发布、资料查询、数据处理等诸多应用而搭建基本平台的服务器，只有将开发的 Web 应用放置到其中，才能使网络中的所有用户通过浏览器访问。开发 Web 应用采用的服务器主要是 Servlet 兼容的 Web 服务器，常用的服务器有以下几个：

1. BEA WebLogic 服务器

WebLogic 是 BEA 公司的产品，分为 WebLogic Server、WebLogic Enterprise 和 WebLogic Portal 系列。其中 WebLogic Server 的功能特别强大，支持企业级、多层次和完全分布式的 Web 应用，并且配置简单且界面友好。对于那些正在寻求能够提供 Java 平台所拥有的一切应用服务器的用户来说，WebLogic 是一个十分理想的选择。

2. IBM WebSphere 应用服务器

IBM WebSphere 应用服务器（IBM WebSphere Application Server）是 IBM WebSphere 软件平台的基础和面向服务体系结构的关键构件。该服务器提供了一个丰富的应用程序部署环境，包括用于事务管理、安全性、群集、性能、可用性、连接性和可伸缩性等全套的应用程序服务。它与 Java EE 兼容，并为可与数据库交互并提供动态 Web 内容的 Java 组件、XML 和 Web 服务提供了可移植的 Web 部署平台。

目前 IBM 推出了 WebSphere Application Server V7，该产品是基于 Java EE 5 认证且支持 EJB 3.0 技术的应用程序平台。它提供了安全、可伸缩及高性能的应用程序基础架构，这些基础架构是实现 SOA 所需要的，提高了业务灵活性。

3. Tomcat 服务器

Tomcat 服务器最为流行，它是 Apache-Jakarta 开源项目中的一个子项目，是一个小型、轻量级并支持 JSP 和 Servlet 技术的 Web 服务器。它已经成为学习开发 Java Web 应用的首选。

Web 服务器的工作原理如图 1-2 所示。

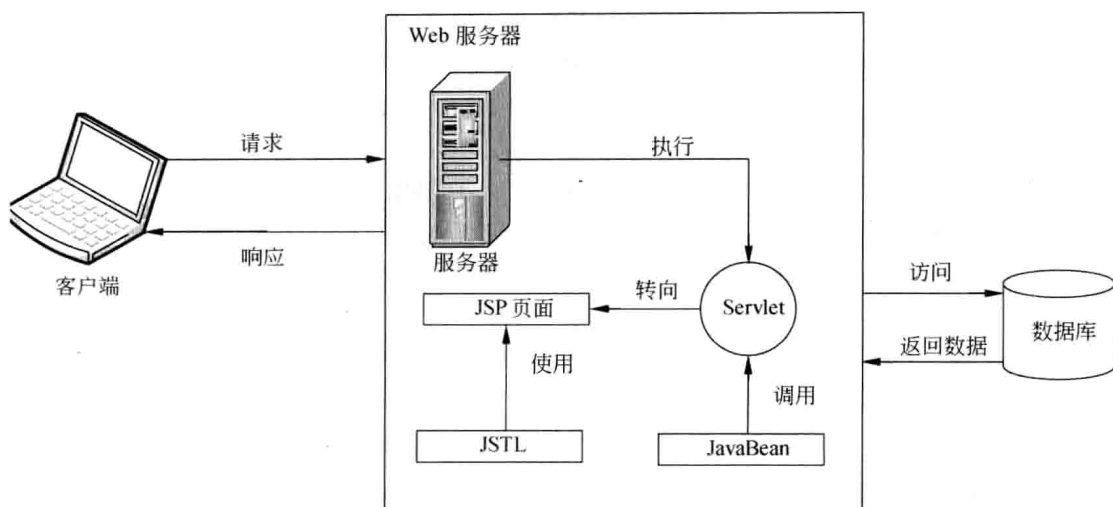


图 1-2 Web 服务器的工作原理

如图 1-2 所示，Web 服务器的工作需要分为以下三步：

- (1) Web 浏览器向一个特定的 Web 服务器发出 Web 页面请求。
- (2) Web 服务器接收到 Web 页面请求后，寻找所请求的 Web 页面，执行相应的功能。
- (3) 将用户请求的最后结果以 Web 页面形式发送到客户的 Web 浏览器。

1.2.3 数据库

开发动态网站时数据库是必不可少的，它主要用来保存网站中需要的信息。根据网站的规模应采用合适的数据库，如大型网站可采用 Oracle，中型网站可采用 Microsoft SQL Server 或 MySQL，小型网站则可以采用 Microsoft Access。Microsoft Access 数据库的功能远不如 Microsoft SQL Server 和 MySQL 强大，但它具有方便和灵活的特点，对于一些小型网站来说是比较理想的选择。

1.2.4 Web 浏览器

浏览器是客户端用户访问 Web 应用的工具，与开发 JSP 应用关系不大。所以开发 JSP 对浏览器的要求并不是很高，任何支持 HTML 的浏览器都可以。目前比较流行的 Web 浏览器是 IE 和火狐浏览器。

1.3 JDK

JDK 是整个 Java 的核心，包括了 Java 运行环境、Java 工具和 Java 基础的类库。在开

发 Java Web 应用之前，首先应该安装 JDK 组件。

1.3.1 JDK 的下载和安装

无论 Java 应用服务器是哪种，实际上都是内置了某个版本的 JDK，因此 Java 程序的运行是离不开 JDK 的。本小节将介绍 JDK 的下载和安装。

1. 获取 JDK 开发工具包

JDK 可以在 Oracle 公司的官方网站 <http://www.oracle.com> 下载，目前最新版本为 JDK 7u5，下载的步骤如下：

(1) 打开 Oracle 公司的官方网站，在首页的栏目中选择 Downloads | Java for Developers 选项，如图 1-3 所示。

(2) 单击 Java for Developers 超链接后，进入 Java SE 的下载页面，如图 1-4 所示。

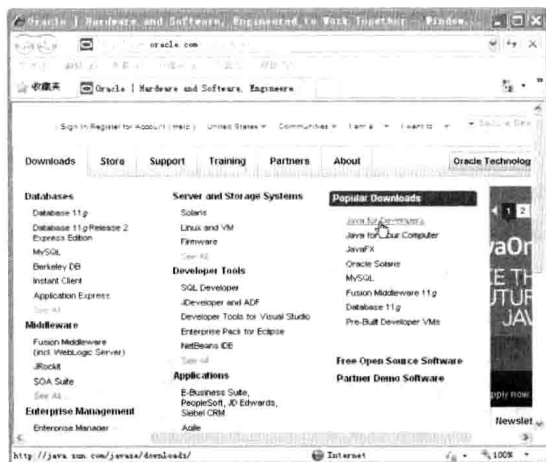


图 1-3 Oracle 官网首页



图 1-4 Java SE 的下载页面

提示

由于 Java 版本不断更新，当读者浏览 Java SE 的下载页面时，显示的是当前最新的版本。

(3) 单击 Java Platform (JDK) 7u5 上方的 DOWNLOAD 按钮，打开 Java SE 的下载列表页面，其中包括 Windows、Solaris 和 Linux 等平台的不同环境 JDK 的下载，如图 1-5 所示。

(4) 在下载之前需要选中 Accept License Agreement 单选按钮，接受许可协议。由于本书中使用的是 32 位版的 Windows 操作系统，因此这里需要选择与平台相对应的 Windows x86 类型的 jdk-7u5-windows-i586.exe 超链接，对 JDK 7u5 进行下载，如图 1-6 所示。