

高等职业教育计算机软件、计算机网络专业系列教材

JAVA 程序设计基础

刘文军 编

重庆大学出版社

内容提要

本书共分 12 章, 在全面讲述 Java 语言语法的基础上, 结合大量的程序实例对 Java 语言的使用、功能、特性等诸方面进行了详细的讲解, 结构严谨、布局合理、重点突出、实例丰富, 能够使读者很快地掌握 Java 语言程序设计的方法和技巧, 同时对面向对象程序的设计也有较深入的了解。

本书既可以作为高职高专计算机软件、计算机网络专业学生使用, 也可作为其他工程技术人员特别是软件开发者的自学用书。

图书在版编目(CIP)数据

JAVA 程序设计基础/刘文军编. —重庆: 重庆大学出版社, 2004. 9

(高等职业教育计算机软件、计算机网络专业系列教材)

ISBN 7-5624-3009-8

. J 刘 JAVA 语言—程序设计—高等学校: 技术学校—教材

. TP312

中国版本图书馆 CIP 数据核字(2004)第 078980 号

计算机软件
高等职业教育 专业系列教材
计算机网络

Java 程序设计基础

编 刘文军

责任编辑:王海琼 温佐丹 版式设计:吴庆渝

责任校对:任卓惠 责任印制:秦 梅

*

重庆大学出版社出版发行

出版人:张鸽盛

社址:重庆市沙坪坝正街 174 号重庆大学(A区)内

邮编:400030

电话:(023)65102378 65105781

传真:(023)65103686 65105565

网址: <http://www.cqup.com.cn>

邮箱: fxk@cqup.com.cn(市场营销部)

全国新华书店经销

重庆华林天美彩色报刊印务有限公司印刷

*

开本:787×1092 1/16 印张:18.75 字数:410 千

2004 年 9 月第 1 版 2004 年 9 月第 1 次印刷

印数:1—5 000

ISBN 7-5624-3009-8/TP·444 定价:29.00 元(含 1 CD)

本书如有印刷、装订等质量问题,本社负责调换

版权所有 翻印必究

编
委
会

顾 问 邱玉辉

主 任 樊启宙 张学礼

副主任 杨滨生 任德齐 刘彩琴

委 员 (以姓氏笔画为序)

王 津 吴 焱 孙 辉

陈 晴 张洪星 张 英

黄顺强 袁开榜 龚小勇



序

高等职业教育具有“高等”和“职业”的双重特征,其目标是培养生产、建设、管理、服务第一线需要的高等技术应用型专门人才,是世界教育发展的共同趋势。近年来,我国高等教育的结构改革极大促进了高等职业教育事业的发展,高等职业教育已成为我国高等教育的重要组成部分。

为了适应我国高等教育的改革,进一步满足高等职业教育计算机软件计算机网络专业的教学及学科建设的需要,在全国各高等职业技术学院的支持下,重庆大学出版社采取学校、企业合作的形式,在全国十余所高等职业技术学院及企业(武汉职业技术学院、邢台职业技术学院、南昌工程学院、昆明冶金高等专科学校、重庆电子职业技术学院、重庆正大软件技术学院、重庆正大软件有限公司等)计算机相关专业的专家、学者中成立了编委会,并组建了一批具有丰富教学和实践经验的“双师型”作者队伍,力求编写出一套适合高等职业教育特点的高质量系列教材。

教学与生产相结合,理论和实践相结合,学校和社会相结合是高等职业教育的生命线;以技术应用能力和职业素质为主线来设计教学体系是高等职业教育教学改革的方向。依据高等职业教育的发展方向,本系列教材将强调理论知识的应用;注重基本能力、专业能力、综合能力及其技能的培养作为编写宗旨。

本系列教材将计算机与信息技术行业的标准及其技术岗位的需求作为组织编写的依据;在保证理论够用的基础上,根据产业结构、技术岗位体系以及职业岗位能力的要求组织理论和实训教材,并将职业教育的教学模式和方法融入其中。为了便于教学,今后将进一步建立学习资源网站,开

发立体化教材。

本系列教材特点如下:

1. 以培养计算机网络、软件应用型人才为目标,遵循教育规律,系列教材的各分册相互衔接,并具有相关性和独立性。

2. 教材编写模块化。即将两个专业各自划分为若干个模块,它们既共同拥有共享的基础模块,又各自拥有一定选择余地的专业模块。各门专业课程教材均可以一条逐步深化的主线将教学贯穿于学生学习的始终,形成“基础”、“提高”和“应用”3个层次的分阶段教学模式,学生在不断提高应用水平后可以直接承揽工程。

本系列教材的体系结构如下:

通用 模块	基础模块	计算机专业英语	* 计算机应用数学(上)	计算机应用电子技术	
		* 计算机网络技术基础	计算机应用数学(下)	* JAVA 程序设计基础	
		Delphi 程序设计基础	Visual Basic 程序设计基础	* Visual C++ 程序设计基础	
		* 计算机网络操作系统	计算机硬件技术基础	网页设计与网站建设	
	数据库模块	* 数据库技术基础与应用	数据库技术提高	数据库技术应用	
专业 模块	软件专业	软件工程模块	* 软件工程	软件测试技术	
		可视化编程模块	JAVA 程序设计提高	Visual Basic 程序设计提高	* Delphi 程序设计提高与应用
			JAVA 程序设计应用	Visual Basic 程序设计应用	Delphi 程序设计应用
	Visual C++ 程序设计提高		Visual C++ 程序设计应用		
	多媒体编程模块	* 多媒体程序设计(VB版)			
	网络编程模块	网络程序设计			
	网络专业	局域网模块	网络专业局域网技术基础	局域网技术应用	
		广域网模块	广域网技术应用		
工程模块		* 网络安全与防火墙技术	网络系统集成与综合布线工程技术		

注: * 课程为秋季推出的教材,其他课程将陆续推出,实训教材正在筹划之中。

希望各院校和企业教师、专家参与本系列教材的建设,并请毛遂自荐担任后续教材的主编或参编,联系 E-mail: lich@cqup.com.cn。

3. 理论知识以够用为度,以实例、项目的工程实现为主线,将重点放在应用及操作技能上。

4. 力求创新。将新技术、新工艺纳入教材,尽可能体现文化性、社会性和艺术性,以利于提高学生综合的素质。

5. 思考题和习题具有启迪性和创新性。在编程、网络工程类教材的各章习题中大都有包含与教材内容同步的中小型工程习题(或试验),全书最终将完成多个完整的工程实例。

本系列教材面向高等职业教育,适合于各类高等专科学校、高等职业学校、成人高等学校及高等院校主办的二级职业技术学院,并可作为从事计算机工作的工程技术人员的自学参考书。

该套教材的出版,重庆大学出版社的领导和编辑做了大量的工作,各教材的作者付出了艰苦的努力。但是,由于教材从策划到出版仅用了一年多一点的时间,承担教材编写任务的教师大多都担负着繁重的教学任务。在时间紧、任务重的情况下,教材中一定有不少不尽如人意之处,诚挚希望读者提出批评和建议,以便再版时改进。

编委会
2004年8月



前 言

因特网的互联性、开放性和共享信息的模式,打破了信息传播的传统方式,改变了人们的工作和生活方式。随着社会信息化程度的提高,社会生活中依赖于计算机解决的问题越来越多,也越来越复杂。面向对象的技术正在逐步取代传统的面向过程的程序设计方法,成为计算机应用开发领域的主流方向。

诞生于 20 世纪 90 年代初的 Java 语言是一种面向对象的程序设计语言。它所具有的与平台无关的特性、较高的可靠性和安全机制以及内嵌的网络支持等特色,特别适合于 Internet 应用程序的开发。它不包括 C++ 中的比较难学难用的指针,使得 Java 语言易学、易懂、易用,已成为当今编写网络程序的首选工具之一。

如今,Java 已经成为主流开发语言之一。它在作为 Web 编程工具的同时,在企业应用开发中也扮演着重要的角色。

全书共分 12 章。第 1 章介绍了 Java 语言的发展和特点,Java 环境设置;第 2 章介绍了 Java 语言的基本组成、数据类型、运算符及数组的定义和使用方法以及各种流程控制语句;第 3 章介绍了面向对象程序设计的方法,包括类、对象及方法的概念及应用;第 4 章介绍了类高级特征,例如接口、内部类等;第 5 章介绍了异常的概念及 Java 异常处理机制;第 6 章介绍了 Java 抽象窗口工具集(AWT)的图形界面组件;第 7 章介绍了 Java 的图形处理,包括如何绘制图形、显示图像,如何设计动画;第 8 章介绍了 Java 的委托事件模型以及对各种事件的响应方法,同时介绍了 Swing 组件在使用上与 AWT 组件的不同;第 9 章讲解了 Java 的文件操作,介绍了几种常见的流类的使用。第 10 章介绍了多线程程序设计方法,包括线程的概念、线程的创建与管理方法,详细说明了线程的互斥

和同步的问题,列举了大量的实际例子。第 11 章介绍了 Java 网络编程技术;第 12 章介绍了 Java Applet 小应用程序的开发。

本书所针对的读者是高等职业教育的学生,以及其他对 Java 语言和面向对象编程技术感兴趣的读者。本书内容遵循教学规律,按照易学、易懂、易操作、易掌握的原则,结合 Java 语言的特点,由浅入深、难易结合,通过丰富的例子循序渐进地介绍了 Java 语言及其程序设计的方法。本书所有例题作者都进行了编译和调试运行通过。

在本书的编写过程中,得到了南昌工程学院各级领导的大力支持和协助,承蒙重庆大学杨瑞龙教授审阅了全部书稿,并提出了许多宝贵意见和建议,谨此一并致以衷心的感谢。

本书的全部程序和有关软件在本书所附的光盘上,如果对本书有什么批评和建议,可以发送电子邮件至: myprivacy@sina.com。

由于编者水平有限,书中难免存在一些错误和疏漏,恳请各位专家和读者批评指正。

刘文军

2004 年 5 月

目 录

1 Java 简介及环境设置

1.1	Java 简介	1
1.1.1	Java 的历史	1
1.1.2	Java 的特性	2
1.1.3	Java 与 C++ 的比较	4
1.1.4	Java 的发展	5
1.2	Java 开发环境的设置	5
1.3	编辑、编译和运行 Java 程序	7
1.3.1	编译程序	8
1.3.2	虚拟机程序	10
	习题 1	12

2 Java 的基本语法

2.1	Java 语言的基本组成部分	13
2.1.1	标识符	13
2.1.2	关键字	14
2.1.3	程序块与注释	14
2.1.4	分隔符	15
2.1.5	转义字符	16
2.1.6	数据类型	17
2.1.7	变量	19
2.1.8	常量	20
2.1.9	数据类型转换	21
2.1.10	运算符与表达式	23

2.2	Java 语句及其控制结构	28
2.2.1	分支语句	28
2.2.2	循环语句	34
2.2.3	转移语句	40
2.2.4	数组	44
2.2.5	字符串	48
	习题 2	52

3 面向对象技术

3.1	类与对象	54
3.2	类的创建	54
3.2.1	类的声明	54
3.2.2	类的主体	56
3.3	对象的创建	56
3.3.1	创建对象	56
3.3.2	构造函数	57
3.3.3	对象的使用	58
3.4	类的封装	59
3.4.1	封装的概念	59
3.4.2	封装	59
3.4.3	实例成员和类成员	61
3.5	类的继承	61
3.5.1	创建子类	61
3.5.2	this 引用、super 引用和 instanceof 对象运算符	62
3.5.3	最终类和抽象类	65
3.6	类的多态性	67
3.6.1	方法的重载	67
3.6.2	方法的覆盖	69
3.7	递归方法	70
3.8	包	72
3.8.1	Java 的 API 介绍	75
3.8.2	Java 的常用包	81
3.8.3	包及类的引用	81
3.8.4	自定义包	86
	习题 3	86

4 Java 类和对象的高级特征

4.1	接口	88
4.1.1	定义接口	88
4.1.2	实现接口	89
4.2	内部类	90
4.2.1	内部类特性	90
4.2.2	静态内部类	91
4.2.3	内部类的引用	93
4.2.4	方法中定义的内部类	94
4.3	Java 中的文档注释	95
	习题 4	97

5 Java 的异常处理

5.1	异常	98
5.2	使用 try 和 catch 捕获异常	99
5.3	创建自定义异常	103
5.4	异常的抛出	104
5.5	try...catch 语句的嵌套	107
	习题 5	110

6 图形界面设计

6.1	创建图形界面	111
6.1.1	抽象窗口工具集 AWT	111
6.1.2	Java 的图形类结构	111
6.1.3	窗口与面板	113
6.1.4	按钮	123
6.1.5	标签	127
6.1.6	列表框	128
6.1.7	文本域	128
6.1.8	菜单	130
6.1.9	对话框	134
	习题 6	135

7 Java 的图形处理

7.1	Java 图形处理概述	136
7.2	基本图形的输出.....	138
7.2.1	直线的输出	138
7.2.2	矩形图的输出	141
7.2.3	椭圆、圆及圆弧的输出.....	145
7.2.4	多边形的输出	147
7.3	设置图形的颜色.....	149
7.3.1	创建 Color 类对象	150
7.3.2	设置当前颜色	150
7.4	文字的图形化输出	152
7.4.1	Font 类	152
7.4.2	文本的显示和输出	152
	习题 7	154

4

8 Java 中的键盘和鼠标事件

8.1	鼠标事件	157
8.1.1	MouseListener 接口	157
8.1.2	MouseMotionListener 接口.....	157
8.2	处理鼠标事件	158
8.3	键盘事件	160
8.4	处理键盘事件	161
8.5	窗口事件	163
8.6	Swing 基础	165
8.6.1	Swing 的层次结构.....	165
8.6.2	Swing 组件.....	166
8.6.3	Swing 组件使用举例.....	167
	习题 8	168

9 Java 的输入输出

9.1	Java 输入输出概述	170
-----	-------------------	-----

9.1.1	输入输出流基本概念	170
9.1.2	输入输出流类	171
9.2	文件处理	176
9.2.1	File 类	177
9.2.2	文件的随机访问——RandomAccessFile 类	181
9.3	目录处理	184
	习题 9	186

10 多线程

10.1	多线程基础	188
10.1.1	线程状态和生命周期	189
10.1.2	线程优先级和线程调度	191
10.2	多线程设计	192
10.2.1	继承 Thread 类	192
10.2.2	实现 Runnable 接口	195
10.2.3	线程的互斥	198
10.2.4	线程间同步	209
10.2.5	线程死锁	214
10.2.6	其他方法的说明	220
10.2.7	线程的优先级	222
10.3	线程组	225
	习题 10	226

11 网络

11.1	Internet 地址	229
11.2	URL 通用资源定位	230
11.3	使用 URLConnection 对象来连接 URL 服务器	232
11.4	数据包方式实现服务器与客户通信	233
11.5	使用流的方式实现客户和服务之间的通信	236
11.5.1	客户端 Socket	236
11.5.2	服务器端 Socket	238
11.6	一个网络服务器类	239
11.6.1	线程组的使用	243

习题 11 247

12 Java 小应用程序 —— Applet

12.1 第一个 Applet 小应用程序 248

12.2 Applet 的运行平台—浏览器 249

12.2.1 浏览器 250

12.2.2 < Applet > 标签 250

12.2.3 用 getParameter() 方法取得参数 251

12.3 Applet 程序的运行机理 253

12.4 Applet 的程序结构 255

12.5 Applet 的程序设计 262

12.5.1 向 Applet 传递参数 264

12.5.2 标识网络资源 265

12.5.3 显示图像 265

12.5.4 播放声音 266

习题 12 275

附录 JBuilder X Enterprise 的使用 277

参考文献 284

1 Java 简介及环境设置

1.1 Java 简介

1.1.1 Java 的历史

如今十分热门的 Java 语言,最初是源自于 Sun 公司内部的绿色计划,其主要的目的是开发能用来控制“消费类电子产品”的软件。Sun 公司的开发小组最初的构想是以 C 或 C++ 来开发这类软件,可是很快他们发现 C 或 C++ 并不适合,因为能用来控制“消费性电子产品”的软件必须拥有独立的平台,而用 C 或 C++ 编写的程序,却必须针对不同的硬件或平台来编写不同的程序,再加上 C 和 C++ 本身的复杂性,很可能在使用一般的“消费性电子产品”时产生错误。因此开发小组决定放弃继续使用 C/C++ 来开发软件,而重新开发一种新的语言,于是在 1990 年设计出一个名为“Oak”的新语言。

Oak 便是 Java 的前身,是一种简单、可靠并且能够跨平台的程序语言,后来因为 Oak 这个名称已经被其他程序语言所使用,最后 Java 才从众多的名称中脱颖而出,其灵感来自于大多数程序员所喜爱的热咖啡,这也是 Java 启动时有一杯热咖啡出现的由来。

1993 年,开发小组发现像 Java 这种跨平台的程序语言非常适于应用在网络上,因此他们开发了第一个支持 Java Applet 的浏览器——HotJava。由于 Java 在 www 上所表现出的优越性,使得 Java 受到各界的关注,当 1995 年 Sun 公司正式发表 Java 的技术时,NetScape 公司也立即宣布 Netscape 2.0 将支持 Java Applet,此举让 Java 立刻成为计算机界的焦点,Java 的出现将网络带入了一个新纪元。未来 Java 将会不断地扩大它的应用领域,以网络为发展基础,把触角延伸至无线通讯、电子商务等相关领域,使 Java 在未来可以向多元化发展。

1.1.2 Java 的特性

Java 语言是一种简单、面向对象、分布式、稳定、安全、跨平台、可移植、可直译、高性能、多线程、动态加载性的语言。下面介绍以下几个方面特点:

简单性 Java 简单特性的定义是让用户易学易懂。Java 引入 C/C++ 的语法概念,其主要目的就是让原来使用 C/C++ 的程序员能够很快地接受 Java,并且很容易地开发程序。不过 Java 也删除了在 C/C++ 内使用较少或较容易被混淆的功能,其中最主要的是删除指针功能,以避免使用指针时可能会产生内存存取错误,Java 语言不需要使用指针。

另外,Java 还提供内存垃圾自动回收的功能,让程序员能够更专注于程序设计,而不必在内存管理的问题上分心,因而简化了程序设计的复杂程度。

面向对象 在面向对象的程序语言中,对象是类的实例,而类则是描述对象的模板。例如:用户可以将“电路设计图”视为一个类,而根据这个设计图所制造出来的“电路板”就是“电路设计图”类的对象。

Java 是一种面向对象的程序语言,在 Java 的语法结构中,用户可以发现它具有数据封装、继承、抽象化、多态性和可重用性等 5 种面向对象程序设计的特性。当用 Java 来设计程序时,应该以面向对象的方式来思考,而不是专注于程序流程。

分布性 Java 具备强大且易于使用的连网能力,在设计时就考虑到网络传输协议,因此它能很容易地处理 TCP/IP 协议。再加上 Java 是一种面向对象的程序语言,使 Java 运行时所需的对象可以分散于网络上,而不必局限于同一主机,通过 HTTP 或 FTP 等协议,Java 就可以利用网络来打开或是存取远程对象,就像存取本机的文件一样方便。

稳定性 Java 中不使用指针,Java 提供内存垃圾自动收集机制和异常处理机制,来排除程序运行时可能出现的错误。内存垃圾自动收集机制可以自动回收不用的内存空间,避免内存短缺的情况发生;异常处理机制则是将程序中所有非正常情况下的条件句柄集合在一起,以防止程序在特殊情况下被损坏。

安全性 安全是网络语言首先要考虑的问题,Java 具有安全的特性。Java 不允许用户利用程序直接对内存进行存取操作,而是通过 JVM(Java Virtual Machine,Java 虚拟机)在运行时动态配置内存区域,这样,用户便无法在程序中使用内存地址调用的方式来读取或更改内存地址中的内容。

JVM 在解释 Java 字节码时,会将所有不合法的数据和无权限的存取操作加以阻隔,不合法的字节码是无法被解释和运行的。

跨平台性 Java 字节码近似于机器码的中间码,不受限于计算机硬件设备和操作平台的种类,只要计算机中有 Java 的运行环境(即 JVM),Java 字节码就可以在那台计算

机上运行,如图 1.1 所示。Java 的跨平台性也是 Sun 公司最为强调的 Java 优势:一次编译,随处运行。

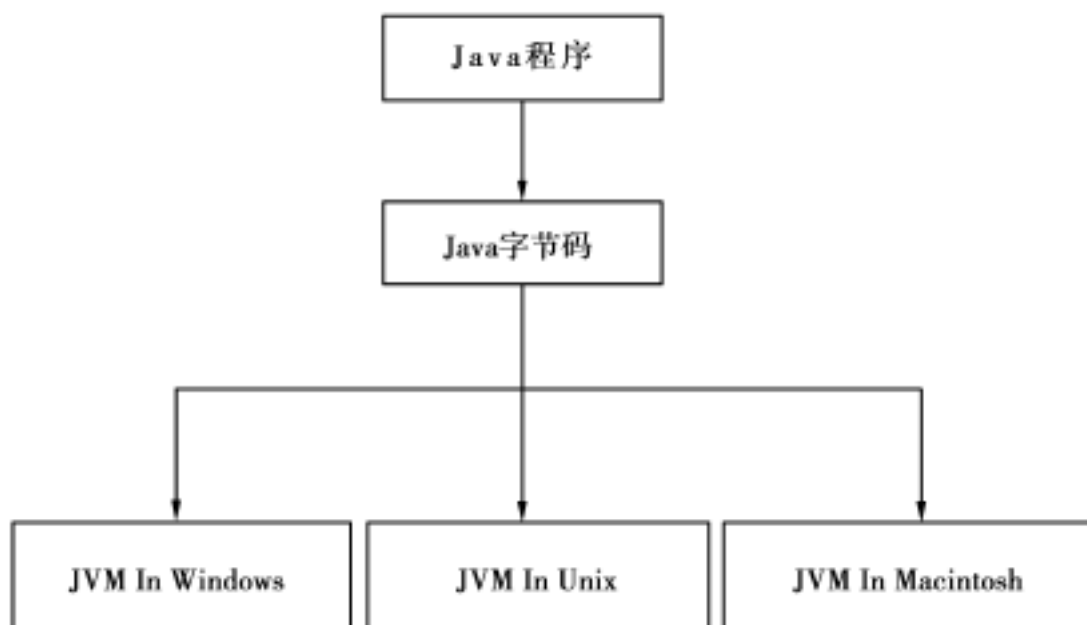


图 1.1 Java 的跨平台运行

可移植性 跨平台性是确保程序可移植的主要原因,除此之外还需要其他条件的配合。Java 严格规定数据类型和类结构的标准,使程序可以适用于所有的计算机平台,Java 的链接库(Class Library)还定义了一些可移植的接口(如抽象的窗口对象类)以及这些类在不同的计算机平台(如 Unix、Windows 和 Macintosh)上的定义结构,这些均使 Java 具有良好的可移植性。

JVM 并不是一个真正的机器,而是 Java 定义用来运行字节码的环境,实际上字节码并不直接在计算机平台上运行,而是在 JVM 上运行,在不同平台上,JVM 有不同的规范,因此可以满足平台独立的要求。

高性能 字节码是一种中间码,在运行时必须先解释并加载系统信息才能运行,因此无法像 C/C++ 所产生的运行文件可随时运行。但是 Java 字节码在编译时便考虑到机器码的产生,使字节码在转换为机器码时的速度和性能与 C/C++ 的运行文件几乎没有差别。

多线程 Java 具有开发多线程程序的功能,允许程序同时拥有数个线程来运行不同的工作,当一个线程在处理计算工作时,另一个线程可以接受输出命令,而第三个线程还能运行编辑工作。因此使用 Java 所开发出来的多线程程序比传统的单一线程程序所占用的 CPU 资源少。

10 动态加载性 Java 在设计加载对象时,并不是在程序一开始运行时就将所需对象全部加载,而是在程序运行过程中需要时才加载,即当 JVM 发现所需的对象未被加载时,便先在本机中查找,如果该对象在本机中不存在则通过网络查找并加载,因此 Java 运行时,所需的对象是根据程序运行情况被动态加载的。