



高等学校计算机教材

Java

实用教程

| 郑阿奇 主编 | 姜乃松 殷红先 编著 |



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

高等学校计算机教材

Java 实用教程

郑阿奇 主编

姜乃松 殷红先 编著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本教程包含了实用教程、习题、实验、综合应用实习等几个部分。内容主要包括 Java 基本程序设计、Java 图形界面设计、JBuilder 数据库和多媒体应用程序设计。实用教程一般在讲解内容后紧跟实例，并且实例程序一般均上机调试通过，实验部分通过实例先一步一步引导读者进行操作，然后提出问题让读者思考和在原来基础上自己进行操作和编程练习；实习部分是综合训练实际解决问题的能力。本教程各部分层次清楚，并依据教学特点进行精心编排，方便用户根据自己需要进行选择。

本书可作为大学本、专科有关课程的教材，由于其内容实用，也适合各类 Java 培训和广大用户自学与参考。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目(CIP)数据

Java 实用教程 / 郑阿奇主编. —北京：电子工业出版社，2005.4

高等学校计算机教材

ISBN 7-121-01039-9

I. J… II. 郑… 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（2005）第 022803 号

责任编辑：张荣琴 特约编辑：晓 鸽

印 刷：北京市京科印刷有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经 销：各地新华书店

开 本：787×1 092 1/16 印张：29.25 字数：768 千字

印 次：2005 年 4 月第 1 次印刷

印 数：5 000 册 定价：39.00 元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系。
联系电话：（010）68279077。质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前 言

Java 是 Sun 公司开发的新一代编程语言，它的平台无关性非常适合当前的网络应用。所以学习 Java 不仅是一种时尚，而且是一种潮流。

随着 Java 技术的日渐流行与普及，传统的 JDK 开发方式难以满足需要，许多公司相继推出了可视化的基于 Java 的集成开发工具，如 Visual Age for Java, JCreator, Workshop, JBuilder 等，其中 Borland 公司推出的 JBuilder 是目前最好的 Java 可视化集成开发工具之一。JBuilder 9.0 是最新推出的 Java 可视化集成开发工具，自带 JDK1.4.1，完全支持 JDK，它为 Java 的学习与开发提供了很好的环境，可以大大减少 Java 开发过程中遇到的种种不便。

本书将学习 Java 和用 Java 解决问题较好地结合起来，通过一个小实例，先介绍 JDK 开发方法，然后介绍用 JBuilder 开发方法，这样比较轻松地把读者领进了 JBuilder 的集成开发环境。此后，比较系统地介绍 Java，读者就可以在 JBuilder 的集成开发环境调试程序，同时更熟悉 JBuilder 的集成开发环境，为后面系统介绍 JBuilder 的开发应用准备好了条件。同时，后面介绍的 Java 内容又对前面的内容进行深化。实例程序一般也可在 JDK 下运行。

实用教程一般是在讲解内容后紧跟实例，并且实例程序均上机调试通过。实验部分是通过实例先一步一步引导读者进行操作，然后提出问题让读者思考和在原来基础上自己进行操作和编程练习，综合训练实际解决问题的能力。

实际上，本教程不仅适合于教学，也非常适合于 Java 的各类培训和用 JBuilder 编程开发应用程序的用户学习和参考。只要阅读本书，结合上机操作进行练习和实习，就能在较短的时间内基本掌握 Java 及其应用技术。

本教程配备同步 PowerPoint 课件和应用程序文件，可直接用于课堂教学和练习。需要者可以在 <http://www.hxedu.com.cn> 网站上免费下载。

本书由姜乃松（南京师范大学）、殷红先（南京师范大学）编写，郑阿奇（南京师范大学）对全书进行统编、定稿。其他很多同志对本书的编写提供了许多帮助，在此一并表示感谢！

目前，参加本套丛书编写的还有梁敬东、顾韵华、王洪元、杨长春、王一莉、曹弋、徐文胜、丁有和、刘启芬、张为民、丁有和、郑进、周怡君、赵阳、周旭琴、陈金辉、李含光、黄群等。

由于作者水平有限，不当之处在所难免，恳请读者批评指正。

编 者

2005.1

目 录

第 1 部分 Java 实用教程

第 1 章	Java 和 JBuilder 9.0 集成开发环境	1
1.1	Java 语言简介	1
1.1.1	Java 语言特点	1
1.1.2	一个简单的程序	2
1.1.3	JDK 简介	4
1.2	JBuilder 9 集成开发环境	6
1.3	JBuilder 应用程序开发	9
1.4	利用 JBuilder 9 制作可执行文件	14
1.5	小结	17
第 2 章	Java 基本程序设计	18
2.1	数据类型与变量	18
2.1.1	数据类型	18
2.1.2	常量	19
2.1.3	变量	21
2.1.4	Java 语言中的关键字	22
2.2	运算符和表达式	22
2.2.1	运算符	22
2.2.2	表达式	26
2.3	顺序结构	27
2.4	选择结构	27
2.4.1	if 语句	27
2.4.2	switch 语句	28
2.5	循环结构	30
2.5.1	循环语句	30
2.5.2	break 语句	34
2.5.3	continue 语句	34
2.6	小结	35
第 3 章	数组、字符串和排序	36
3.1	数组	36
3.1.1	一维数组	36
3.1.2	多维数组	38
3.2	字符串	41
3.2.1	字符串常量	41

3.2.2	String 类	42
3.2.3	StringBuffer 类	44
3.2.4	字符串的比较	45
3.2.5	字符串与其他数据类型互相转换	45
3.2.6	StringTokenizer 类	46
3.3	综合示例	47
3.4	排序与查找	51
3.5	小结	53
第 4 章	方法与递归	54
4.1	方法的定义与调用	54
4.1.1	方法的定义	54
4.1.2	return 语句	55
4.1.3	局部变量	55
4.1.4	形式参数	56
4.1.5	方法调用	56
4.2	参数传递	57
4.2.1	值传递机制	57
4.2.2	数组作为形式参数	58
4.2.3	引用作为形式参数	59
4.3	递归	59
4.4	System 类常用方法	62
4.5	小结	63
第 5 章	Java 面向对象程序设计	64
5.1	基本概念	64
5.2	类的定义	65
5.2.1	类的定义	65
5.2.2	公开类	66
5.2.3	final 类	66
5.2.4	成员变量	66
5.2.5	成员方法	71
5.2.6	构造函数	72
5.2.7	方法重载	73
5.3	对象	74
5.3.1	创建对象	74
5.3.2	使用对象	74
5.3.3	this 与 super	75
5.3.4	删除对象	75
5.4	继承	76
5.4.1	定义子类	76
5.4.2	子类构造函数	77

5.4.3	数据成员隐藏	78
5.4.4	方法重写	79
5.5	多态	80
5.5.1	编译时多态	81
5.5.2	运行时多态	81
5.6	抽象类与接口	83
5.6.1	抽象类	83
5.6.2	接口	84
5.7	初始化	85
5.7.1	类初始化块	85
5.7.2	对象初始化块	86
5.8	包 package	87
5.9	内部类	88
5.9.1	static 型的内部类	88
5.9.2	非 static 型的内部类	89
5.9.3	匿名内部类	90
5.10	综合示例	90
5.10.1	计算 24 点	90
5.10.2	航班管理	92
5.11	小结	101
第 6 章	异常处理	102
6.1	什么是异常	102
6.2	try-catch-finally 语句	102
6.3	声明抛出异常子句	104
6.4	抛出异常语句	105
6.5	定义自己的异常类	106
6.6	小结	108
第 7 章	输入/输出处理	109
7.1	字节流	109
7.1.1	InputStream 类	109
7.1.2	OutputStream 类	109
7.2	文件字节流	110
7.2.1	文件字节流的顺序处理	110
7.2.2	文件字节流的随机读/写	112
7.3	文件与目录的处理	117
7.4	过滤流与流的串接	120
7.4.1	DataInputStream 和 DataOutputStream 类	120
7.4.2	BufferedInputStream 和 BufferedOutputStream 类	122
7.4.3	PushbackInputStream 类	124
7.5	字符流	125

85	7.5.1 Reader 类和 Writer 类	125
95	7.5.2 InputStreamReader 和 OutputStreamWriter	126
98	7.5.3 FileReader 类和 FileWriter 类	127
118	7.5.4 BufferedReader 类和 BufferedWriter 类	129
118	7.5.5 PrintWriter 类	130
128	7.5.6 StringReader 和 StringWriter 类	131
128	7.5.7 PushbackReader 类	132
138	7.6 对象序列化(Serialization)	134
138	7.6.1 什么是对象的序列化	134
138	7.6.2 对序列化的进一步讨论	136
138	7.7 小结	137
138	第 8 章 多线程	138
138	8.1 什么是线程	138
138	8.2 线程的状态转换	139
138	8.3 创建线程	140
139	8.3.1 通过继承 Thread 类创建线程	140
139	8.3.2 通过接口创建线程	141
139	8.4 线程的控制	142
139	8.4.1 启动线程	142
140	8.4.2 终止线程	142
140	8.4.3 “线程中断”标记	144
140	8.4.4 自愿暂时放弃处理器	144
140	8.4.5 线程睡眠	145
140	8.4.6 检测线程状态	145
140	8.4.7 线程优先级及线程调度	145
140	8.4.8 线程间协作 join()	146
140	8.5 线程间的同步与互斥	146
140	8.5.1 临界区	146
140	8.5.2 对象锁机制	148
140	8.5.3 线程间同步	151
140	8.5.4 死锁	155
140	8.6 线程的其他通信方式——管道流	155
140	8.7 守护线程	157
140	8.8 线程组	158
140	8.9 小结	158
140	第 9 章 AWT 组件及应用	159
140	9.1 AWT 简介	159
140	9.2 容器	160
140	9.2.1 窗口 Frame	160
140	9.2.2 面板 Panel	162

9.3	布局管理器	164
9.3.1	FlowLayout	164
9.3.2	BorderLayout	166
9.3.3	GridLayout	167
9.3.4	CardLayout	169
9.3.5	GridBagLayout	171
9.3.6	null	173
9.4	AWT 事件处理机制	174
9.4.1	AWT 事件类	175
9.4.2	AWT 事件监听器及接口	175
9.4.3	注册监听器	176
9.4.4	事件适配器	177
9.4.5	组件与事件的关联	179
9.5	AWT 常用基本组件	179
9.5.1	标签 Label	179
9.5.2	命令按钮 Button	180
9.5.3	单行文本编辑框 TextField	181
9.5.4	多行文本编辑框 TextArea	185
9.5.5	复选按钮及单选按钮 Checkbox	187
9.5.6	列表框 List	191
9.5.7	下拉列表框 Choice	195
9.5.8	选单 Menu	197
9.6	小结	203
第 10 章	Swing 组件及应用	204
10.1	Swing 简介	204
10.2	Swing 常用组件(一)	206
10.2.1	窗口 JFrame	206
10.2.2	容器 JPanel	208
10.3	Swing 常用组件(二)	210
10.3.1	标签 JLabel	210
10.3.2	命令按钮 JButton	211
10.3.3	单行文本编辑框 JTextField	215
10.3.4	多行文本编辑框 JTextArea	216
10.3.5	口令输入框 JPasswordField	217
10.3.6	开关按钮 JToggleButton	220
10.3.7	复选按钮 JCheckBox	222
10.3.8	单选按钮 JRadioButton	223
10.3.9	按钮组 ButtonGroup	224
10.3.10	列表框 JList	230
10.3.11	下拉列表框 JComboBox	234

10.4	Swing 常用组件 (三)	235
10.4.1	选单	235
10.4.2	弹出式选单 JPopupMenu	242
10.4.3	工具栏 JToolBar	247
10.5	Swing 常用组件 (四)	253
10.5.1	滚动容器 JScrollPane	253
10.5.2	选项卡容器 JTabbedPane	255
10.5.3	进度条 JProgressBar	259
10.5.4	滑动条 JSlider	261
10.5.5	表格 JTable	264
10.5.6	树状视图 JTree	269
10.6	Swing 常用组件 (五)	274
10.6.1	用户对话框 JOptionPane	274
10.6.2	通用对话框 JDialog	278
10.6.3	文件选择对话框 JFileChooser	281
10.6.4	颜色选择对话框 JColorChooser	285
10.6.5	子窗口 JInternalFrame	288
10.7	JBuilder 中的图形用户界面设计示例——计算器	292
10.8	小结	300
第 11 章	JBuilder 数据库设计与应用	301
11.1	关系数据库概述	301
11.1.1	关系数据库概述	301
11.1.2	SQL 语言	301
11.1.3	Java 环境与数据库的连接——JDBC	303
11.2	常用的 JDBC API	306
11.2.1	DriverManager 类	306
11.2.2	Connection 接口	307
11.2.3	Statement 类和接口	307
11.2.4	PreparedStatement 接口	308
11.2.5	ResultSet 接口	309
11.2.6	ResultSetMetaData 接口	311
11.3	JBuilder 中的常用数据库组件 DataExpress	313
11.3.1	Database 类	314
11.3.2	QueryDataSet 类	315
11.3.3	ParameterRow 类	318
11.3.4	ProcedureDataSet 类	320
11.3.5	TextDataFile 类	322
11.3.6	TableDataSet 类	324
11.4	JBuilder 中的常用数据库组件 dbSwing	325
11.4.1	JdbLabel, JdbTextField, JdbTextArea 类	326

11.4.2	JdbToggleButton, JdbCheckBox, JdbRadioButton 类	326
11.4.3	JdbComboBox, JdbList 类	330
11.4.4	JdbTable, TableScrollPane 类	333
11.4.5	JdbTree 类	335
11.4.6	JdbStatusLabel, JdbNavToolBar 类	338
11.5	JBuilder 中的常用数据库组件 More dbSwing	339
11.5.1	JdbNavField 类	339
11.5.2	JdbNavList, JdbNavComboBox 类	340
11.6	其他数据库常用技术	341
11.6.1	可计算列	341
11.6.2	数据验证	343
11.6.3	排序	345
11.6.4	过滤	345
11.7	小结	348
第 12 章	多媒体程序设计	349
12.1	图形图像处理 Graphics 类	349
12.2	图形图像处理 Graphics2D 类	355
12.3	声音	359
12.4	多媒体	360
12.4.1	JMF 的安装设置	360
12.4.2	JMF 的常用功能	361
12.5	小结	364
第 13 章	网络程序设计	365
13.1	TCP 网络程序设计	365
13.1.1	InetAddress 类	365
13.1.2	ServerSocket 类	367
13.1.3	Socket 类	367
13.1.4	一个简单的 TCP 通信程序	368
13.1.5	多线程的 TCP 服务器模型	370
13.2	UDP 网络程序设计	372
13.2.1	DatagramSocket 类	373
13.2.2	DatagramPacket 类	373
13.2.3	一个 UDP 通信程序	374
13.2.4	组播	376
13.2.5	广播	379
13.3	URL	379
13.3.1	URL 类	379
13.3.2	URLConnection 类	380
13.3.3	向服务器发送 FORM 格式的数据	382
13.3.4	通过 HTTP 代理进行网络通信	385

13.4 小结	385
第 14 章 Java 小应用程序 Applet	386
14.1 Java Applet	386
14.1.1 什么是 Applet	386
14.1.2 Applet 生命周期	388
14.1.3 创建一个 Applet	388
14.1.4 嵌入 Applet	390
14.1.5 Applet 的安全性	390
14.1.6 在 Applet 中进行 AWT 绘制	391
14.2 Applet 通信	395
14.2.1 同一页 Applet 之间的通信	395
14.2.2 Applet 与浏览器之间的通信	397
14.3 小结	397

第 2 部分 习 题

第 1 章 习题	398
第 2 章 习题	398
第 3 章 习题	399
第 4 章 习题	400
第 5 章 习题	400
第 6 章 习题	401
第 7 章 习题	401
第 8 章 习题	402
第 9 章 习题	402
第 10 章 习题	403
第 11 章 习题	405
第 12 章 习题	406
第 13 章 习题	406
第 14 章 习题	407

第 3 部分 实 验

实验 1 在 JBuilder 集成开发环境中调试 JDK 程序	408
实验目的	408
实验准备	409
实验内容	409
思考与练习题	412

实验2 Java 基本程序设计	413
实验目的	413
实验准备	413
实验内容	413
实验步骤	413
思考与练习题	415
实验3 Java 数组、字符串和排序	415
实验目的	415
实验准备	415
实验内容	415
实验步骤	415
思考与练习题	417
实验4 Java 方法与递归	417
实验目的	417
实验准备	417
实验内容	417
实验步骤	417
思考与练习题	418
实验5 面向对象程序设计	418
实验目的	418
实验准备	418
实验内容	418
实验步骤	419
思考与练习题	420
实验6 Java 异常处理	420
实验目的	420
实验准备	421
实验内容	421
实验步骤	421
思考与练习题	422
实验7 Java 输入/输出处理	422
实验目的	422
实验准备	422
实验内容	422
实验步骤	422
思考与练习题	424
实验8 Java 线程处理	424
实验目的	424
实验准备	424

实验内容	424
实验步骤	424
思考与练习题	425
实验 9 AWT 组件及应用	425
实验目的	425
实验准备	425
实验内容	425
实验步骤	426
思考与练习题	428
实验 10 JBuilder 图形用户界面设计 (一)	428
实验目的	428
实验准备	428
实验内容	429
实验步骤	429
思考与练习题	434
实验 11 JBuilder 图形用户界面设计 (二)	434
实验目的	434
实验准备	434
实验内容	434
实验步骤	435
思考与练习题	438
实验 12 JBuilder 图形用户界面设计 (三)	438
实验目的	438
实验准备	438
实验内容	439
实验步骤	439
思考与练习题	440
实验 13 JBuilder 图形用户界面设计 (四)	441
实验目的	441
实验准备	441
实验内容	441
实验步骤	441
思考与练习题	443
实验 14 JBuilder 数据库设计与应用 (一)	443
实验目的	443
实验准备	443
实验内容	443
实验步骤	443
思考与练习题	444

实验 15 JBuilder 数据库设计与应用（二） 444

 实验目的 444

 实验准备 444

 实验内容 444

 实验步骤 445

 思考与练习题 446

第 4 部分 综合应用实习

1. 实习题目和功能 447

2. 实习目的 447

3. 实习要求 447

4. 设计参考 448

第 1 部分 Java 实用教程

第 1 章 Java 和 JBuilder 9.0 集成开发环境

1.1 Java 语言简介

Java 是一个由 Sun 公司于 1991 年开发而成的新一代编程语言,最初的目的是为家用消费类电子产品开发一个分布式代码系统。为了使整个系统与平台无关,该项目小组的领导人 James Gosling 决定开发一个新的语言,称为 Oak,这就是 Java 语言的前身,不过当时并没有引起人们的注意。后来发现 Oak 已被别人注册,才改名为 Java。Gosling 在设计 Java 时,并不局限于扩充语言机制本身,更侧重于语言所运行的软、硬件环境。他要建立一个系统,这个系统运行于一个巨大的、分布的、异构的网格环境中,完成各电子设备之间的通信与协同工作。Gosling 在设计中采用了虚拟机码 (Virtual Machine Code) 方式,虚拟机运行在一个解释器上,每一个操作系统均有一个解释器。这样,Java 就成了与平台无关的语言。

1994 年下半年,随着 Internet 的迅猛发展,WWW 的应用日益广泛,Gosling 决定用 Java 开发一个实时性较高、可靠安全、有交互功能的新的 Web 浏览器,它不依赖于任何硬件平台和软件平台。新的浏览器称为 HotJava,并于 1995 年 5 月 23 日发表,立即在业界引起了巨大的轰动,Java 的地位也随之而得到肯定。一些著名的计算机公司纷纷购买了 Java 语言的使用权,如 MicroSoft, IBM, Netscape, Novell, Apple, DEC, SGI 等,因此,Java 语言被美国著名杂志 PC Magazine 评为 1995 年十大优秀科技产品,随之大量出现了用 Java 语言编写的软件产品。

Java 语言的发展速度是惊人的,1995 年 3 月 Sun 公司发布了 Java 语言的 Alpha1.0a 2 版本,1996 年 1 月发布了 Java 语言的第一个开发包 JDK v1.0,1997 年 2 月发布了 Java 语言的开发包 JDK v1.1,从此奠定了 Java 语言在计算机语言中的地位。1998 年 12 月 Sun 公司发布 Java 2 平台 JDK v1.2。Java 2 平台是 Java 发展史上的里程碑。1999 年 6 月, Sun 公司重新组织 Java 平台的集成方法,并将企业级应用平台作为 Java 语言发展方向,主要有 3 个成员:

J2ME—Java 2 Micro Edition, 用于嵌入式应用的 Java2 平台。

J2SE—Java 2 Standard Edition, 用于工作站、PC 机的 Java2 标准平台。

J2EE—Java 2 Enterprise Edition, 可扩展的企业级应用的 Java2 平台

1.1.1 Java 语言特点

Java 语言是一个广泛使用的网络编程语言,它简单、面向对象,不依赖于机器的结构,

不受 CPU 和环境的限制, 具有可移植性、鲁棒性、安全性, 并且提供了并发的机制, 具有很高的性能。其次, Java 语言提供了丰富的类库, 使程序设计人员可以方便地建立自己的系统。

(1) 简单性。Java 语言是一种面向对象的语言, 语法规则和 C 类似, 它通过提供最基本的方法来完成指定的任务, 只需理解一些基本的概念, 就可以用它编写出适合于各种情况的应用程序。Java 语言略去了指针、运算符重载、多重继承等模糊的概念, 并且通过实现自动垃圾收集大大简化了程序设计者的内存管理工作。另外, Java 语言对环境要求非常低, 它的基本解释器及类的支持只有几十千字节, 加上标准类库和线程的支持也只有两百多千字节。

(2) 面向对象。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以及动态的接口模型。对象中封装了它的状态变量以及相应的方法, 实现了模块化和信息隐藏; 而类则提供了一类对象的原型, 并且通过继承机制, 子类可以使用父类所提供的方法, 实现代码的复用。

(3) 分布性。Java 语言是面向网络的语言。通过它提供的类库可以处理 TCP/IP 协议, 用户可以通过 URL 地址在网络上很方便地访问其他对象。

(4) 鲁棒性。Java 语言在编译和运行程序时, 都要对可能出现的问题进行检查, 以消除错误的产生。它提供自动垃圾收集来进行内存管理, 防止程序员在管理内存时容易产生的错误。通过集成的面向对象的异常处理机制, 在编译时, Java 语言提示出可能出现但未被处理的异常, 帮助程序员正确地进行选择以防止系统的崩溃。另外, Java 语言在编译时还可捕获类型声明中的许多常见错误, 防止动态运行时不匹配问题的出现。

(5) 安全性。用于网络、分布环境下的 Java 语言必须防止病毒的入侵。Java 语言不支持指针, 一切对内存的访问都必须通过对象的实例变量来实现, 这样就防止程序员使用“特洛伊”木马等欺骗手段访问对象的私有成员, 同时也避免了指针操作中容易产生的错误。

(6) 体系结构中立。Java 解释器生成与体系结构无关的字节码指令, 只要安装了 Java 虚拟机, Java 程序就可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。

(7) 可移植性。与平台无关的特性使 Java 程序可以方便地被移植到网络上的不同机器。同时, Java 的类库中也实现了与不同平台的接口, 使这些类库可以移植。

(8) 解释执行。Java 解释器直接对 Java 字节码进行解释执行。字节码本身携带了许多编译时的信息, 使得连接过程更加简单。

(9) 高性能。和其他解释执行的语言不同, Java 字节码的设计使之能很容易地直接转换成对应于特定 CPU 的机器码, 从而得到较高的性能。

(10) 多线程。多线程机制使应用程序能够并行执行, 而且同步机制保证了对共享数据的正确操作。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, 这样就很容易地实现网络上的实时交互行为。

(11) 动态性。Java 语言的设计使它适合于一个不断发展的环境。在类库中可以自由地加入新的方法和实例变量而不会影响用户程序的执行。并且 Java 通过接口来支持多重继承, 使之比严格的类继承具有更灵活的方式和扩展性。

1.1.2 一个简单的程序

本节通过一个简单的 Java 应用程序, 初步了解 Java 程序编写的基本过程。