

封超  
晁阳  
编著

# 开发技术详解

- **Java语言基础** 详细讲解Java语言的基础内容和语法知识。
- **开发工具** 为读者推荐JCreator集成开发环境。
- **Web开发基础** 介绍JSP技术、Servlet的编写以及Tomcat 6.0容器的安装、部署和使用。
- **数据库编程** 介绍MySQL数据库的安装和使用、JDBC编程和SQL编程基础。
- **综合案例** 本书最后两章通过有针对性的综合性应用实例，详细地讲解了开发Web应用程序的步骤、方法和技术。

## Tomcat与 Java Web



清华大学出版社

# Tomcat 与 Java Web 开发技术详解

封超晁阳 编著



清华大学出版社

北 京

## 内 容 简 介

利用 Java 语言不仅能开发桌面版应用程序, JavaEE 更是广泛应用于基于 B/S 架构的 Web 企业应用程序领域。尤其是 EJB、JSP、Servlet、JDBC 等技术使得用 Java 语言编写的企业级应用程序更易于维护和扩展。

本书共 19 章, 分为三篇。第一篇为基础篇, 从第 1 章到第 11 章, 详细介绍了 Java 语言的基础语法和理论知识等内容, 为后续两篇的学习打下了坚实的基础; 第二篇为 Web 编程篇, 从第 12 章到第 16 章, 分别讲解了利用 Java 语言开发 Web 应用程序所要用到的各项技术, 包括 JSP 技术、Servlet 技术、JavaBean 技术以及 Java 数据库编程技术(JDBC、SQL 基础和 MySQL 数据库的使用), 还介绍 Tomcat 容器的安装配置和使用方法, 特别是第 15 章, 要求读者认真阅读和理解; 第三篇为实例篇, 从第 17 章到第 19 章, 深入讲解了 MVC 设计模式、在线相册案例和网上商品案例, 此篇为前两个篇章的综合应用篇。

本书通过作者简洁详实的语言和思路清晰的知识结构为读者学习 Java 语言以及如何开发 Web 应用程序做了深入的剖析。本书适合作为初学者的指导书, 也适合有一定 Java 语言基础的读者使用。

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

Tomcat 与 Java Web 开发技术详解/封超, 晁阳编著. —北京: 清华大学出版社, 2008.6  
ISBN 978-7-302-17724-1

I. T… II. ①封… ②晁… III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2008)第 072382 号

责任编辑: 邹 杰 杨作梅

封面设计: 山鹰工作室

版式设计: 杨玉兰

责任印制: 王秀菊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

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 刷 者: 北京四季青印刷厂

装 订 者: 三河市李旗庄少明装订厂

经 销: 全国新华书店

开 本: 190×260 印 张: 30.5 字 数: 722 千字

版 次: 2008 年 6 月第 1 版 印 次: 2008 年 6 月第 1 次印刷

印 数: 1~4000

定 价: 48.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 028376-01

# 前 言

Java 是由 Sun Microsystems 公司于 1995 年 5 月推出的, 由于具有跨平台、面向对象、简单、体系结构中立、分布式、解释型、可靠、安全、健壮、高性能、多线程、动态性和面向网络的编程语言等特性, 得到了许多软件开发商和程序开发人员的支持和好评, 使得它在短短的 10 年中得以快速发展。

2005 年 6 月以来, Java 的各种版本的版本号中都取消了数字 2, J2EE 更名为 Java EE, J2SE 更名为 Java SE, J2ME 更名为 Java ME。

从 1995 年 5 月正式发布 Java 1.0 版本到现在的 JDK6.0, Java 的功能日益完善, 为开发人员提供了越来越强大的类库, 使得开发更简洁, 更容易。

## 本书的特点

本书不仅介绍了 Java 语言的基础知识, 还详细地讲解了如何使用 Java 的高级技术进行 Web 应用程序的开发。本书的前面部分都是按照知识结构分章的, 每个章节中都包括了大量的示例程序, 可以帮助读者更好地理解书中的内容, 各个章节都相互关联。本书的整体知识结构如下。

- Java 语言基础: 详细讲解 Java 语言的基础内容和语法知识。
- 开发工具: 为读者推荐 JCreator 集成开发环境。
- Web 开发基础: 介绍 JSP 技术、Servlet 的编写以及 Tomcat 6.0 容器的安装、部署和使用。
- 数据库编程: 介绍 MySQL 数据库的安装和使用、JDBC 编程和 SQL 编程基础。
- 综合案例: 本书最后两章通过有针对性的综合性应用实例, 详细地讲解了开发 Web 应用程序的步骤、方法和技术。
- 配套光盘: 书中所有示例程序的源代码和案例的应用目录都按照章节顺序和程序名称存放, 读者可以在阅读本书时结合这些示例来巩固所学内容和加深理解。

在综合案例的章节中, 由于篇幅的原因, 作者只列出了主要的功能性代码, 省略了美工部分, 读者可以在配套光盘中找到完整的代码并运行。

## 本书面向的读者

本书主要面向初学者和有一定 Java 基础或者是编程基础的开发人员。

- 对于初学者, 本书从比较低的起点讲解了 Java 语言的基础语法知识, 并且结合大量的例子为读者深入剖析了在学习过程中容易出现错误的地方, 后续章节则深入介绍了 Java 语言的高级知识, 使读者的能力得到进一步提升。
- 对于有一定 Java 基础或者是编程基础(比如具备了 C++语言基础)的开发人员, 阅读本书将会更加容易, 读者将能迅速掌握 Java 语言的开发技巧。

本书由封超、晁阳编著，参编的人员还有席晓燕、崔一辉、贾尚帅、程欣桐、郭盛越、李研、吴超、田涛、薛晓光、李秀荣、郭巧芝、李炜、朱昌建、李隆庚、陈松等，在此一并向他们表示感谢。

由于编者水平有限，书中难免有不足之处，欢迎广大读者批评指正。

本书光盘下载地址为 <http://www.tup.com.cn>，<http://www.wenyuan.com.cn>。

编 者



# 目 录

## 第一篇 基 础 篇

|                                                 |                               |
|-------------------------------------------------|-------------------------------|
| 第 1 章 Java 语言概述.....1                           | 2.1.3 注释、分隔符及转义字符.....31      |
| 1.1 Java 语言的发展与特点.....2                         | 2.2 基本数据类型及其常量值.....32        |
| 1.1.1 Java 语言的发展.....2                          | 2.3 类.....35                  |
| 1.1.2 Java 语言的特点.....2                          | 2.3.1 什么是类.....35             |
| 1.1.3 Java 语言的执行.....5                          | 2.3.2 类的声明和修饰符.....35         |
| 1.2 JDK 的安装和配置.....5                            | 2.3.3 定义方法.....37             |
| 1.2.1 在 Microsoft Windows 平台上<br>安装配置 JDK.....7 | 2.3.4 变量的声明.....41            |
| 1.2.2 在 Linux 平台上<br>安装配置 JDK.....12            | 2.4 对象.....47                 |
| 1.3 Java 程序举例.....14                            | 2.4.1 创建对象.....47             |
| 1.3.1 编写 HelloWorld.....14                      | 2.4.2 this 关键字.....49         |
| 1.3.2 编译 Java 源文件.....15                        | 2.5 包和 main 方法.....50         |
| 1.3.3 运行 Java 程序.....15                         | 2.5.1 包的声明和导入.....50          |
| 1.3.4 在 Linux 平台上编写和<br>运行 HelloWorld.....15    | 2.5.2 程序入口 main()方法的声明.....52 |
| 1.4 Java 语言编码规范.....16                          | 2.6 本章小结.....53               |
| 1.4.1 Java 源文件以及文件组织.....16                     | 第 3 章 运算符与表达式.....55          |
| 1.4.2 缩进排版.....16                               | 3.1 运算符.....56                |
| 1.4.3 注释.....18                                 | 3.1.1 赋值运算符.....56            |
| 1.4.4 声明.....19                                 | 3.1.2 算术运算符.....60            |
| 1.4.5 空白行.....21                                | 3.1.3 比较运算符.....63            |
| 1.4.6 命名规范.....21                               | 3.1.4 移位运算符.....66            |
| 1.5 开发工具——JCreator.....22                       | 3.1.5 等于运算符.....68            |
| 1.5.1 JCreator 简介.....22                        | 3.1.6 位运算符.....70             |
| 1.5.2 安装和使用 JCreator.....23                     | 3.1.7 位反运算符.....71            |
| 1.6 本章小结.....26                                 | 3.1.8 逻辑运算符.....72            |
| 第 2 章 Java 编程基础.....27                          | 3.2 给方法传递参数.....74            |
| 2.1 标识符和关键字.....28                              | 3.2.1 原始数据类型.....74           |
| 2.1.1 标识符.....28                                | 3.2.2 变量数据类型.....75           |
| 2.1.2 Java 中的保留关键字.....28                       | 3.3 本章小结.....76               |
|                                                 | 第 4 章 数组.....77               |
|                                                 | 4.1 数组简介.....78               |
|                                                 | 4.1.1 数组的声明.....79            |

|                                              |            |                                     |            |
|----------------------------------------------|------------|-------------------------------------|------------|
| 4.1.2 数组的创建 .....                            | 80         | 7.2.1 重写规则 .....                    | 128        |
| 4.1.3 初始化数组 .....                            | 82         | 7.2.2 方法的重写 .....                   | 130        |
| 4.2 java.util.Arrays 类 .....                 | 85         | 7.2.3 重载规则 .....                    | 132        |
| 4.3 本章小结 .....                               | 90         | 7.2.4 方法的重载 .....                   | 133        |
| <b>第5章 流程控制和异常处理 .....</b>                   | <b>91</b>  | 7.3 接口的声明和实现 .....                  | 134        |
| 5.1 分支语句 .....                               | 92         | 7.3.1 使用接口实现契约式设计 .....             | 135        |
| 5.1.1 if-else 语句 .....                       | 92         | 7.3.2 定义接口 .....                    | 135        |
| 5.1.2 switch 语句 .....                        | 94         | 7.3.3 实现接口 .....                    | 136        |
| 5.2 循环语句 .....                               | 97         | 7.3.4 接口的多重继承 .....                 | 137        |
| 5.2.1 while 循环语句 .....                       | 97         | 7.4 封装类 .....                       | 138        |
| 5.2.2 do-while 循环语句 .....                    | 99         | 7.4.1 封装类简介 .....                   | 138        |
| 5.2.3 for 循环语句 .....                         | 99         | 7.4.2 创建封装器类对象 .....                | 139        |
| 5.2.4 在循环语句中使用 break 和<br>continue 关键字 ..... | 101        | 7.4.3 封装类的常用方法 .....                | 139        |
| 5.3 标号语句 .....                               | 102        | 7.5 本章小结 .....                      | 141        |
| 5.4 异常处理 .....                               | 104        | <b>第8章 字符串的处理 .....</b>             | <b>143</b> |
| 5.4.1 异常的层次结构 .....                          | 104        | 8.1 String 类及其重要方法 .....            | 144        |
| 5.4.2 try-catch 处理异常 .....                   | 105        | 8.1.1 字符与字符串 .....                  | 144        |
| 5.4.3 使用 finally .....                       | 106        | 8.1.2 String 类 .....                | 144        |
| 5.4.4 异常的匹配 .....                            | 108        | 8.1.3 字符串对象的创建 .....                | 144        |
| 5.4.5 编写并使用自己的异常类 .....                      | 109        | 8.1.4 String 类的主要方法 .....           | 147        |
| 5.4.6 异常的再抛出 .....                           | 110        | 8.2 StringBuffer 类及其方法 .....        | 150        |
| 5.5 本章小结 .....                               | 112        | 8.2.1 StringBuffer 类 .....          | 150        |
| <b>第6章 面向对象程序设计 .....</b>                    | <b>113</b> | 8.2.2 append()方法 .....              | 150        |
| 6.1 封装及其应用 .....                             | 114        | 8.2.3 StringBuffer 类的主要方法 .....     | 151        |
| 6.1.1 封装的特点 .....                            | 114        | 8.3 Pattern 类和 Matcher 类 .....      | 152        |
| 6.1.2 封装的应用 .....                            | 114        | 8.3.1 Pattern 类 .....               | 152        |
| 6.2 继承 .....                                 | 116        | 8.3.2 Matcher 类 .....               | 154        |
| 6.3 多态 .....                                 | 121        | 8.3.3 综合举例 .....                    | 155        |
| 6.4 本章小结 .....                               | 122        | 8.4 Java 数学运算 .....                 | 156        |
| <b>第7章 构造函数、重载和<br/>重写、接口 .....</b>          | <b>123</b> | 8.4.1 java.lang.Math 类 .....        | 156        |
| 7.1 构造函数 .....                               | 124        | 8.4.2 java.math.BigInteger 类 .....  | 159        |
| 7.1.1 构造函数基础 .....                           | 124        | 8.4.3 java.math.BigDecimal 类 .....  | 160        |
| 7.1.2 构造函数的重载 .....                          | 127        | 8.5 本章小结 .....                      | 162        |
| 7.2 重写和重载方法 .....                            | 128        | <b>第9章 集合框架 .....</b>               | <b>163</b> |
|                                              |            | 9.1 重写 hashCode()和 equals()方法 ..... | 164        |
|                                              |            | 9.1.1 Object 类 .....                | 164        |
|                                              |            | 9.1.2 重写 equals()方法 .....           | 165        |

|        |                          |     |
|--------|--------------------------|-----|
| 9.1.3  | 重写 hashCode()方法.....     | 168 |
| 9.2    | 集合框架.....                | 170 |
| 9.2.1  | Java 集合框架简介.....         | 171 |
| 9.2.2  | 集合框架中常用的接口<br>及其实现类..... | 171 |
| 9.3    | Set 接口及其实现类.....         | 172 |
| 9.3.1  | HashSet 类.....           | 172 |
| 9.3.2  | LinkedHashSet 类.....     | 174 |
| 9.3.3  | TreeSet 类.....           | 176 |
| 9.4    | List 接口及其实现类.....        | 177 |
| 9.4.1  | LinkedList 类.....        | 177 |
| 9.4.2  | ArrayList 类.....         | 178 |
| 9.4.3  | Vector 类.....            | 179 |
| 9.5    | Map 接口及其实现类.....         | 181 |
| 9.5.1  | Hashtable 类.....         | 182 |
| 9.5.2  | LinkedHashMap 类.....     | 183 |
| 9.5.3  | HashMap 类.....           | 184 |
| 9.5.4  | TreeMap 类.....           | 185 |
| 9.6    | 本章小结.....                | 187 |
| 第 10 章 | 内部类.....                 | 189 |

|        |                 |     |
|--------|-----------------|-----|
| 10.1   | 内部类简介.....      | 190 |
| 10.1.1 | 编写普通的内部类.....   | 190 |
| 10.1.2 | 内部类的实例化与访问..... | 191 |
| 10.2   | 局部内部类.....      | 192 |
| 10.3   | 匿名内部类.....      | 197 |
| 10.4   | 顶级嵌套类.....      | 201 |
| 10.5   | 本章小结.....       | 204 |
| 第 11 章 | 多线程.....        | 207 |
| 11.1   | 多线程基础.....      | 208 |
| 11.1.1 | 线程的定义.....      | 208 |
| 11.1.2 | 实例化线程.....      | 210 |
| 11.1.3 | 启动线程.....       | 212 |
| 11.2   | 线程的阻塞和优先级.....  | 216 |
| 11.2.1 | 线程睡眠.....       | 216 |
| 11.2.2 | 线程的优先级.....     | 218 |
| 11.2.3 | 线程的让步.....      | 220 |
| 11.3   | 同步.....         | 223 |
| 11.3.1 | 同步与锁定.....      | 223 |
| 11.3.2 | 死锁.....         | 226 |
| 11.4   | 本章小结.....       | 227 |

## 第二篇 Web 编程篇

|        |                                                      |     |
|--------|------------------------------------------------------|-----|
| 第 12 章 | Tomcat 服务器简介.....                                    | 229 |
| 12.1   | Tomcat 简介.....                                       | 230 |
| 12.2   | 下载 Tomcat.....                                       | 230 |
| 12.3   | 安装 Tomcat.....                                       | 231 |
| 12.3.1 | Tomcat 在 Windows 平台的<br>安装.....                      | 231 |
| 12.3.2 | Tomcat 在 Linux 平台的<br>安装.....                        | 233 |
| 12.3.3 | 测试 JSP.....                                          | 234 |
| 12.3.4 | 测试 Servlet.....                                      | 236 |
| 12.4   | Tomcat 6.0 的目录结构.....                                | 237 |
| 12.5   | 配置 Tomcat 5.5 Administration Web<br>Application..... | 239 |
| 12.6   | Java 开源 Web 框架简介.....                                | 240 |
| 12.6.1 | WebWork.....                                         | 240 |

|        |                             |     |
|--------|-----------------------------|-----|
| 12.6.2 | Struts.....                 | 241 |
| 12.6.3 | Spring.....                 | 241 |
| 12.6.4 | Java Server Faces(JSF)..... | 241 |
| 12.6.5 | Tapestry.....               | 241 |
| 12.7   | 本章小结.....                   | 241 |
| 第 13 章 | JSP 编程.....                 | 243 |
| 13.1   | JSP 概述.....                 | 244 |
| 13.1.1 | JSP 简介.....                 | 244 |
| 13.1.2 | 一个简单的 JSP 程序.....           | 244 |
| 13.2   | JSP 基本语法.....               | 245 |
| 13.2.1 | JSP 语法.....                 | 245 |
| 13.2.2 | 操作 Cookie.....              | 254 |
| 13.3   | JSP 内建对象.....               | 255 |
| 13.3.1 | request 对象.....             | 256 |
| 13.3.2 | response 对象.....            | 258 |

|        |                              |     |        |                                        |     |
|--------|------------------------------|-----|--------|----------------------------------------|-----|
| 13.3.3 | out 对象.....                  | 259 | 15.2.4 | 数据定义语言(DDL).....                       | 300 |
| 13.3.4 | session 对象.....              | 260 | 15.2.5 | 数据操纵语言(DML).....                       | 301 |
| 13.3.5 | application 对象 .....         | 264 | 15.3   | 通过 JDBC 访问 MySQL 数据库.....              | 301 |
| 13.3.6 | config 对象.....               | 266 | 15.3.1 | JDBC 简介 .....                          | 301 |
| 13.3.7 | pageContext 对象.....          | 266 | 15.3.2 | 获取 MySQL 数据库<br>支持 JAR 包 .....         | 302 |
| 13.3.8 | page 对象 .....                | 267 | 15.3.3 | JDBC 访问 MySQL<br>数据库 .....             | 303 |
| 13.3.9 | exception 对象.....            | 267 | 15.4   | 数据源(DataSource)简介 .....                | 306 |
| 13.4   | 本章小结 .....                   | 269 | 15.5   | 在 Tomcat 6.0 中配置 MySQL<br>数据源连接池 ..... | 306 |
| 第 14 章 | Servlet 编程 .....             | 271 | 15.6   | 在 Java 程序中访问数据源 .....                  | 307 |
| 14.1   | Servlet 简介 .....             | 272 | 15.6.1 | JNDI 基础.....                           | 307 |
| 14.1.1 | Servlet 的主要功能和特点 ...         | 272 | 15.6.2 | Java 程序访问数据源 .....                     | 307 |
| 14.1.2 | Servlet 的基本结构 .....          | 272 | 15.7   | 处理 MySQL 中文编码 .....                    | 309 |
| 14.2   | Servlet 的生命周期 .....          | 274 | 15.8   | SQLJ 编程 .....                          | 311 |
| 14.3   | Servlet 编程类和接口 .....         | 277 | 15.8.1 | SQLJ 简介 .....                          | 311 |
| 14.3.1 | HttpSession 接口 .....         | 278 | 15.8.2 | SQLJ 与 JDBC .....                      | 311 |
| 14.3.2 | HttpServletRequest 接口 .....  | 278 | 15.8.3 | 使用 SQLJ 访问 Oracle<br>数据库 .....         | 312 |
| 14.3.3 | HttpServletResponse 接口 ..... | 278 | 15.9   | 本章小结 .....                             | 314 |
| 14.4   | JSP 与 Servlet.....           | 283 | 第 16 章 | JavaBean 编程 .....                      | 315 |
| 14.5   | 本章小结 .....                   | 285 | 16.1   | JavaBean 简介 .....                      | 316 |
| 第 15 章 | Java 数据库编程 .....             | 287 | 16.2   | JavaBean 规范 .....                      | 316 |
| 15.1   | 安装和配置 MySQL 数据库 .....        | 288 | 16.2.1 | 如何编写 JavaBean .....                    | 316 |
| 15.1.1 | MySQL 数据库简介 .....            | 288 | 16.2.2 | 在 JSP 中使用 JavaBean .....               | 317 |
| 15.1.2 | 下载 MySQL 数据库 .....           | 289 | 16.2.3 | JavaBean 与 EJB .....                   | 320 |
| 15.1.3 | 安装和配置 MySQL .....            | 290 | 16.3   | 本章小结 .....                             | 320 |
| 15.2   | 数据库基础——SQL 编程 .....          | 299 |        |                                        |     |
| 15.2.1 | SQL 简介 .....                 | 299 |        |                                        |     |
| 15.2.2 | 数据查询语言 .....                 | 299 |        |                                        |     |
| 15.2.3 | 数据控制语言(DCL).....             | 300 |        |                                        |     |

### 第三篇 实 例 篇

|        |                        |     |        |                                             |     |
|--------|------------------------|-----|--------|---------------------------------------------|-----|
| 第 17 章 | MVC 设计模式.....          | 321 | 17.2.4 | 使用 SmartUpload 组件<br>下载文件 .....             | 330 |
| 17.1   | MVC 设计模式简介 .....       | 322 | 17.2.5 | 在 Tomcat 中部署并运行<br>应用程序 .....               | 331 |
| 17.2   | SmartUpload 组件 .....   | 324 | 17.3   | JSP+Servlet+JavaBean 综合案例——<br>用户登录验证 ..... | 335 |
| 17.2.1 | SmartUpload 组件基础 ..... | 324 | 17.3.1 | 数据库实现 .....                                 | 335 |
| 17.2.2 | 编写前台 JSP .....         | 327 |        |                                             |     |
| 17.2.3 | 编写中心控制器 Servlet.....   | 328 |        |                                             |     |

|                             |            |                             |            |
|-----------------------------|------------|-----------------------------|------------|
| 17.3.2 开发持久层 JavaBean.....  | 336        | 18.6.4 处理用户订单 .....         | 428        |
| 17.3.3 开发中心控制器 Servlet..... | 337        | 18.7 在 Tomcat 中部署并运行        |            |
| 17.3.4 开发显示层 JSP .....      | 338        | 应用程序 .....                  | 434        |
| 17.3.5 在 Tomcat 中部署并运行      |            | 18.7.1 web.xml 文件的配置.....   | 434        |
| 应用程序 .....                  | 340        | 18.7.2 应用程序的目录结构设计....      | 436        |
| 17.4 本章小结 .....             | 342        | 18.7.3 运行网上商店应用程序.....      | 436        |
| <b>第 18 章 网上商店综合案例.....</b> | <b>343</b> | 18.8 本章小结 .....             | 437        |
| 18.1 系统分析 .....             | 344        | <b>第 19 章 在线相册综合案例.....</b> | <b>439</b> |
| 18.1.1 概述 .....             | 344        | 19.1 系统分析 .....             | 440        |
| 18.1.2 需求分析 .....           | 344        | 19.1.1 概述 .....             | 440        |
| 18.2 系统架构设计 .....           | 344        | 19.1.2 需求分析 .....           | 440        |
| 18.2.1 前台架构分析 .....         | 344        | 19.2 系统架构设计 .....           | 440        |
| 18.2.2 后台架构分析 .....         | 345        | 19.2.1 系统架构分析 .....         | 440        |
| 18.3 数据库设计 .....            | 346        | 19.2.2 功能分析说明 .....         | 440        |
| 18.3.1 MySQL 数据类型参考 .....   | 346        | 19.3 数据库设计 .....            | 441        |
| 18.3.2 数据库建表 .....          | 347        | 19.3.1 MySQL 存取图片 .....     | 441        |
| 18.3.3 数据库表关系图 .....        | 353        | 19.3.2 数据库建表 .....          | 442        |
| 18.4 开发数据库 JavaBean.....    | 353        | 19.4 开发数据库 JavaBean.....    | 444        |
| 18.5 系统前台主要功能模块的            |            | 19.5 系统主要功能模块的设计与实现....     | 456        |
| 设计与实现 .....                 | 377        | 19.5.1 用户身份验证模块 .....       | 457        |
| 18.5.1 用户注册模块 .....         | 379        | 19.5.2 照片显示模块 .....         | 463        |
| 18.5.2 用户登录模块 .....         | 382        | 19.5.3 用户上传照片模块 .....       | 466        |
| 18.5.3 商品搜索模块 .....         | 386        | 19.5.4 新建相册模块 .....         | 468        |
| 18.5.4 用户管理模块 .....         | 393        | 19.5.5 删除相册模块 .....         | 470        |
| 18.5.5 用户资料修改模块 .....       | 406        | 19.6 在 Tomcat 中部署并运行        |            |
| 18.5.6 购物车模块 .....          | 413        | 应用程序 .....                  | 473        |
| 18.6 系统后台主要功能的              |            | 19.6.1 web.xml 文件和          |            |
| 模块设计与实现 .....               | 416        | 数据源的配置 .....                | 473        |
| 18.6.1 商品搜索模块 .....         | 417        | 19.6.2 应用程序的目录结构设计....      | 474        |
| 18.6.2 商品管理模块 .....         | 422        | 19.6.3 运行在线相册应用程序.....      | 474        |
| 18.6.3 商品图片更新 .....         | 427        | 19.7 本章小结 .....             | 475        |

# 第一篇 基础篇

## 第1章

### Java 语言概述

本章主要介绍 Java 语言的发展、特点, JDK 的下载和安装以及 Java 语言编码规范等。Java 是由 Sun Microsystems 公司于 1995 年 5 月推出的,其中包括:Java 程序设计语言(以下简称 Java)和 Java 运行平台(包括 Java 虚拟机和 Java 应用编程接口(Java Application Programming Interface, API))。Java 语言相对于 C、C++等编程语言具有更好的安全性,Java 是跨平台的、面向对象的、简单的、面向网络的编程语言,这也是 Java 得以快速发展的根本所在。

网络技术给现代企业带来了许多新的机遇和挑战,如何改善客户关系服务、全球化和信息检索等成为现代企业面对的主要难题,这些在技术上反映为信息的获取、系统管理、系统集成、新技术的开发、Intranet 等与商业的结合。Java 语言的出现迅速引起所有程序员和软件公司的极大关注,他们纷纷尝试用 Java 语言编写网络应用程序,并利用网络把程序发布到世界各地进行运行。包括 IBM、Oracle、微软、SGI 等公司纷纷与 Sun Microsystems 公司签订合同,被授权使用 Java 平台技术。目前,Java 语言已经成为最流行的网络编程语言,截止到 2001 年,全世界大约有 310 万 Java 程序员,许多高等院校纷纷开设 Java 课程,Java 正逐步成为世界上使用人数最多的编程语言。

微软公司总裁比尔·盖茨先生曾经说过:“Java 语言是长时间以来最卓越的程序设计语言”。

#### 内容提要

- Java 语言的发展与特点
- JDK 的安装与配置
- HelloWorld

## 1.1 Java 语言的发展与特点

任何事物的产生都既有必然的原因也有偶然的因素,Java 语言的出现就很好地验证了这一观点。

### 1.1.1 Java 语言的发展

1991 年 4 月, Sun 公司的 James Gosling 领导的绿色计划(Green Project)开始着力开发一种分布式系统结构,使其能够在各种消费性电子产品上运行,包括 PDA、机顶盒、手机等。为了使整个系统与平台无关, Gosling 首先从改写 C 编译器着手。但是 Gosling 在改写过程中感到只使用 C 编译器是无法满足需要的,于是在 1991 年 6 月开始准备开发一个新的语言,那么给它起一个什么名字呢? Gosling 回首向窗外望去,看见一棵老橡树,于是将这种新的程序设计语言命名为 Oak,这就是 Java 语言的前身,后来发现 Oak 已是另一个语言的注册商标,于是改名为 Java,即太平洋上一个盛产咖啡的岛屿的名字,一杯飘香的咖啡因此成为了 Java 语言的标志。

Java 语言的发展得益于万维网(WWW)的发展。刚开始时,万维网的发展比较缓慢,每个网页都是静态的画面,不能与用户进行交互。即使是后来的 CGI(计算机图形接口)也只是在服务器端运行,速度太慢,人们迫切需要能够在浏览器端与用户进行交互,并且画面能够动起来的网页,但是,万维网上的计算机结构各不相同,操作系统也千差万别,由于 Java 语言具有的平台无关性适应了网络发展的这种需要,因此 Java 语言在网络上流行起来,同时也推动了网络的发展。

### 1.1.2 Java 语言的特点

Java 是一种跨平台,适合于分布式计算环境的面向对象的编程语言。具体来说,它具有如下特性:简单性、面向对象、体系结构中立、分布式、解释型、可靠、安全、平台无关、可移植、健壮、高性能、多线程、动态性等。

下面将重点介绍 Java 语言的面向对象、平台无关、分布式、多线程、可靠和安全、体系结构中立等特性。

#### 1. 面向对象

面向对象是现实世界模型的自然延伸,现实世界中任何实体都可以看作是对象,对象之间通过消息相互作用。另外,现实世界中任何实体都可归属于某类事物,任何对象都是某一类事物的实例。

所有面向对象编程语言都支持封装性、多态性、抽象性和继承性,Java 也不例外。现实世界中的对象都有属性和行为,映射到计算机程序上,属性则表示对象的数据,行为表示对象的方法(其作用是处理数据或同外界交互)。所谓封装,就是用一个自主式框架把对象的数据和方法连在一起形成一个整体。可以说,对象是支持封装的手段,是封装的基本单位。Java 语言的封装性较强,因为 Java 无全程变量,在 Java 中绝大部分成员是对象,只有简单的数字类型、字符类型和布尔类型除外。而对于这些类型,Java 也提供了相应的对象类型以便与其他对象交

互操作。

多态性是指使用同样的方法名因为调用的方式不同而执行完全不同的例程的能力,具体来说,可以用“一个对外接口,多个内在实现方法”表示。例如,运算符重载(operator over load)就是一种优秀的多态机制体现。

继承是指一个对象直接使用另一个对象的属性和方法。事实上,我们遇到的很多实体都有继承的含义。例如,若把汽车看成一个实体,它可以分成多个子实体,如卡车、公共汽车等。这些子实体都具有汽车的特性,因此,汽车是它们的父类,而这些子实体则是汽车的子类。Java 提供给用户一系列类(Class),Java 的类有层次结构,子类可以继承父类的属性和方法。与另外一些面向对象编程语言不同,Java 只支持单一继承。但是,Java 支持一个接口继承多个接口。

## 2. 平台无关性

Java 是平台无关的语言,是指用 Java 写的应用程序不用修改就可可在不同的软硬件平台上运行。平台无关分为两种:源代码级和目标代码级。C 和 C++ 语言具有一定程度的源代码级平台无关,表明用 C 或 C++ 语言写的应用程序不用修改只需重新编译就可以在不同平台上运行。

Java 主要靠 Java 虚拟机(Java Virtual Machine, JVM)在目标代码级实现平台无关性。JVM 是一种抽象机器,它附着在具体操作系统之上,本身具有一套虚机器指令,并有自己的栈、寄存器组等。但 JVM 通常是在软件上而不是在硬件上实现的。目前, Sun Microsystems 公司已经设计实现了 Java 芯片,主要使用在网络计算机(Network Computer, NC)上。

另外,Java 芯片的出现也会使 Java 更容易嵌入到家用电器中。JVM 是 Java 平台无关的基础,在 JVM 上,有一个 Java 解释器用来解释 Java 编译器编译后的程序。Java 编程人员在编写完软件后,通过 Java 编译器将 Java 源程序编译为 JVM 的字节代码。任何一台机器只要配备了 Java 解释器,就可以运行这个程序,而不管这种字节码是在何种平台上生成的。另外,Java 采用的是基于 IEEE 标准的数据类型。通过 JVM 可以保证数据类型的一致性,也确保了 Java 的平台无关性。

Java 解释器,每当需要使用某种应用软件时,下载该软件的字节代码即可,运行结果也可以发回服务器。目前,已有数家公司开始使用这种新型的计算模式构筑自己的企业信息系统。

## 3. 分布式

分布式包括数据分布和操作分布两种。数据分布是指数据可以分散在网络的不同主机上,操作分布是指把一个计算分散在不同主机上处理。

Java 支持 WWW 客户机/服务器计算模式,因此,它支持这两种分布方式。对于前者,Java 提供了一个叫做 URL 的对象(java.net.URL 类),利用这个对象,可以打开并访问具有相同 URL 地址的对象,访问方式与访问本地文件系统相同。对于后者,Java 的 applet 小程序可以从服务器下载到客户端,即部分计算在客户端进行,从而可以提高系统执行效率。

Java 提供了一整套网络类库,开发人员可以利用类库进行网络程序设计,方便地实现 Java 的分布式特性。

## 4. 可靠性和安全性

Java 最初的设计目的是应用于电子类消费产品,因此要求较高的可靠性。Java 虽然源于

C++语言,但它消除了C++语言中许多不可靠因素,可以防止许多编程错误。首先,Java是强类型的语言,要求显式的方法声明,这保证了编译器可以发现方法调用错误,保证程序更加可靠;其次,Java不支持指针,这杜绝了内存的非法访问;再次,Java的自动单元收集防止了内存丢失等动态内存分配导致的问题;第四,Java解释器运行时实施检查,可以发现数组和字符串访问的越界;最后,Java提供了异常处理机制,程序员可以把一组错误代码放在一个地方,这样可以简化错误处理任务以便于恢复。

由于Java主要用于网络应用程序开发,因此对安全性有较高的要求。如果没有安全保证,用户从网络下载程序执行就非常危险。Java通过自己的安全机制防止了病毒程序的产生和下载程序对本地系统的威胁破坏。当Java字节码进入解释器时,首先必须经过字节码校验器的检查,然后,Java解释器将决定程序中类的内存布局,随后,类装载器负责把来自网络的类装载到单独的内存区域,避免应用程序之间相互干扰破坏。最后,客户端用户还可以限制从网络上装载的类只能访问某些文件系统。

## 5. 多线程

线程是操作系统的一种新概念,它又被称作轻量进程,是比传统进程更小的可并发执行的单位。

C和C++语言采用单线程体系结构,而Java却提供了多线程支持。Java在两方面支持多线程。一方面,Java环境本身就是多线程的。若干个系统线程运行负责必要的无用单元回收,系统维护等系统级操作;另一方面,Java语言内置多线程控制,可以大大简化多线程应用程序开发。Java提供了一个类Thread,由它负责启动运行,终止线程,并可检查线程状态。Java的线程还包括一组同步原语,这些原语负责对线程实行并发控制。利用Java的多线程编程接口,开发人员可以方便地写出支持多线程的应用程序,提高程序执行效率。必须注意的一点是,Java的多线程支持在一定程度上受运行时支持平台的限制。例如,如果操作系统本身不支持多线程,Java的多线程特性可能就表现不出来。

## 6. 体系结构中立

Java语言是体系结构中立的。Java程序(后缀为.java的文件)在Java平台上被编译为体系结构中立的字节码格式(后缀为.class的文件),然后可以在实现这个Java平台的任何系统中运行。这种途径适合于异构的网络环境和软件的分发。

另外,Java使用Unicode作为它的标准字符,这项特性使得Java的程序在不同语言的平台上都能撰写和执行。简单地说,可以把程序中的变量、类别名称使用中文来表示,当程序移植到其他语言平台时,还是可以正常地执行。Java也是目前所有计算机语言当中,唯一一直使用Unicode的语言。

Java企业应用编程接口(Java Enterprise APIs)为企业计算及电子商务应用系统提供了有关技术和丰富的类库供开发人员学习和使用。具体如下:

- JDBC(Java Database Connectivity)——它为Java应用程序连接各种关系数据库提供了统一的编程接口,通过各个数据库厂商提供的驱动程序就可以不用修改应用程序源代码就能实现对各种数据库的操作。
- EJB(Enterprise JavaBeans)——Java基于组件的企业JavaBean(当前最新版本为EJB 3.0)

- JavaRMI(Java Remote Method Invocation)——分布式 Java 应用程序, Java 远程方法调用, 它是 JavaEE 的基石。
- JavaIDL(Java Interface Definition Language)——提供 Java 与 CORBA(Common Object Request Broker Architecture, 通用对象请求代理体系结构)的集成。
- JNDI(Java Naming and Directory Interface)——Java 命名和目录服务接口, 它是 JavaEE 的基石。
- JMS(Java Message Service)——Java 消息服务(如消息队列、发布/订阅等有关技术), 通过异步调用实现, 提高了应用程序系统的响应性能。
- JTS(Java Transaction Service)——Java 事务服务(存取事务处理资源的标准, 包括事务处理应用程序、事务处理管理及监控)。

### 1.1.3 Java 语言的执行

Java 语言和 C 语言等有所不同, 编译后的文件为.class 字节码文件, 它由虚拟机解释执行。Java 语言的一个非常重要的特点就是与平台无关(这里的平台是指操作系统平台)。而 Java 虚拟机(Java Virtual Machine, JVM)是实现 Java 跨平台的关键。Java 语言编译执行的过程, 如图 1-1 所示。

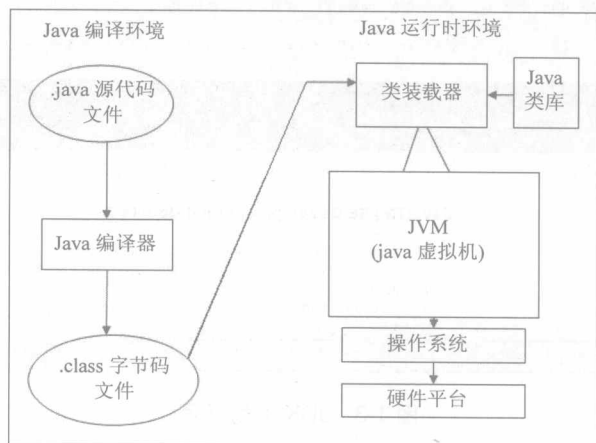


图 1-1 Java 语言编译执行过程图

Java 虚拟机由 5 个部分组成：一组指令集、一组寄存器、一个栈、一个无用单元收集堆 (Garbage-collected-heap) 和一个方法区域。这五部分是 Java 虚拟机的逻辑成分, 不依赖任何实现技术或组织方式, 但它们的功能必须在真实机器上以某种方式实现。

## 1.2 JDK 的安装和配置

JDK(Java Development Kit), 即 Java 开发工具包, 是整个 Java 的核心, 它包括了 Java 的运行环境(Java Runtime Environment)、一些 Java 工具和 Java 基础的类库(rt.jar)。不论什么 Java 应用服务器, 实质都是内置了某个版本的 JDK。因此安装 JDK 是学习 Java 的第一步。最主流

的 JDK 是 Sun 公司发布的 JDK, 可以从 <http://java.sun.com/javase/downloads/index.jsp> 下载最新版本的 JDK, 如图 1-2 所示。本书将使用 JDK 6.0。

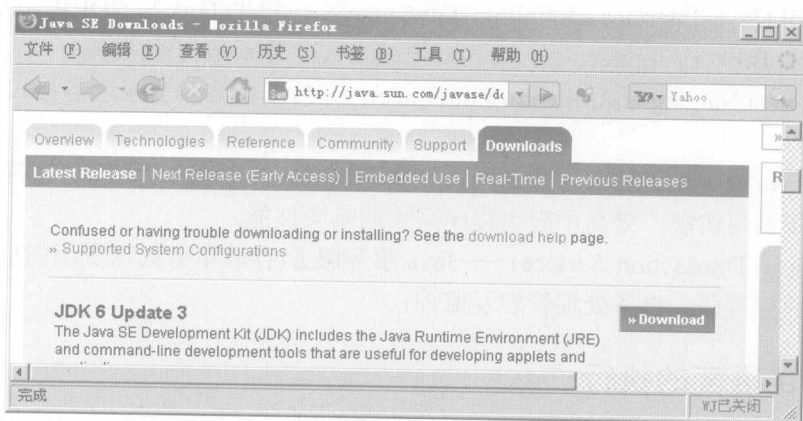


图 1-2 JDK 下载首页

单击图 1-2 所示页面中的 Download 按钮进入 JDK 下载页面, 如图 1-3 所示。

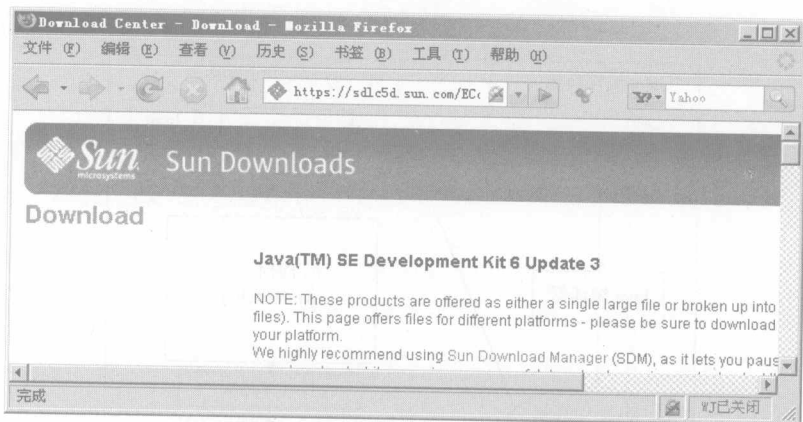


图 1-3 JDK 下载页面

如图 1-4 所示, 选中 Accept License Agreement | Review License Agreement 单选按钮。只有在选择了此项之后才能正常下载 JDK。

Required: You must accept the license agreement to download the product.  
☒ Accept License Agreement | Review License Agreement  
☐ Decline License Agreement

图 1-4 要求选择接受注册协议的界面

图 1-5 所示为 Microsoft Windows 平台的 JDK 下载项, 这里选择 Windows Offline Installation, Multi-language 下载。其他操作系统平台的 JDK 下载与此类似, 例如 Linux 平台, 如图 1-6 所示, 只要选择其中任何一个下载即可, 在这里使用 Linux RPM in self-extracting file。

| Windows Platform - Java(TM) SE Development Kit 6 Update 3                                                             |                                                                                                                                |                                 |           |
|-----------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|---------------------------------|-----------|
| <input checked="" type="checkbox"/>  |                                                                                                                                |                                 |           |
| Download the full version as a <b>single file</b> .                                                                   |                                                                                                                                |                                 |           |
| <input type="checkbox"/>                                                                                              |  Windows Offline Installation, Multi-language | jdk-6u3-windows-i586-p.exe      | 65.64 MB  |
| <input type="checkbox"/>                                                                                              |  Windows Online Installation, Multi-language  | jdk-6u3-windows-i586-p-iftw.exe | 373.39 KB |

图 1-5 Windows 平台的下载项的选择界面

| Linux Platform - Java(TM) SE Development Kit 6 Update 3 |                                   |                            |          |
|---------------------------------------------------------|-----------------------------------|----------------------------|----------|
| <input checked="" type="checkbox"/>                     |                                   |                            |          |
| <input type="checkbox"/>                                | Linux RPM in self-extracting file | jdk-6u3-linux-i586-rpm.bin | 61.64 MB |
| <input type="checkbox"/>                                | Linux self-extracting file        | jdk-6u3-linux-i586.bin     | 65.40 MB |

图 1-6 Linux 平台的下载项的选择界面

除 Sun 之外,还有很多公司和组织都开发了自己的 JDK,例如 IBM 公司开发的 JDK, BEA 公司开发的 Jrocket 等。JDK 是一个编写 Java 的 applet 和应用程序的程序开发环境。它由一个处于操作系统层之上的运行环境以及开发者编译、调试和运行用 Java 语言编写的 applet 和应用程序所需的工具组成。

下面开始介绍 JDK 在 Microsoft Windows 平台和 Linux(本书以 Fedora Core 6 为例,以下简称 FC6)平台上的安装配置。主要是以图解的形式进行介绍。

### 1.2.1 在 Microsoft Windows 平台上安装配置 JDK

在 Microsoft Windows 平台上安装应用软件相对简单,JDK 的安装也是一样。首先,用鼠标左键双击下载的 JDK 安装文件 jdk-6u3-windows-i586-p.exe,出现如图 1-7 所示的界面。

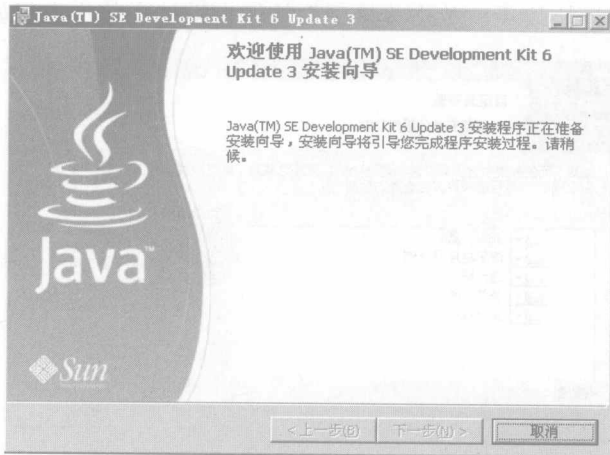


图 1-7 JDK 的安装界面(第一步)

JDK 成功安装以后,可以在 JDK 的指定安装目录中看到一些子目录,下面是对这些子目录的一些解释:

- bin 目录是开发 Java 程序需要用到的工具。