

普通高等教育“十二五”规划教材

Technology  
实用技术

# Java

## 程序设计 实用教程

钱慎一 编著

- 内容全面、结构合理、由浅入深、循序渐进、全方位满足教学要求
- 案例分析+详细讲解，浅显易懂、形象生动、易教易学
- 所有实例均经过调试运行通过，可直接参照使用，方便自学

PPT电子教案

超值程序源代码

科学出版社

普通高等教育“十二五”规划教材

# Java 程序设计实用教程

钱慎一 编著

科学出版社

北京

## 内 容 简 介

Java 是 Sun Microsystems 公司推出的一种程序设计语言,以其面向对象和跨平台的特性风靡全球,是目前软件设计中极为流行的编程语言。Java语言不仅可以用来开发大型的应用程序,而且特别适合基于网络的应用开发,Java 已经成为网络时代最重要的编程语言。

本书通俗易懂,便于读者自学。对于一些较为难以理解的概念,采用循序渐进、逐步展开的方式进行介绍;对于不容易理解的问题,采用大量简单的示例进行说明,书中的这些示例都是作者精心设计的,具有很强的针对性。

全书共 13 章,分别介绍了 Java 的运行开发环境、基本数据类型、控制语句、类、对象、接口、常用类、异常处理、Swing 组件、多线程、输入/输出流、网络编程和数据库连接技术等内容。最后,通过一个实例对书中各章节的知识点进行了综合应用。

本书不仅可以作为高等院校本、专科计算机软件、信息系统、电子商务等相关专业的教材,同时也适合自学者及软件开发人员参考使用。

### 图书在版编目(CIP)数据

Java 程序设计实用教程/钱慎一编著. —北京:科学出版社,2011  
ISBN 978-7-03-031972-2

I. J… II. 钱… III. Java 语言-程序设计-高等学校-教材 IV. TP312  
中国版本图书馆 CIP 数据核字(2011)第 155545 号

责任编辑:张莉莉 杨 凯 / 责任制作:董立颖 魏 谨

责任印制:赵德静 / 封面设计:YOLEN'S

北京东方科龙图文有限公司 制作

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

**科 学 出 版 社** 出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

北京中科印刷有限公司 印刷

科学出版社发行 各地新华书店经销

\*

2011 年 9 月第 一 版 开本:787×1092 1/16

2011 年 9 月第一次印刷 印张:25 1/2

印数:1—4 000 字数:560 000

定 价:49.00 元

(如有印装质量问题,我社负责调换)

# 前 言

Java 是 Sun Microsystems 公司推出的一种程序设计语言,以其面向对象和跨平台的特性风靡全球,它是目前国内外使用最为广泛的程序设计语言。可以说,Java 技术已经成为当今全世界最流行的开发工具和主流技术。

现在市面上有关 Java 的书很多,但适合课堂教学的书却不多,为此本书作者结合自己多年的教学经验和工程实践经验,力图使本书成为适合课堂教学的教材。

本书具有以下几个特点:

① 循序渐进,对新概念的引入和讲解循序渐进,逐步展开,确保读者能够更加容易理解和掌握这些新的概念。

② 浅显易懂,通过大量简单易懂的实例使读者能够在较短的时间掌握较多和较为复杂的知识。

③ 系统性、整体性较强,通过一个综合实例实现了贯穿全书各个知识点的目标。

④ 特别适合读者动手练习,所有实例程序都在 Java SE 6.0 环境下调试运行通过,读者可以直接参照使用。

全书共分 13 章,其内容如下所述:

第 1 章简单介绍 Java 语言的来历和特点以及 Java 的运行和开发环境。

第 2 章重点介绍 Java 语言的基础知识。

第 3~4 章涉及类、继承、多态等概念,对许多重要但不易理解的概念,例如,继承、多态、接口等,都结合实例给予了全面的介绍。

第 5~6 章主要讨论了常用的类和接口及异常处理。

第 7~8 章介绍如何利用 Java 的图形组件创建用户界面。

第 9 章对输入、输出流进行较为详尽的介绍。

第 10 章通过针对性的例子介绍多线程这一重要概念。

第 11 章主要介绍 Java 在网络编程方面的应用。

第 12 章首先介绍数据库相关的基础知识,进而讲述如何在 Java 程序中连接数据库、并实现数据的存取。

第 13 章介绍一个 C/S 结构的图书信息查询系统,该系统综合运用了本书各章节的知识和技术,包括对象的基本操作、继承机制、接口、图形用户界面、线程技术、异常处理、网络套接字编程技术、数据库编程技术等。这既是 Java 综合应用的举例,也是实用性的体现。这将进一步提高学生学习 Java 的兴趣。

本书内容翔实,示例丰富,结构合理,语言简洁,图文并茂。作为一本教程,本书面向初中级用户。

本书由郑州轻工业学院钱慎一老师编著,参加本书编写的还有金保华、朱付保、金松河、王捷、黄永丽、朱会东、宋宝卫、张阳、张旭等老师,尼春雨、王国胜、徐明华、刘松云、张丽、傅晓锋、蒋军军等也参与了本书的校对与代码调试工作,对他们的工作表示衷心的感谢。

## 前 言

谢。另外, 特别感谢郑州轻工业学院教务处对本书的大力支持。

由于编写时间仓促, 加之作者水平有限, 书中难免会有错误和疏漏之处, 恳请广大读者给予批评指正。

编 者

2011年5月



# 目 录

## 第 1 章 Java 基础知识

|                          |    |
|--------------------------|----|
| 1.1 Java 语言简介及特点 .....   | 1  |
| 1.1.1 Java 语言简介 .....    | 1  |
| 1.1.2 Java 语言特点 .....    | 3  |
| 1.2 Java 开发运行环境 .....    | 5  |
| 1.2.1 JDK 的安装 .....      | 5  |
| 1.2.2 系统环境变量的设置 .....    | 8  |
| 1.2.3 编译命令和执行命令的用法 ..... | 10 |
| 1.2.4 编辑器介绍 .....        | 11 |
| 1.3 一个简单 Java 应用程序 ..... | 14 |
| 1.4 本章小结 .....           | 16 |
| 习 题 .....                | 16 |

## 第 2 章 Java 语言基础

|                         |    |
|-------------------------|----|
| 2.1 标识符和关键字 .....       | 17 |
| 2.1.1 标识符 .....         | 17 |
| 2.1.2 关键字 .....         | 17 |
| 2.2 常量和变量 .....         | 18 |
| 2.2.1 常 量 .....         | 18 |
| 2.2.2 变 量 .....         | 18 |
| 2.2.3 变量作用域 .....       | 19 |
| 2.3 基本数据类型 .....        | 20 |
| 2.3.1 基本数据类型 .....      | 20 |
| 2.3.2 数据类型转换 .....      | 22 |
| 2.4 运算符和表达式 .....       | 23 |
| 2.4.1 赋值运算符与赋值表达式 ..... | 23 |
| 2.4.2 算术运算符与算术表达式 ..... | 23 |
| 2.4.3 关系运算符与关系表达式 ..... | 25 |
| 2.4.4 逻辑运算符与逻辑表达式 ..... | 25 |
| 2.4.5 位运算符 .....        | 26 |
| 2.4.6 移位运算符 .....       | 27 |
| 2.4.7 条件运算符 .....       | 28 |

|       |                   |    |
|-------|-------------------|----|
| 2.4.8 | 运算符的优先级与结合性 ..... | 29 |
| 2.5   | 控制语句 .....        | 29 |
| 2.5.1 | 分支语句 .....        | 30 |
| 2.5.2 | 循环语句 .....        | 34 |
| 2.5.3 | 跳转语句 .....        | 38 |
| 2.6   | 注释语句 .....        | 39 |
| 2.7   | 数 组 .....         | 40 |
| 2.7.1 | 声明数组 .....        | 41 |
| 2.7.2 | 创建数组 .....        | 41 |
| 2.7.3 | 数组元素的使用 .....     | 42 |
| 2.7.4 | 数组的初始化 .....      | 42 |
| 2.7.5 | 数组使用举例 .....      | 42 |
| 2.8   | 本章小结 .....        | 44 |
|       | 习 题 .....         | 45 |

## 第 3 章 面向对象的编程 I(基本语法)

|       |   |    |
|-------|---|----|
| 3.1   | 面向对象程序设计概述 .....                            | 47 |
| 3.1.1 | 面向对象方法的特征 .....                             | 47 |
| 3.2   | 类的定义 .....                                  | 49 |
| 3.2.1 | 类的声明 .....                                  | 49 |
| 3.2.2 | 创建类的实例对象 .....                              | 52 |
| 3.3   | 构造函数 .....                                  | 55 |
| 3.4   | 方法的使用 .....                                 | 59 |
| 3.4.1 | 方法的声明 .....                                 | 59 |
| 3.4.2 | 方法的定义 .....                                 | 60 |
| 3.4.3 | 方法的调用 .....                                 | 60 |
| 3.4.4 | 参数传递的方式 .....                               | 60 |
| 3.4.5 | 对象作为参数的传递 .....                             | 62 |
| 3.4.6 | 命令行参数的使用 .....                              | 63 |
| 3.5   | 访问控制 .....                                  | 64 |
| 3.5.1 | Java 中的访问控制(public,private,protected) ..... | 64 |
| 3.5.2 | static 关键字的使用 .....                         | 66 |
| 3.5.3 | final 关键字的使用 .....                          | 69 |
| 3.6   | 本章小结 .....                                  | 70 |
|       | 习 题 .....                                   | 70 |

## 第 4 章 面向对象的编程 II(高级语法)

|       |                 |    |
|-------|-----------------|----|
| 4.1   | 继承和多态性的概念 ..... | 71 |
| 4.1.1 | 继承的概述 .....     | 71 |

|       |                        |    |
|-------|------------------------|----|
| 4.1.2 | 多态性的概述 .....           | 72 |
| 4.2   | 继承机制 .....             | 72 |
| 4.2.1 | 继承的定义 .....            | 72 |
| 4.2.2 | 类中属性的继承与隐藏 .....       | 73 |
| 4.2.3 | 类中方法的继承、覆盖与重载 .....    | 74 |
| 4.2.4 | 在子类中使用构造函数 .....       | 78 |
| 4.3   | 多态性与重载 .....           | 81 |
| 4.3.1 | 多态性的体现 .....           | 81 |
| 4.3.2 | 父类对象与子类对象间的使用和转化 ..... | 81 |
| 4.3.3 | 构造函数的重载与继承 .....       | 83 |
| 4.3.4 | 方法的动态调用 .....          | 84 |
| 4.4   | 抽象类 .....              | 84 |
| 4.4.1 | 抽象类的定义 .....           | 84 |
| 4.4.2 | 抽象的方法 .....            | 85 |
| 4.4.3 | 抽象类的使用 .....           | 85 |
| 4.5   | 包与接口 .....             | 88 |
| 4.5.1 | Java 语言中的包 .....       | 88 |
| 4.5.2 | Java 语言中的接口 .....      | 89 |
| 4.6   | 本章小结 .....             | 95 |
|       | 习 题 .....              | 95 |

## 第 5 章 Java 常用类和接口

|       |                            |     |
|-------|----------------------------|-----|
| 5.1   | 字符串处理类 .....               | 97  |
| 5.1.1 | String 类 .....             | 97  |
| 5.1.2 | StringBuffer 类 .....       | 106 |
| 5.2   | 集合类 .....                  | 109 |
| 5.2.1 | 集合的概念 .....                | 109 |
| 5.2.2 | 核心接口 .....                 | 110 |
| 5.2.3 | 常用集合类 .....                | 113 |
| 5.2.4 | 对集合类遍历的接口 .....            | 115 |
| 5.3   | 时间日期类 .....                | 116 |
| 5.3.1 | Date 类 .....               | 116 |
| 5.3.2 | Calendar 类 .....           | 118 |
| 5.3.3 | DateFormat 类 .....         | 119 |
| 5.3.4 | SimpleDateFormat 类 .....   | 120 |
| 5.4   | Math 类 .....               | 121 |
| 5.4.1 | Math 类的属性和方法 .....         | 121 |
| 5.4.2 | Math 类的应用示例 .....          | 122 |
| 5.5   | 随机数处理类 Random .....        | 123 |
| 5.6   | 系统类 System 和 Runtime ..... | 124 |



|       |                 |     |
|-------|-----------------|-----|
| 5.6.1 | System 类 .....  | 124 |
| 5.6.2 | RunTime 类 ..... | 126 |
| 5.7   | 本章小结 .....      | 128 |
|       | 习 题 .....       | 129 |

## 第 6 章 Java 异常处理

|       |                              |     |
|-------|------------------------------|-----|
| 6.1   | 异常的概念 .....                  | 131 |
| 6.1.1 | Java 的异常处理机制 .....           | 132 |
| 6.1.2 | Java 中异常类的结构 .....           | 133 |
| 6.2   | 异常处理 .....                   | 135 |
| 6.2.1 | 使用 try 和 catch 捕获和处理异常 ..... | 135 |
| 6.2.2 | 使用 throws 子句声明异常 .....       | 142 |
| 6.2.3 | throw 语句 .....               | 144 |
| 6.2.4 | 使用异常处理语句的注意事项 .....          | 145 |
| 6.3   | 自定义异常 .....                  | 146 |
| 6.4   | 本章小结 .....                   | 148 |
|       | 习 题 .....                    | 148 |

## 第 7 章 图形用户界面的创建

|       |                                |     |
|-------|--------------------------------|-----|
| 7.1   | 图形用户界面的概述 .....                | 149 |
| 7.1.1 | AWT 与 Swing .....              | 149 |
| 7.1.2 | 图形用户界面元素分类 .....               | 150 |
| 7.2   | 常用容器类与容器布局管理 .....             | 151 |
| 7.2.1 | 顶层容器(JFrame) .....             | 151 |
| 7.2.2 | 中间容器——面板类(JPanel) .....        | 153 |
| 7.2.3 | 中间容器——滚动面板类(JScrollPane) ..... | 154 |
| 7.2.4 | 布局管理器 .....                    | 156 |
| 7.3   | 事件处理 .....                     | 164 |
| 7.3.1 | 事件处理模型 .....                   | 164 |
| 7.3.2 | 事件及监听者 .....                   | 167 |
| 7.3.3 | 窗口事件 .....                     | 169 |
| 7.3.4 | 事件适配器 .....                    | 171 |
| 7.3.5 | 键盘事件 .....                     | 171 |
| 7.3.6 | 鼠标事件 .....                     | 173 |
| 7.4   | 本章小结 .....                     | 176 |
|       | 习 题 .....                      | 176 |

## 第 8 章 常用 Swing 组件

|     |                   |     |
|-----|-------------------|-----|
| 8.1 | 常用 Swing 组件 ..... | 177 |
|-----|-------------------|-----|

|       |                           |     |
|-------|---------------------------|-----|
| 8.1.1 | 常用 Swing 组件的继承关系 .....    | 177 |
| 8.1.2 | 标签(JLabel) .....          | 177 |
| 8.1.3 | 文本组件 .....                | 178 |
| 8.1.4 | 按钮组件 .....                | 181 |
| 8.1.5 | 组合框(JComboBox) .....      | 187 |
| 8.1.6 | 列表框(JList) .....          | 189 |
| 8.1.7 | 表格(JTable) .....          | 191 |
| 8.2   | 菜单设计 .....                | 194 |
| 8.2.1 | 弹出式菜单 .....               | 198 |
| 8.3   | 对话框设计 .....               | 199 |
| 8.3.1 | 对话框(JDialog) .....        | 199 |
| 8.3.2 | 标准对话框(JOptionPane) .....  | 200 |
| 8.3.3 | 文件对话框(JFileChooser) ..... | 203 |
| 8.4   | 本章小结 .....                | 205 |
|       | 习 题 .....                 | 205 |

## 第 9 章 I/O 处理

|       |                          |     |
|-------|--------------------------|-----|
| 9.1   | Java 流概述 .....           | 207 |
| 9.1.1 | 流的基本概念 .....             | 207 |
| 9.1.2 | Java 流类的层次结构 .....       | 208 |
| 9.1.3 | 流类的基本用法 .....            | 209 |
| 9.2   | Java 流相关类 .....          | 210 |
| 9.2.1 | 字节输入流 .....              | 210 |
| 9.2.2 | 字节输出流 .....              | 214 |
| 9.2.3 | 字符输入流 .....              | 219 |
| 9.2.4 | 字符输出流 .....              | 224 |
| 9.2.5 | 流的转换 .....               | 227 |
| 9.3   | 随机文件访问 .....             | 230 |
| 9.4   | File 类 .....             | 231 |
| 9.5   | 对象流和序列化 .....            | 234 |
| 9.5.1 | 序列化的概念 .....             | 234 |
| 9.5.2 | ObjectOutputStream ..... | 235 |
| 9.5.3 | ObjectInputStream .....  | 235 |
| 9.5.4 | 序列化示例 .....              | 236 |
| 9.5.5 | 定制序列化 .....              | 238 |
| 9.6   | 本章小结 .....               | 239 |
|       | 习 题 .....                | 239 |

## 第 10 章 多线程编程

|        |                |     |
|--------|----------------|-----|
| 10.1   | 线程基础 .....     | 241 |
| 10.1.1 | 线程基本概念 .....   | 241 |
| 10.1.2 | 线程的运行机制 .....  | 242 |
| 10.2   | 线程的创建和启动 ..... | 243 |
| 10.2.1 | 线程的创建 .....    | 243 |
| 10.2.2 | 线程的启动 .....    | 247 |
| 10.3   | 线程状态和转换 .....  | 248 |
| 10.4   | 线程控制 .....     | 250 |
| 10.4.1 | 线程睡眠 .....     | 250 |
| 10.4.2 | 线程让步 .....     | 251 |
| 10.4.3 | 线程间协作 .....    | 253 |
| 10.4.4 | 后台线程 .....     | 254 |
| 10.4.5 | 线程优先级 .....    | 256 |
| 10.5   | 线程同步处理 .....   | 256 |
| 10.5.1 | 多线程引发的问题 ..... | 257 |
| 10.5.2 | 同步代码块 .....    | 258 |
| 10.5.3 | 同步方法 .....     | 261 |
| 10.5.4 | 线程间通信 .....    | 263 |
| 10.5.5 | 死 锁 .....      | 266 |
| 10.6   | 本章小结 .....     | 268 |
|        | 习 题 .....      | 268 |

## 第 11 章 Java 网络编程技术

|        |                       |     |
|--------|-----------------------|-----|
| 11.1   | 网络通信基本知识 .....        | 271 |
| 11.1.1 | 网络通信基本概念 .....        | 271 |
| 11.1.2 | TCP 协议和 UDP 协议 .....  | 272 |
| 11.1.3 | Java 网络程序设计技术 .....   | 273 |
| 11.2   | URL 编程技术 .....        | 273 |
| 11.2.1 | URL 和 URL 类 .....     | 274 |
| 11.2.2 | URLConnection 类 ..... | 276 |
| 11.2.3 | InetAddress 类 .....   | 277 |
| 11.3   | TCP 编程技术 .....        | 279 |
| 11.3.1 | 网络套接字 Socket .....    | 279 |
| 11.3.2 | Socket 类 .....        | 280 |
| 11.3.3 | ServerSocket 类 .....  | 281 |
| 11.3.4 | C/S 程序设计实例 .....      | 283 |
| 11.4   | UDP 编程技术 .....        | 295 |

|        |                   |     |
|--------|-------------------|-----|
| 11.4.1 | 数据报通信基本概念         | 295 |
| 11.4.2 | DatagramPacket 类  | 295 |
| 11.4.3 | DatagramSocket 类  | 296 |
| 11.4.4 | MulticastSocket 类 | 303 |
| 11.5   | 本章小结              | 310 |
|        | 习 题               | 310 |

## 第 12 章 数据库编程

|        |                   |     |
|--------|-------------------|-----|
| 12.1   | 关系数据库与 SQL 语言     | 311 |
| 12.1.1 | 关系数据库的基本概念        | 311 |
| 12.1.2 | 数据定义语言            | 311 |
| 12.1.3 | 数据操纵语言            | 313 |
| 12.1.4 | 数据查询语言            | 314 |
| 12.2   | Java 数据库编程简介      | 314 |
| 12.2.1 | JDBC 基础           | 314 |
| 12.2.2 | JDBC 驱动程序类型       | 315 |
| 12.3   | 通过 JDBC 访问数据库     | 317 |
| 12.3.1 | 利用 JDBC 访问数据库基本流程 | 317 |
| 12.3.2 | 加载数据库驱动程序         | 317 |
| 12.3.3 | 创建与数据源的连接         | 318 |
| 12.3.4 | 操作数据库             | 320 |
| 12.3.5 | 处理操作结果            | 324 |
| 12.3.6 | 关闭操作              | 326 |
| 12.4   | JDBC 编程实例         | 326 |
| 12.4.1 | 数据库连接设置           | 326 |
| 12.4.2 | 创建数据库连接           | 326 |
| 12.4.3 | 创建数据表             | 327 |
| 12.4.4 | 向表中添加数据           | 328 |
| 12.4.5 | 修改数据              | 330 |
| 12.4.6 | 删除数据              | 331 |
| 12.4.7 | 查询数据信息            | 332 |
| 12.5   | JDBC 事务           | 335 |
| 12.6   | 本章小结              | 338 |
|        | 习 题               | 338 |

## 第 13 章 图书信息查询系统

|        |            |     |
|--------|------------|-----|
| 13.1   | 图书信息查询系统概述 | 339 |
| 13.1.1 | 系统简介       | 339 |
| 13.1.2 | 需求描述       | 339 |

|        |                     |     |
|--------|---------------------|-----|
| 13.2   | 系统设计 .....          | 340 |
| 13.2.1 | 系统结构 .....          | 340 |
| 13.2.2 | 系统角色业务流程分析 .....    | 341 |
| 13.2.3 | 数据库设计 .....         | 341 |
| 13.3   | 系统实现 .....          | 343 |
| 13.3.1 | 实现系统的类及其之间的关系 ..... | 344 |
| 13.3.2 | 主要界面 .....          | 346 |
| 13.3.3 | 各 Java 类的设计 .....   | 349 |
| 13.3.4 | 配置文件 .....          | 389 |
| 13.3.5 | 图片文件 .....          | 390 |
| 13.4   | 系统的运行与发布 .....      | 390 |
| 13.4.1 | 运行环境 .....          | 390 |
| 13.4.2 | 配置数据源 .....         | 391 |
| 13.4.3 | 系统的运行 .....         | 392 |
| 13.4.4 | 系统的发布 .....         | 394 |
| 13.5   | 本章小结 .....          | 395 |
|        | 习 题 .....           | 395 |



# 第 1 章

# Java 基础知识

Java 是一种安全的程序设计语言,它提供了诸多安全保障机制。Java 语言适用于 Internet 环境,是一种被广泛使用的网络编程语言。Java 语言从根本上改变了网络应用程序的开发和使用方式,并成为在许多应用领域特别是 Internet 网络应用领域最受欢迎的开发与编程语言。本章将对 Java 语言的发展历程、特点、开发运行环境,以及如何编译并执行 Java 程序等内容进行介绍。通过本章的学习,读者将会对 Java 语言有一个初步的了解,并能够顺利地搭建 Java 程序的开发运行环境。

## 1.1 Java 语言简介及特点

本节主要介绍 Java 语言的发展历史和 Java 语言的特点。

### 1.1.1 Java 语言简介

Java 是由 Sun Microsystems 公司于 1995 年 5 月推出的 Java 程序设计和 Java 平台的总称。下面我们来简单了解一下其发展的过程。

#### 1. Java 发展简史

Java 的历史要追溯到 1991 年,当时美国 Sun Microsystems 公司的 Patrick Naughton 及其伙伴 James Gosling 带领的工程师小组想要设计一种小型的计算机语言,主要应用对象是类似有线电视转换盒这类消费设备。由于这些消费设备的处理能力和内存都很有限,所以语言必须非常小且能够生成非常紧凑的代码。另外,由于不同的设备生产商会选择不同的中央处理器(CPU),因此这种语言的关键是不能与任何特定的体系结构捆绑在一起。这个项目被命名为 Green。

为了使整个系统与平台无关,Gosling 首先从改写 C/C++ 语言编译器着手。但是 Gosling 在改写过程中感到仅仅使用 C 语言是无法满足需要的,因为 C++ 语言太复杂且庞大,安全性也差,无法满足项目设计的需要,于是他在 1991 年 6 月开始开发一种新的语言,并命名为 Oak(一种精巧而安全的网络语言,适合于多线程编程),这就是 Java 语言的前身。后来发现 Oak 已是另一个公司的注册商标,这种语言才改名为 Java(Java 本是太平洋上一个盛产咖啡的岛屿的名字),并配了一杯冒着热气的咖啡图案作为它的标志。

由于要求 Java 这种语言必须非常小而且能够生成紧凑的代码,还要求该语言与平台无关。这些要求促使开发团队想起了很早以前的一种模型,某些 Pascal 的实现曾经在早期的 PC 上尝试过这种模型。以 Pascal 的发明者 Niklaus Wirth 为先驱,率先设计出一

种为假想的机器(虚拟机)生成中间代码的可移植语言,这种中间代码可以应用于所有已经正确安装解释器的机器上。于是,Green 项目组的工程师也使用了虚拟机(Java 虚拟机),从而解决了课题中的主要问题(平台无关性)。

1992 年,Green 项目发布了它的第一个产品,称之为“\* 7”。这个产品具有非常智能的远程控制。遗憾的是 Sun 公司对生产这个产品并不感兴趣,并且 Green 项目组的人员也没有找出其他的方法来将他们的技术推向市场。到了 1994 年 Green 项目组(这时换了一个新名字——First Person 公司)解散了。

在此期间,Internet 的万维网也日渐发展壮大,Web 的关键是把超文本页面转换到屏幕上的浏览器,当时的浏览器主要是 Mosaic。

Java 语言的开发者设计并开发了一个功能更加强大的浏览器,该浏览器最终演变为 HotJava 浏览器。为了炫耀 Java 语言超强的能力,HotJava 浏览器采用 Java 编写,他们让 HotJava 浏览器具有执行网页中内嵌代码的能力。这一“技术印证”在 1995 年的 Sun-World 上得到了展示,同时引发了人们延续至今的对 Java 的狂热追逐。

1996 年初,Sun 发布了 Java 的第 1 个版本 Java 1.0,但 Java 1.0 不能用来进行真正的应用开发,后来的 Java 1.1 弥补了其中的大多部分明显的缺陷,大大改进了反射能力,并为 GUI 编程增加了新的事件处理模型。

1998 年 JavaOne 会议的头号新闻是即将发布 Java 1.2 版。这个版本取代了早期玩具式的 GUI,并且它的图形工具箱更加精细而且具有较强的可伸缩性,更加接近“一次编写,随处运行”的承诺。然后,Sun 公司将其名称改为更加吸引人的“Java 2 标准版软件开发工具箱 1.2 版”。

标准版的 1.3 和 1.4 版本对最初的 Java 2 版本做出了某些改进,扩展了标准类库,提高了系统性能。5.0 版是自 1.1 版以来第一个对 Java 语言做出重大改进的版本(这一版本原来被命名为 1.5 版,在 2004 年的 JavaOne 会议之后,版本数字升至 5.0)。这个版本添加了泛型类型(generic type),其挑战性在于添加这一特性并没有对虚拟机做出任何修改。

2005 年 6 月,JavaOne 大会召开,Sun 公司发布了 Java SE 6。此时,Java 的各种版本已经更名并取消其中的数字“2”,J2EE 更名为 Java EE,J2SE 更名为 Java SE,J2ME 更名为 Java ME。

## 2. Java 语言的影响及应用前景

Java 语言是新一代面向对象的程序设计语言,特别适合于 Internet 应用程序的开发,它的硬件和软件平台的无关性直接威胁到 Windows 和 Intel 的垄断地位。用 Java 编程成为技术人员的一种时尚,并对未来软件的开发产生了重大影响。

Java 语言对软件开发技术的影响可以归纳为如下几个方面:

① 软件的需求分析。可将用户的需求进行动态的、可视化描述,以提供设计者更加直观的要求。而用户的需求是各种各样的,不受地区、行业、部门、爱好的影响,这些需求都可以用 Java 语言描述清楚。

② 软件的开发方法。由于 Java 语言是纯面向对象的程序设计语言,所以完全可以用面向对象的技术与方法来开发软件,这符合最新的软件开发规范要求。

③ 动画效果。Java 语言的动画效果远比 GUI 技术逼真,尤其是利用 WWW 提供的巨大动画资源空间,可以共享全世界的动态画面资源。

④ 软件最终产品。用 Java 语言开发的软件具有“可视化、可听化、可操作化、交互、动画、动作”等特点。

Java 语言有着广泛的应用前景,大体上可以从以下几个方面来考虑其应用:

- ① 所有面向对象的应用开发,包括面向对象的事件描述、处理等。
- ② 计算过程的可视化、可操作化的软件开发。
- ③ 动态画面的设计,包括图形图像的调用。
- ④ 交互操作的设计(选择交互、定向交互、控制流程等)。
- ⑤ Internet 的系统管理功能模块的设计,Web 页的动态设计、管理交互操作设计等。
- ⑥ Intranet 上的软件开发(直接面向企业内部用户的软件)。
- ⑦ 其他应用类型的程序(移动计算、嵌入式 Java 技术、实时 Java 等)。

## 1.1.2 Java 语言特点

Java 众多的突出特点使得它受到了大众的欢迎。归纳起来,Java 语言具有以下一些显著特点。

### 1. 简单性

人们希望构建一个无需深奥的专业训练就可以进行编程的系统,并且要符合当今的标准惯例。因此,尽管人们发现 C++ 不太适用,但在设计 Java 的时候还是尽可能地接近 C++,以便系统更易于理解。Java 剔除了 C++ 中许多很少使用、难以理解、易混淆的特性。例如,Java 中没有指针、结构体等概念,没有 #include 和 #define 等预处理器,也没有多重继承的机制。

简单的另一个方面是“小”。Java 的目标之一是支持开发能够在小型机器上独立运行的软件。基本的解释器以及类支持大约仅为 40KB;再加上基础的标准类库和对线程的支持(基本上是一个自包含的微内核)大约需要增加 175KB。在当时,这是一个了不起的成就。当然,由于不断的扩展,类库已经相当庞大了。现在有一个独立的具有较小类库的 Java 微型版(Java Micro Edition)用于嵌入式设备。

### 2. 面向对象性

Java 是一个纯粹的面向对象的语言,强调的是面向对象的特性,对软件工程技术能够提供很强的支持。Java 语言的设计集中于对象及其接口,它提供了简单的类机制及动态的接口模型。与其他面向对象的语言一样,Java 具备继承、封装及多态性等通常的特性,更提供了一些类的原型,程序员可以通过继承机制,实现代码的复用。另外 Java 的继承机制独特,在设计时去掉了不安全的因素,因此使用 Java 可以编制出非常复杂但逻辑清晰的系统。

### 3. 分布式与安全性

从诞生之日起 Java 就与网络联系在一起,它强调网络特性,使它成为一种分布式程序设计语言。Java 语言包括一个支持 HTTP 和 FTP 等基于 TCP/IP 协议的子库,它提供一个 Java.net 包,通过它可以完成各种层次上的网络连接。因此 Java 语言编写的应用程序可以凭借 URL 打开并访问网络上的对象,其访问方式与访问本地文件系统几乎完全相同。Java 语言另一个 Socket 类提供的可靠流式网络的连接,使程序设计者可以非常方便地创建分布式应用程序。

Java 程序在语言定义阶段、字节码检查阶段及程序执行阶段进行的三级代码安全检



查机制,对参数类型匹配、对象访问权限、内存回收、Java 小应用程序的正确使用等都进行了严格的检查和控制,可以有效地防止非法代码的侵入,阻止对内存的越权访问,能够避免病毒的侵害。

#### 4. 与平台无关性

如果基本数据类型设计依赖于具体的计算机和操作系统,会给程序的移植带来很大不便。Java 语言通过定义独立于软、硬件平台的基本数据类型及其相关运算,确保数据在任何硬件平台上保持一致。为了实现平台无关性,Java 语言规定了统一的基本数据类型。

Java 程序编译后成为二进制代码,即字节码(bytecode)。字节码就是虚拟机的机器指令,与平台无关。字节码有统一的格式,不依赖于具体的硬件环境。在任何安装 Java 运行时环境的系统上,都可以执行这些代码。也就是说,只要安装了 Java 运行环境,Java 程序就可以在任意的处理器上运行。这些字节码指令对应于 Java 虚拟机中的表示,Java 解释器得到字节码后,对它进行转换,使之能够在不同的平台运行。运行环境针对不同的处理器指令系统,把字节码转换为不同的具体指令,保证了程序的“到处运行”。

#### 5. 解释和编译特性

Java 开发环境把 Java 源程序编译后生成一种称为字节代码(bytecode)的中间代码,字节代码非常类似于机器指令代码,但并不是二进制的机器指令代码,且字节代码并不针对一种特定的机器,所以 Java 程序不需要重新编译便可在众多不同的计算机上执行,只要该机器上预先装有 Java 语言运行系统,这是其编译特性。Java 程序编译后产生字节代码,其运行要借助于 Java 解释器,Java 解释器直接对 Java 字节代码进行解释执行。以字节代码形式发布的 Java 程序运行在 JVM 环境上,JVM 将字节代码翻译成具体的 CPU 机器指令,因此 Java 解释器是与硬、软件平台有关的,在不同的平台上使用不同的 JVM 实现(与平台相关部分的工作由 JVM 而不是 Java 编译器来完成,因为平台的种类比起应用软件的数量要少得多)。Java 解释器使 Java 程序在某一特定硬、软件平台环境中直接运行目标代码指令,这种连接程序通常比编译程序所需资源少,所以程序员可以花上更多的时间用在创建源程序上,而不必考虑运行环境,这是其解释特性。

#### 6. 多线程

多线程机制使应用程序能够并行执行,通过使用多线程,程序设计者可以分别用不同的线程完成特定的行为,而不需要采用全局的事件循环机制,这样就很容易实现网络上的实时交互行为和实时控制性能。

大多数高级语言(包括 C、C++ 等)都不支持多线程,用它们只能编写顺序执行的程序(除非有操作系统 API 的支持)。Java 内置了语言级多线程功能,提供现成的 Thread 类,只要继承这个类就可以编写多线程的程序,使用户程序并行执行。Java 提供的同步机制可保证各线程对共享数据的正确操作,完成各自的特定任务。在硬件条件允许的情况下,这些线程可以直接分布到各个 CPU 上,充分发挥硬件性能,减少用户等待的时间。

#### 7. 动态执行

Java 执行代码是在运行时动态载入的,这种动态特性使它适合于一个不断发展的环境。在网络环境下,Java 语言编写的代码用于瘦客户机架构可减少维护工作。另外,类