



Java 程序设计 基础与提高

江春华 蔡洪斌 编著

电子科技大学出版社

TP312
435

DZ45/21

Java 程序设计基础与提高

江春华 蔡洪斌 编著



00471181



电子科技大学出版社

内 容 提 要

Java 既是 Internet 上的网络编程语言, 又是计算机单机环境下的编程语言。本书分基础篇和提高篇, 对 Java 进行了系统的介绍。基础篇由浅入深地阐述 Java 的基本语言和面向对象技术的基本概念, 介绍如何用 Java 语言实现面向对象程序设计。提高篇系统地介绍 Java 的所有类库, 分析用户界面、图形、声音、动画、输入输出流等的程序设计方法, 介绍了 Applet 程序和 Application 程序的设计方法, 对 Application 程序还重点讲解了菜单系统、对话框等特殊部件的使用方法, 最后较为详细地介绍网络环境下的程序设计以及 Java 和 C 语言的接口方法。

本书注重实践, 以大量的程序实例帮助读者理解面向对象技术的概念, 掌握 Java 程序设计方法, 使读者更加容易理解有关面向对象程序设计的概念和方法。全书以 JDK1.1.1 版为基础进行阐述, 所有程序实例都在 Windows 95 支持下的 100MHz 奔腾兼容机上调试通过。

本书既适合作为本科生和研究生的实用教材, 也适合有 Java 基础的程序设计员特别是网络编程人员使用。

声 明

本书无四川省版权防盗标识, 不得销售, 版权所有, 违者必究, 举报有奖, 举报电话: (028) 6636481 6241146 3201496

Java 程序设计基础与提高

江春华 蔡洪斌 编著

出 版: 电子科技大学出版社 (成都建设北路二段四号, 邮编: 610054)

责任编辑: 杜 倩

发 行: 新华书店

印 刷: 峨眉电影制片厂印刷厂

开 本: 787×1092 1/16 印张 22 字数 535 千字

版 次: 1998 年 6 月第一版

印 次: 1999 年 1 月第二次印刷

书 号: ISBN 7—81043—931—6/TP·409

印 数: 3001—6000 册

定 价: 26.00 元

TP312
435

序

自半个世纪前人类发明了电子数字计算机以来，这一技术领域即以历史上少有的速度更新换代，向前猛进。到目前为止，发展速度有增无减，的确令人惊叹！

Java 程序设计语言吸收了当前计算机领域中一系列技术革新成果中最有价值的部分，从而在一个体现从量变到质变的计算机技术发展过程中，标志了一个里程碑性质的新成果的出现。它代表了面向对象程序设计方法在目前的最高实用水平。新一代的计算模式——网络计算模式，也因为得到 Java 语言的支持，从而大大地向前迈进了一步。Java 语言的确是计算机领域中一项非常有价值的最新成果。

另一个让人感叹之处，是在当今计算机行业竞争如此激烈的时期，由一个计算机硬件公司研制出的 Java 语言，居然一下子得到了几乎全部世界上的各大计算机软、硬件公司的支持，到 1997 年底，世界将会有 1 亿部以上的台式机使用它。实际上，它很有可能成为计算机程序设计的标准语言，用它编写的程序可以运行于各种计算机上。如果真能如此，这将是计算机行业里破天荒的一件大事。

对于 Java 语言这样一个极为新颖而重要的题材，自然有不少学者适时涉及，出版了一些书籍。但对目前为止，像此书这样既包含程序设计的基本概念和技法的说明，又涉及高层次的程序设计应用技巧的介绍，在国内尚不多见。此书另外一个成功之处在于强调实用性，凡是重要的概念和知识点，均以恰当的实例作为文字解说的补充，使读者加深了对各种重要概念和基础知识的理解，又掌握了多种灵活应用 Java 语言的基本方法。此书的行文也颇为流畅，完全不像有些计算机应用速成类书籍中那样充满了外国人的习语。以上这些特点，使此书首先成了一本非常合适的 Java 语言教材。对于已有工作经验的科技工作者或者工程技术人员，此书又是一本很好的教学参考书。

编著者虽然都是年轻人，但他们有扎实的专业理论基础知识和丰富的实践经验，有对新知识、新技术的敏感和很强的领悟能力，再加上勤奋进取，广览精探，因而在发展极快的计算机领域作出了可喜的成绩。我衷心祝愿他们再接再厉，精益求精，不断取得新的成绩。

电子科技大学

杨国纬

1997 年 10 月 5 日

目 录

基础篇

第一章 Java 简介	(3)
1.1 Java 语言的特点	(3)
1.1.1 简洁性	(3)
1.1.2 安全性	(3)
1.1.3 面向对象	(4)
1.1.4 动态性	(4)
1.1.5 体系结构中中和可移植性	(4)
1.1.6 高性能性	(4)
1.1.7 多线程	(5)
1.1.8 解释执行	(5)
1.1.9 分布式	(5)
1.2 Java 开发工具 (JDK) 简介	(5)
1.2.1 Java 编译器——javac	(5)
1.2.2 Java 解释器——java	(6)
1.2.3 Java 类分解器——javap	(7)
1.2.4 C 文件生成器——javah	(7)
1.2.5 小应用程序观察器——AppletViewer	(8)
1.3 Java 应用程序——Application 和 Applet	(8)
1.3.1 Application 程序编写与运行步骤	(8)
1.3.2 Applet 程序编写与运行步骤	(10)
1.4 Java 虚拟机原理	(11)
1.4.1 Java 生成可执行代码过程	(11)
1.4.2 Java 虚拟机规范	(12)
小结	(12)
第二章 Java 语言基础	(14)
2.1 Java 程序的基本结构	(14)
2.2 Java 符号集	(15)
2.2.1 关键字	(15)
2.2.2 标识符	(16)
2.2.3 程序中的注释	(16)
2.3 数据的简单类型	(17)
2.4 常量	(17)

2.4.1 整数	(18)
2.4.2 浮点数	(18)
2.4.3 字符型常量	(18)
2.4.4 布尔型常量	(19)
2.5 变量和变量声明	(19)
2.6 变量的作用域及初始化	(20)
2.6.1 作用域	(20)
2.6.2 初始化	(21)
2.7 运算符	(21)
2.7.1 算术运算符	(22)
2.7.2 关系和布尔运算符	(25)
2.7.3 位运算符	(26)
2.7.4 赋值运算符和数据的类型转换	(27)
2.7.5 条件运算符	(29)
2.8 表达式	(30)
小结	(30)
第三章 流程控制、数组	(31)
3.1 条件语句: if-else	(32)
3.2 多分支语句 switch 和中断语句 break	(34)
3.3 循环语句	(36)
3.3.1 while 语句	(36)
3.3.2 do-while 语句	(37)
3.3.3 for 语句	(37)
3.4 continue 语句	(40)
3.5 return 语句	(41)
3.6 数组和字符串	(42)
3.6.1 数组声明	(42)
3.6.2 数组元素的引用及初始化	(43)
3.6.3 字符串	(45)
小结	(45)
第四章 类和对象	(46)
4.1 面向对象的程序设计	(46)
4.1.1 对象	(47)
4.1.2 消息	(48)
4.1.3 类	(48)
4.2 类的创建	(49)
4.2.1 类声明	(49)
4.2.2 类体定义	(49)
4.2.3 类定义形式	(54)

4.3	方法重载	(55)
4.4	构造方法	(56)
4.5	对象	(57)
4.5.1	对象创建	(57)
4.5.2	对象使用	(58)
4.5.3	对象清除	(62)
4.5.4	finalize()方法	(62)
4.6	三种特殊方法的固定声明方式	(63)
4.7	静态成员	(63)
	小结	(64)
第五章 超类、子类 and 继承性		(65)
5.1	创建子类	(65)
5.2	成员变量的隐藏和方法覆盖	(65)
5.3	变量 null、this 和 super	(66)
5.3.1	null	(66)
5.3.2	this	(66)
5.3.3	super	(67)
5.4	运行时的多态	(69)
5.5	方法覆盖 (overriding)	(70)
5.6	final 类和方法	(70)
5.7	抽象类和方法	(71)
5.8	Object 类	(72)
	小结	(74)
第六章 包和接口		(75)
6.1	包	(75)
6.1.1	包声明 --- package 语句	(75)
6.1.2	包引入 --- import 语句	(76)
6.2	编译和运行包中的类	(77)
6.3	访问权限	(78)
6.3.1	公共类型 public	(79)
6.3.2	受保护类型 protected	(79)
6.3.3	缺省类型 friendly	(79)
6.3.4	私有类型 private	(80)
6.4	举例	(80)
6.5	实例成员和类成员	(84)
6.5.1	实例变量和类变量	(85)
6.5.2	实例方法和类方法	(85)
6.6	创建和使用接口	(87)

6.6.1 接口的定义	(87)
6.6.2 接口的实现	(89)
6.6.3 接口的类型	(89)
6.7 完整的 Java 源文件	(90)
小结	(90)
第七章 Java Applet 编程	(91)
7.1 Applet 执行框架	(92)
7.2 Applet 程序和结构	(92)
7.3 Applet 的安全限制	(94)
7.4 Applet 的生命周期	(95)
7.4.1 加载 Applet	(96)
7.4.2 卸载和重载 Applet	(96)
7.4.3 多次加载 Applet	(96)
7.4.4 退出浏览器	(96)
7.5 浏览器类库构成	(97)
7.6 Applet 类的层次	(97)
7.7 Applet 类的构造方法、实例变量和方法	(98)
7.8 Applet 编程的方法	(99)
7.8.1 扩充 Applet 基类	(99)
7.8.2 Applet 属性参数	(100)
7.8.3 装载图像	(101)
7.8.4 显示图像	(102)
7.8.5 在 HTML 页中加入 Applet	(105)
小结	(106)

提 高 篇

第八章 AWT：用户界面设计	(109)
8.1 java.awt 包概述	(109)
8.2 容器和部件	(110)
8.3 AWT：GUI 布置管理器	(114)
8.3.1 FlowLayout（流布置管理器）	(114)
8.3.2 BorderLayout（周边布置管理器）	(115)
8.3.3 CardLayout（卡片布置管理器）	(117)
8.3.4 GridLayout（格栅布置管理器）	(118)
8.3.5 GridBagLayout 和 GridBagConstraints	(119)
8.4 AWT 部件	(122)
8.4.1 Button（按钮）	(127)
8.4.2 Label（标签）	(128)
8.4.3 Checkbox（检查框）	(129)

8.4.4	Choice (选择框)	(131)
8.4.5	List (列表框)	(132)
8.4.6	Scrollbar (滚动条)	(134)
8.4.7	TextField (单行文本区)	(135)
8.4.8	TextArea (文本区)	(137)
8.4.9	Canvas (画布)	(138)
小结		(139)
第九章 事件处理		(140)
9.1	GUI 事件	(140)
9.2	事件的种类	(141)
9.2.1	ActionEvent 事件	(142)
9.2.2	ItemEvent 事件	(142)
9.2.3	TextEvent 事件	(142)
9.2.4	KeyEvent 事件	(142)
9.2.5	MouseEvent 事件	(143)
9.2.6	AdjustmentEvent 事件	(143)
9.2.7	FocusEvent 事件	(144)
9.3	事件处理方法	(144)
9.3.1	用覆盖方法处理事件	(144)
9.3.2	实现接口处理事件	(148)
9.4	部件与事件	(151)
9.4.1	Button 部件与事件	(152)
9.4.2	Checkbox 部件与事件	(153)
9.4.3	Choice 部件与事件	(156)
9.4.4	List 部件与事件	(157)
9.4.5	TextArea 部件与事件	(162)
9.4.6	TextField 部件与事件	(163)
9.4.7	Scrollbar 部件与事件	(164)
9.4.8	Canvas 部件与事件	(166)
小结		(169)
第十章 Application 程序设计		(170)
10.1	Application 程序设计	(170)
10.2	菜单系统	(178)
10.2.1	菜单类	(178)
10.2.2	加入菜单	(182)
10.2.3	ActionEvent 事件	(185)
10.2.4	ItemEvent 事件	(188)
10.2.5	动态调整菜单	(191)
10.2.6	增加快捷键	(196)
10.2.7	弹出式菜单	(198)

10.3	对话框	(200)
10.3.1	生成对话框	(201)
10.3.2	事件处理	(202)
10.3.3	数据传输	(205)
10.4	文件对话框	(209)
10.5	Applet 和 Application 的集成程序	(212)
	小结	(215)
第十一章	异常处理	(217)
11.1	异常类的层次	(217)
11.2	异常处理	(219)
11.3	嵌套的异常处理	(221)
11.4	throw 语句	(223)
11.5	throws 语句	(226)
	小结	(229)
第十二章	多线程设计	(230)
12.1	基本概念	(230)
12.1.1	线程	(230)
12.1.2	优先级与调度	(230)
12.1.3	线程生命周期	(231)
12.1.4	线程控制方法	(231)
12.2	多线程实现方法	(233)
12.2.1	生成 Thread 子类	(233)
12.2.2	实现 Runnable 接口	(236)
12.3	线程的同步	(238)
	小结	(243)
第十三章	多媒体应用	(244)
13.1	概述	(244)
13.1.1	图形、图像的基本概念	(244)
13.1.2	音频系统	(244)
13.1.3	动画基础	(244)
13.2	图形显示	(245)
13.2.1	基本绘图方法	(245)
13.2.2	绘图颜色	(252)
13.2.3	绘图模式	(255)
13.2.4	字体	(258)
13.2.5	绘制文字	(260)
13.2.6	坐标转换	(261)
13.3	图像处理	(261)

13.3.1 图像显示	(262)
13.3.2 用算法生成图像	(270)
13.3.3 图像处理的其他类	(273)
13.4 播放声音	(273)
13.5 动画基础	(277)
13.5.1 实时动画	(278)
13.5.2 块动画	(281)
小结	(287)
第十四章 输入输出流类	(288)
14.1 文件系统	(288)
14.1.1 文件路径和属性	(290)
14.1.2 创建目录和删除文件	(292)
14.1.3 文件更名	(293)
14.1.4 目录清单	(294)
14.2 读写文件	(295)
14.3 抽象流类	(301)
14.3.1 InputStream 类	(301)
14.3.2 OutputStream 类	(302)
14.4 文件输入输出流类	(302)
14.4.1 FileInputStream 类	(302)
14.4.2 FileOutputStream 类	(304)
14.5 加强输入输出流类	(305)
14.5.1 FilterInputStream 类和 FilterOutputStream 类	(305)
14.5.2 BufferedInputStream 类和 BufferedOutputStream 类	(306)
14.5.3 DataInputStream 类和 DataOutputStream 类	(308)
14.6 其他输入输出流类	(308)
14.6.1 SequenceInputStream 类	(308)
14.6.2 管道输入输出流类	(310)
14.7 新增流类	(313)
小结	(315)
第十五章 网络编程	(316)
15.1 网络基础	(316)
15.2 java.net 包	(320)
15.3 Internet 寻址	(321)
15.4 URL 类与 URLConnection 类	(323)
15.5 编写服务程序	(327)
小结	(333)

第十六章 本地方法——与 C 语言接口	(334)
16.1 基本步骤	(334)
16.2 数据传送	(336)
16.2.1 定义参数	(336)
16.2.2 访问 Java 成员变量	(336)
16.2.3 访问 String 对象	(337)
小结	(338)
参考资料	(339)

基础篇

- 第一章 Java 简介
- 第二章 Java 语言基础
- 第三章 流程控制、数组
- 第四章 类和对象
- 第五章 超类、子类和继承性
- 第六章 包和接口
- 第七章 Java Applet 编程

第一章

Java 简介

在计算机技术快速发展的今天, Internet 网络成为现代信息高速公路的代名词。Java 伴随 Internet 的流行发展开来, Java 为 Internet 提供了适应计算机网络环境的编程语言, 其 Applet 应用程序是专门用于 WWW (World Wide Web) 环境, Java 被誉为“长时间来最卓越的程序设计语言”, Java 语言将成为新的开发环境标准。

Java 语言是由 Sun Microsystem 公司从 1991 年开始开发, 它的早期产品是一种交互式操作软件, 取名为 Oak。虽然在技术上是一个成功的系统, 但在商业上遭到失败。正值 WWW 处在不断发展的新型阶段, 虽然 WWW 页面内容丰富, 能够做到对声音、图像、文字并茂处理, 但是它们是静态的, 不能动态交互式处理, 大大地限制了 WWW 的发展。WWW 需要一种具有动态地嵌入式编程语言, 而具有动态交互式操作能力的 Oak 软件正好能满足这一要求, 它语言简洁、精练, 又有较强的安全性, 能支持在各种处理器上运行 Java 编译器。Sun 公司正是看到越来越受人们欢迎的 WWW 更适合用 Oak 语言开发的潜力, 开始对 Oak 语言不断地改进和完善, 以便适应 Internet 网络环境; 另一方面用 Oak 语言开发编制一个真正的应用程序。在 1995 年, Oak 语言被改名为 Java 语言, 而应用程序被作为 WWW 浏览器取名叫 HotJava。

1.1 Java 语言的特点

Java 语言是一种新型的编程设计语言, 广泛地应用于 Internet 网络编程设计。它是跨平台的适用于分布式计算机环境的面向对象程序设计语言。它具有简洁、安全、面向对象、动态、体系结构中立、可移植、高性能、多线程、解释执行和分布式等特性。

1.1.1 简洁性

Java 语言是一种面向对象语言, 它为用户提供了最基本方法来完成某种任务。只要理解了这些相关的概念, 就能够理解并编制出适合于实际情况的程序。

Java 语言简单有效, 这与原设计的初衷有关。在所设计的系统中必须具有简洁性, 一方面尽可能采用面向对象的编程特点; 另一方面要使编程者尽量避免可能的麻烦和错误, Java 的这一特性是以增加系统的复杂性为代价的。

1.1.2 安全性

Java 语言的一切对内存访问都是通过对象实例变量实现的, 防止用户在网络系统或分

布系统环境下使用特洛伊木马等手段访问对象的私有成员。Java 语言不支持指针，避免指针操作的错误，Java 语言提供了内存管理机制，有一个自动搜集“内存垃圾”的程序。Java 在字节码的传输过程中使用了公开密钥加密机制 (PKC)，而在运行环境提供了四级安全性保障机制：

- ① 字节码校验器 (ByteCode Verifier)
- ② 类装载器 (Class Loader)
- ③ 运行时内存布局
- ④ 文件访问限制

1.1.3 面向对象

Java 语言具有真正的面向对象语言的特点，它支持静态和动态的代码继承及重用。所谓面向对象是一种计算机编程方法，它是将具有属性和行为的现实世界中的对象映射到计算机的编程上，属性表示数据，行为表示程序代码。

面向对象语言的任何方面均是基于消息或对象的。而消息是传递支持对象间所有可能的互相作用，面向对象编程就是对对象和消息编程，它支持封装、多态性和继承。封装就是将对象内的数据和代码联编起来，形成一个对象；多态性是指一个接口，有多个内在实现形式表示；继承是指某一对象直接使用另一对象的所有属性和方法的过程，它可以简化对象的定义，即在定义某个对象时可以只定义有别于父对象的属性或行为，其他部分则可以从父对象那里继承来重用。

1.1.4 动态性

Java 语言的设计能够适应于发展变化的环境。由于 Java 语言编程的基本组成单元是类，在运行中 Java 的类是动态装载的，只要 Java 在分布式系统中动态地维护应用程序和对支持类库间一致性，就可以避免像类库升级这一类问题。

Java 语言滞后联编机制充分利用面向对象编程风格的优点，真正做到即插即用的模块功能。

1.1.5 体系结构中立和可移植性

Java 语言编程的程序可以不经任何改动在不同的硬件或软件平台上执行，即 Java 编译器所生成的可执行代码不是基于任何硬件平台的，它是基于一种抽象的处理器——Java 虚拟机 (Java Virtual Machine) 实现的。

Java 程序的运行，首先要经过编译，再进行解释这两个过程实现的。

Java 编译器所生成的代码不是针对某一具体的硬件体系结构的，而是一种叫做字节码指令代码，它与硬件体系结构无关。在运行过程中，则由针对于运行系统硬件体系结构的 Java 解释器，将字节码转换成与该系统相对应的指令。

Java 的可移植性使人们可在网络上的各种硬件平台运行同一 Java 程序。

1.1.6 高性能性

虽然 Java 是解释执行语言，但它的字节码在编译生成时，带有许多的编译信息，在 Java

字节码格式设计中充分考虑到它与机器码的执行效率，很容易直接转换成对应于特定处理器的高性能机器码。

Java 字节码的执行效率非常接近于 C 或 C++ 生成的机器码执行效率。

1.1.7 多线程

使用多线程，可以用不同的线程完成特定的行为，而不需要采用全局事件循环机制，这样利于实现网络上实时交互操作。

Java 多线程支持表现在两个方面，一方面是它自身的多线程，可以利用系统的空闲执行一些常规处理等；另一方面，提供对多线程的语言级支持，提高程序执行效率。

1.1.8 解释执行

Java 解释器直接对 Java 字节码进行解释执行。由于其所带的编译信息，使得执行连接过程更加简单。

1.1.9 分布式

Java 是一个适用于网络的语言。它提供的类库支持对 TCP/IP 协议处理，可以通过 URL 地址访问网络上其他的对象。

Java 支持 WWW 的客户机/服务器计算机网络模型，它可以支持分布式的数据分布和操作分布。它是一门非常适合 Internet 网络和分布式环境的编程语言。

1.2 Java 开发工具 (JDK) 简介

Java 开发工具 JDK (Java Developer's Kit)，有 SPARC/Solaris, X86/Solaris 版本和 Microsoft Windows 95、Windows NT 版本。它们可以从 javasoft.com 站点处免费下载，也可以从其他一些镜像站点下载，但是从 Sun 公司站点能够得到 JDK 的最新的版本。

下面介绍 JDK 环境工具，包括有 Java 编译器 `javac`，Java 解释器 `java`，Java 类分解器 `javap`，Java 剖析工具 `javaprof`，C 文件生成器 `javah`，Java 小应用程序观察器 `AppletViewer`。

1.2.1 Java 编译器——`javac`

Java 编译器本身就是一个 Java 语言编写的应用程序，它与其他编译语言不同，它的作用是把 Java 源程序（.java 文件）编译生成 Java 字节码（.class 文件）。这种字节代码与机器代码类似，却不针对具体的机器。

Java 语言编译器的编译格式为：

```
javac [options] source_file
```

其中，options 为选项，source_file 是扩展名为 .java 的源程序。

options 有如下选项：

`-classpath <路径;路径;...>` 指定 `javac` 在编译过程中，查找出现的某个类定义搜索路径表，路径间以“;”分隔