

21世纪大学计算机系列教材

Java

大学实用教程(第2版)

耿祥义 张跃平 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY <http://www.phei.com.cn>



JAVA

大学实用教程 (第2版)

清华大学出版社

清华大学出版社

TP312/1522=2

2008

21世纪大学计算机系列教材

Java 大学实用教程 (第2版)

耿祥义 张跃平 编著



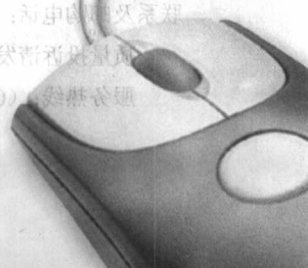
北方工业大学图书馆

C00093259

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING



内 容 简 介

本书是《Java 大学实用教程》的第 2 版, 继续保留原教材的特点——注重教材的可读性和实用性。本书加大了面向对象的知识容量, 对部分例题的内容做了调整, 增加了一些新的知识内容和例题。考虑到 Java 与数据库连接的重要性, 新增一章的内容来讲述 Java 与数据库操作。

全书共分 13 章, 重点讲解了类与对象、类的继承、接口的使用、SDK 1.5 新推出的泛型、字符串与模式匹配、实用类及数据结构、多线程、输入/输出流、基于 SWING 的图形界面用户设计、网络编程、数据库操作、Java Applet 程序设计等内容。本书提供配套的电子课件和教学资源。本书有配套的学习指导书。

本书适合作为高等学校 Java 语言的教材, 也可以作为自学用书。

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 大学实用教程/耿祥义, 张跃平编著. —2 版. —北京: 电子工业出版社, 2008.5

(21 世纪大学计算机系列教材)

ISBN 978-7-121-06255-1

I. J… II. ①耿…②张… 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 (2008) 第 038704 号

策划编辑: 章海涛

责任编辑: 章海涛

印 刷: 北京东光印刷厂

装 订: 三河市皇庄路通装订厂

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

开 本: 787×1 092 1/16 印张: 21.25 字数: 540 千字

印 次: 2008 年 5 月第 1 次印刷

印 数: 5 000 册 定价: 29.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@pei.com.cn, 盗版侵权举报请发邮件至 dbqq@pei.com.cn。

服务热线: (010) 88258888。

总 序

进入 21 世纪, 信息社会发展的脚步越来越快, 对人才的需求也呈现出新的变化趋势。计算机与外语成为新世纪高素质人才必须熟练掌握的工具。大学计算机公共课程也面临新的机遇和挑战, 首先是来自社会和就业市场对人才“知识—能力—素质”要求的挑战; 其次是计算机和相关领域技术及应用快速发展带来的冲击; 最后是普及计算机教育后要求高等计算机教育在教学的“难度—深度—强度”三维同步提高。在这样的大背景下, 大学计算机公共课程在“基础—技术—应用”方面呈现出层次性、通用性和专业需求多样化的特点。我们一直追踪、关注一线教师和专家的卓有成效的课程和教材改革与发展研究, 适时推出了“21 世纪大学计算机系列教材”。

该系列教材在知识结构方面力求覆盖“计算机系统与平台、程序设计与算法、数据分析与信息处理、信息系统开发”四个领域, 内容强调“概念性基础、技术与方法基础、应用技能”三个层次, 第一批教材涉及“大学计算机基础”、“程序设计与算法”、“计算机硬件技术基础”(或“计算机组成与接口技术”)、“数据库技术与应用”、“多媒体技术”和“网络技术与应用”等六门核心课程。同时, 我们也在挖掘其他通用的应用课程教材, 并将陆续推出。我们特别注意到, 高校工科电类专业、理科和工科非电类专业、经管类专业和文史类专业有各自不同的特点, 可以采用“1+X”的课程解决方案, “1”指第一门计算机课程“大学计算机基础”, “X”指适合不同学校和专业特点的其他课程及其组合, 我们的系列教材为此提供了选择的灵活性。

“21 世纪大学计算机系列教材”立足体系创新、知识创新、教学设计和教学模式创新, 全面考虑读者的需求, 努力提升教材的可读性和可用性, 为教学提供尽可能完善的服务。如提供同步的“习题与实验指导”, 一些教材还为教师提供可修改的电子教案、源程序包、教学指导手册或阶段自测题等多种类型的教学服务, 即提供“教材—教辅—课件”教学支持。读者可以通过华信教育资源网站 (<http://www.huaxin.edu.cn> 或 <http://www.hxedu.com.cn>) 了解该系列教材的出版和服务的动态信息。

“21 世纪大学计算机系列教材”的建设得到了很多专家和老师的热情支持, 教材作者来自哈尔滨工业大学、浙江大学、吉林大学、华中科技大学、中国科技大学、中山大学、北京邮电大学、浙江工业大学等高校, 这些课程都是各高校的教改优质课程和精品课程, 体现了作者对课程和教学的探索与创新。希望这套教材的出版能有力地推动大学计算机新课程体系的建立与发展, 同时也能为高等计算机教育带来与时俱进的活力和生机。

由于我们的水平和经验所限, 加之计算机和相关领域技术及应用的发展迅速, 该系列教材一定还存在不少缺点和不足, 欢迎专家和广大读者批评指正。我们会继续努力, 力求不断完善和提高, 以便更好地满足高等计算机教育不断变化的需求。

“21 世纪大学计算机系列教材”编委会

第 2 版前言

本书是《Java 大学实用教程》的第 2 版，继续保留原教材的特点——注重教材的可读性和实用性。

本书加大了面向对象的知识容量，对部分例题的内容做了调整，增加了一些新的知识内容和例题。考虑到 Java 与数据库连接的重要性，新增了一章的内容来讲述 Java 与数据库操作。本书全面地讲解了 Java 的基础内容和编程方法，在内容的深度和广度方面都经过了仔细考虑，对类、对象、继承、接口等重要的基础知识的讲解侧重深度，而对实用类的讲解则侧重广度。通过本书的学习，读者可以掌握 Java 面向对象编程的思想和 Java 在网络编程中的一些重要技术。

全书共分 13 章。

第 1 章主要介绍了 Java 产生的背景和 Java 平台，读者可以了解到 Java 是怎样做到“一次写成，处处运行”的。

第 2 章和第 3 章主要介绍了 Java 的基本数据类型、运算符和控制语句。

第 4 章和第 5 章是本书的重点内容之一，讲述了类、对象、继承、接口以及 Java 语言新增的泛型等重要知识，特别讲述了面向抽象的程序设计思想。

第 6 章讲述了最常用的字符串以及相关的模式匹配的知识。模式匹配问题是很多信息技术经常需要处理的问题之一。

第 7 章讲述了常用的实用类，包括处理日期、数学计算以及数据结构等实用类。

第 8 章讲述了多线程技术，也是很难讲解的一部分内容，本章通过许多有启发的例子来帮助读者理解多线程编程。

第 9 章讲解了 Java 中的输入/输出流技术，其中特别介绍了怎样使用输入/输出流来克隆对象以及 Java 的文件锁技术。

第 10 章是基于 SWING 的 GUI 设计，讲解了常用的组件和容器，对于比较复杂的组件都给出了很实用的例子。

第 11 章讲解了 Java 在网络编程中的一些重要技术，涉及 URL、Socket、InetAddress、DatagramPacket、Broadcast 等重要的网络概念。

第 12 章主要讲解 Java 怎样操作数据库，讲解了预处理、事务处理等重要技术。

第 13 章主要讲解 Java Applet 的运行原理以及在网络中的角色。

本书的例题全部在 JDK 1.6 环境下编译通过。每章都有问答题和作业题，通过回答问答题可以使读者加深知识的理解。理解该内容后，读者完全有能力独立地完成作业题。

本书有配套的电子课件，请教师登录华信教育资源网站（<http://www.huaxin.edu.cn> 或者 <http://www.hxedu.com.cn>），注册之后进行下载，如有问题请发邮件咨询 unicode@phei.com.cn。

本书有配套的辅导书《Java 大学实用教程学习指导（第 2 版）》，除了按照主教材的章节配备实验指导外，还增加了一个综合实验——扫雷游戏。每章的实验指导由实验内容和知识扩展两部分内容组成，学生可按照实验的要求上机编写程序。每个实验都提供了程序

模板，学生完成实验后需填写实验报告。知识扩展是对实验内容的一个补充，结合实例讲解主教材未能涉及到的一些知识或已学知识的深入讨论。综合实验的目的是为了综合运用所学知识设计一个完整的小软件。该书不提供主教材“回答问题”和“作业”的答案（教师可登录到华信教育资源网站 <http://www.hxedu.com.cn> 下载答案），学生可以参考主教材或查找资料进行解答。

希望本书能对读者学习 Java 有所帮助，并请读者批评指正（xygeng0629@sina.com）。

作者

作者简介

耿祥义，1995 年中国科学技术大学博士毕业，获理学博士学位。1997 年从中山大学博士后流动站出站。现任大连交通大学教授，具有多年从事 Java 语言教学的经验，已编写出版多部教材。

张跃平，现任大连交通大学讲师，具有多年从事 Java 语言教学经验。

目 录

第 1 章 Java 语言概述	1
1.1 Java 语言的诞生	2
1.2 学习 Java 的必要性	2
1.3 Java 的特点	3
1.4 Java 与 C/C++ 之关系	5
1.5 Java 运行平台	6
1.6 Java 程序开发	8
1.7 JDK 1.6 编译器的兼容性	13
问答题	13
作业题	13
第 2 章 基本数据类型和数组	14
2.1 标识符和关键字	15
2.2 基本数据类型	15
2.3 基本数据类型的转换	18
2.4 数据的输入与输出	19
2.5 数组	21
问答题	23
作业题	24
第 3 章 运算符、表达式与语句	25
3.1 算术运算符和算术表达式	26
3.2 关系运算符和关系表达式	27
3.3 逻辑运算符和逻辑表达式	27
3.4 赋值运算符和赋值表达式	28
3.5 移位运算符	28
3.6 位运算符	29
3.7 条件运算符	31
3.8 instanceof 运算符	31
3.9 一般表达式	31
3.10 语句概述	31
3.11 分支语句	32

3.12 循环语句	35
3.13 跳转语句	38
问答题	39
作业题	41
第4章 类与对象	42
4.1 面向对象编程	43
4.2 类声明和类体	43
4.3 类体的构成	44
4.4 构造方法与对象的创建	46
4.5 对象的引用与实体	50
4.6 成员变量	51
4.7 方法	53
4.8 方法重载	58
4.9 关键字 this	59
4.10 包	61
4.11 import 语句	63
4.12 访问权限	66
4.13 对象的组合	70
4.14 基本类型数据的类包装	73
4.15 反编译和文档生成器	75
4.16 JAR 文件	75
问答题	77
作业题	78
第5章 继承、接口与泛型	79
5.1 子类与父类	80
5.2 子类的继承性	80
5.3 子类对象的构造过程	82
5.4 成员变量隐藏和方法重写	83
5.5 关键字 super	86
5.6 final 类和 final 方法	88
5.7 对象的上转型对象	88
5.8 继承与多态	90
5.9 abstract 类	91
5.10 面向抽象	91
5.11 接口	94
5.12 接口回调	97
5.13 面向接口	98

5.14	抽象类与接口的比较	100
5.15	内部类	101
5.16	匿名类	102
5.17	异常类	104
5.18	泛型类	106
	问答题	109
	作业题	111
第6章	字符串和正则表达式	112
6.1	String 类	113
6.2	StringBuffer 类	119
6.3	StringTokenizer 类	120
6.4	正则表达式与模式匹配	121
6.5	正则表达式与字符串分解	126
	问答题	127
	作业题	127
第7章	常用实用类	128
7.1	Date 类	129
7.2	Calendar 类	130
7.3	Math 类与 BigInteger 类	132
7.4	LinkedList<E>泛型类	135
7.5	HashSet<E>泛型类	139
7.6	HashMap<K,V>泛型类	142
7.7	TreeSet<E>泛型类	144
7.8	TreeMap<K,V>泛型类	146
7.9	Stack<E>泛型类	147
	问答题	148
	作业题	149
第8章	线程	150
8.1	Java 中的线程	151
8.2	线程的生命周期	152
8.3	线程的优先级与调度管理	156
8.4	Thread 的子类创建线程	156
8.5	Runnable 接口	157
8.6	线程的常用方法	161
8.7	线程同步	164
8.8	在同步方法中使用 wait()、notify()和 notifyAll()	165

8.9 挂起、恢复和终止线程	167
8.10 线程联合	170
8.11 守护线程	171
问答题	172
作业题	172
第9章 输入/输出流	173
9.1 文件	174
9.2 文件字节流	177
9.3 文件字符流	178
9.4 从键盘读取数据	180
9.5 缓冲流	180
9.6 数组流	182
9.7 字符串流	183
9.8 数据流	184
9.9 对象流	185
9.10 序列化和对象克隆	187
9.11 随机读写流	188
9.12 文件锁	191
问答题	192
作业题	192
第10章 基于 SWING 的图形用户界面设计	193
10.1 AWT 组件与 SWING 组件概述	194
10.2 JFrame 窗体	195
10.3 菜单组件	196
10.4 布局设计	198
10.5 中间容器	203
10.6 文本组件	205
10.7 按钮与标签组件	213
10.8 复选框与单选按钮组件	215
10.9 列表组件	217
10.10 表格组件	218
10.11 树组件	221
10.12 进度条组件	223
10.13 组件常用方法	226
10.14 窗口事件	229
10.15 鼠标事件	231
10.16 焦点事件	236

10.17	键盘事件	236
10.18	AWT 线程	239
10.19	计时器	241
10.20	MVC 设计模式	243
10.21	播放音频	245
10.22	对话框	247
10.23	多文档界面	253
10.24	发布应用程序	257
	问答题	257
	作业题	258
第 11 章 Java 中的网络编程		259
11.1	URL 类	260
11.2	读取 URL 中的资源	261
11.3	显示 URL 资源中的 HTML 文件	262
11.4	处理超链接	264
11.5	InetAddress 类	266
11.6	套接字 Socket	267
11.7	使用多线程处理套接字连接	270
11.8	UDP 数据报	274
11.9	广播数据报	278
	问答题	281
	作业题	281
第 12 章 Java 与数据库操作		282
12.1	JDBC 简介	283
12.2	Microsoft Access 数据库	283
12.3	JDBC-ODBC 桥接器	284
12.4	查询操作	287
12.5	更新、插入与删除操作	292
12.6	用结果集更新数据库中的表	293
12.7	CachedRowSetImpl 类	295
12.8	预处理语句	296
12.9	事务	299
12.10	使用 JTable 组件操作表	300
12.11	SQL Server 2000 数据库	306
12.12	使用纯 Java 数据库驱动程序	307
	问答题	309
	作业题	309

第 13 章 Java Applet	310
13.1 Java Applet 的运行原理	311
13.2 在 Java Applet 中播放声音	314
13.3 在网页向 Java Applet 传值	315
13.4 在 Java Applet 中使用组件	315
13.5 在 Java Applet 中绘制图形	320
13.6 在 Java Applet 中绘制图像	322
13.7 在 Java Applet 中播放幻灯片	323
13.8 Java Applet 网络聊天室	324
问答题	328
作业题	328

第 1 章

Java 语言概述

本章导读

- Java 语言的诞生
- 学习 Java 的必要性
- Java 语言的特点以及与 C/C++ 之关系
- Java 程序的运行平台和程序开发
- JDK 1.6 编译器的新规定

1.1 Java 语言的诞生

Java 诞生于 1995 年, 是 Sun 公司组织开发的一门编程语言, 主要贡献者是 James Gosling。开发 Java 语言的动力源于对独立于平台的需要, 即用这种语言编写的程序不会因为芯片的变化而发生无法运行或出现运行错误的情况。当时, C 语言已无法满足人们的这一愿望, 因为 C 语言总是针对特定的芯片将源程序编译为机器码, 该机器码的运行就与特定的芯片指令有关, 在其他类型的芯片上可能无法运行或出现运行错误。芯片制造者、芯片使用者和软件编写者往往隶属于不同的公司。芯片制造者会不断地推出性能更好而且价格更便宜的新型芯片。当有一种性价比更高的芯片出现时, 芯片使用者就可能立即使用新的芯片, 这些芯片可能安装在各种计算机或电器设备上。如果不能保证程序在新的芯片上正确运行, 就可能出现难以发现的错误, 最终导致严重的后果, 可能还会引起设备的毁坏等灾难性后果。所以, 软件编写者必须针对新的芯片重新编译源程序, 甚至还需要对源程序进行必要的修改, 这是令软件开发者最头痛的工作, 还可能由于各种原因 (如工作量的巨大等) 而导致工作无法完成。

1990 年, Sun 公司成立了由 James Gosling 领导的开发小组, 开始致力于开发一种可移植的、跨平台的语言, 该语言能生成正确运行于各种操作系统、适应各种 CPU 芯片的代码。他们的精心钻研和努力促成了 Java 语言的诞生。Java 的快速发展得益于 Internet 和 Web 的出现, Internet 上有各种不同的计算机, 它们可能使用完全不同的操作系统和 CPU 芯片, 但仍希望运行相同的程序, Java 的出现标志着真正的分布式系统的到来。

James Gosling 的办公室外面有一棵大橡树, 他最初将 Java 语言命名为 oak, 后来发现已经有一种计算机语言的名字叫 oak, 最后决定为这种新的语言起名为 Java, 其寓意是为世人端上一杯热咖啡。“Java” 是印度尼西亚一个盛产咖啡的岛屿, 中文译名是“爪哇”。

1.2 学习 Java 的必要性

Java 不仅可以用来开发大型的桌面应用程序, 而且特别适合于 Internet 的应用开发。目前, Java 语言不仅是一门正在被广泛使用的编程语言, 而且已成为软件设计开发者应当掌握的一门基础语言。Java 语言面向对象编程, 并涉及网络、多线程等重要的基础知识, 而且很多新的技术领域都涉及到了 Java 语言。因此, 学习和掌握 Java 已成为共识, 国内外许多大学已将 Java 语言列入了本科教学计划。IT 行业对 Java 人才的需求正在不断增长, 一些软件公司对其开发人员周期性地进行了 Java 的基础培训工作。在 IT 行业发达的北美, 有将近 60% 的软件开发人员使用 Java 完成他们的工作。Evans Data 公司在 2002 年做的一项调查中发现: 在北美, Java 的使用率已经接近 C/C++。

2003 年, James Gosling 曾来北京, 与中国的 IT 人士进行了交流。以下是对话的节选:

问: 在近几年的发展过程中, 很多编程语言都逐渐消失, 而 Java 语言却越来越火热, 请问其中的原因是什么?

James Gosling: 我认为, 很多的编程语言在发展中并不是消失, 而是转移到了其他领域中去, 而 Java 的经久不衰, 取决于 Java 的技术基础。如果你问程序员, 为什么会选择 Java, 他会告诉你, Java 提供了多种功能, 提供了方便的平台, 是个足以吸引人的工具。我认为, 推动 Java 最主要的因素是网络, Java 是以网络应用为基础的开发工具, 这是它的强处。

问：在传统计算机领域中，Java 并不是十分大的平台，如 PC。而在其他领域，如移动领域，Java 发展迅速，Java 的未来发展方向是什么？

James Gosling：在 PC 领域，我并不认为 Java 不够强大。在 PC 领域，Java 有很多应用，这是表面上看不到的，这主要是微软花了大力气避免用户看到，实际上 Java 应用很广泛，如人工智能游戏。在其他领域，Java 更是应用广泛，如汽车、铁路机车上的即时控制系统，Java 也广泛应用于军用方面。

问：大家都尊称您为 Java 之父，您能不能跟大家分享一下您在 Java 事业中最深的感受是什么？

James Gosling：当看到 Java 的客户通过 Java 完成了很多神奇的工作，例如，看到夏威夷火山上的观测台使用 Java 控制望远镜，看到荷兰健康医疗组织使用 Java 解决了保护隐私问题等，那真是一种奇妙的感觉。

1.3 Java 的特点

1. 平台无关性

Java 语言和其他语言相比，最大的优势就是能够在所有的计算机上运行，这也是 Java 风靡全球的主要原因。Java 之所以能做到这一点，是因为 Java 可以在计算机的操作系统之上提供了一个 Java 运行环境，即一个运行 Java 程序的平台，这个平台由 Java 虚拟机 (Java Virtual Machine, JVM)、类库以及一些核心文件组成。

Java 语言确实具备了“一旦写成，处处可用”的特点，为了能很好地理解 Java 不依赖于平台的特点，有必要简单回顾一下编程语言的发展过程。尽管正在使用的计算机语言有很多种，但它们可分为三大类：机器语言、汇编语言和高级语言。

计算机处理信息的早期语言是所谓的机器语言。机器语言是由二进制代码书写的且能被计算机直接识别的一种语言，这种语言中的指令都是由 0 和 1 组成的序列。一个这样的序列称为一条机器指令。每一条机器指令，都是由计算机的硬件设计定义好的。比如，某种型号的计算机用 8 位二进制信息 10101010 表示一次加法，以 10010011 表示一次减法等。因此，用机器语言编写的代码是与机器相关的，即某种机器语言仅可用在某特定型号的计算机上。用机器语言进行程序设计既烦琐、费时又难以识别。而且，同样的任务，人们要针对不同型号的计算机分别编写指令。因此，使用机器语言编程也称为面向机器编程。20 世纪 50 年代出现了汇编语言，在编写指令时，它用一些简单的容易记忆的符号来代替二进制指令。但汇编语言仍是面向机器的，需针对不同的机器来编写不同的代码。习惯上，称机器语言、汇编语言为低级语言。

20 世纪 60 年代，人们研制出了高级语言，如 C 语言、FORTRAN 语言等。高级语言的语法更接近人们的自然语言，人们只要按照具体语言的语法要求去编写“源文件”就可以了。所谓源文件，就是根据这门语言的语法编写的具有一定扩展名的文件，如 C 语言编写的源文件的扩展名是 .c，FORTRAN 语言编写的源文件的扩展名是 .for 等。但是，无论使用哪种高级语言编写的源文件，计算机都不能直接执行，因为计算机只能直接识别和执行机器指令。因此，必须把源文件转换成机器指令，然后计算机去执行相应的机器指令。

将高级语言编写的源程序转化成机器指令，经常使用下列两种方式。

① 编译方式 (Compilation): 针对当前的芯片, 将源程序全部翻译成机器指令 (称为目标程序), 再将目标程序交给计算机执行。

② 解释方式 (Interpretation): 这种方式不产生整个目标程序, 而是根据当前的机器处理芯片, 边翻译边执行, 翻译一句执行一句。

无论哪种高级语言, 都必须提供相应的编译器或解释器。例如, C 语言采用的是编译方式, 即针对特定的 CPU 芯片对源文件进行编译, 生成机器代码。采用编译方式生成的目标程序与特定的计算机有关, 一旦环境有所变化就可能需要修改源程序并针对新的环境重新编译, 生成新的目标程序。

Java 语言的处理方式既不是第①种也不是第②种, Java 语言不针对特定的 CPU 芯片进行编译, Java 提供的编译器把源程序编译成称为“字节码”的一个中间代码。字节码是很接近机器码的二进制文件, 不能被机器直接识别, 但是可以在提供了 Java 运行环境平台的任何系统上被解释执行。JVM 是 Java 运行环境的重要组成部分, 正是 JVM 中的解释器负责将字节码解释成本地的机器码。因此, Java 本质上是解释执行的程序, 当字节码加载到内存中之后, 再由 Java 运行环境平台中的 Java 解释器对字节码按解释方式执行, 即翻译一句执行一句, 不产生整个机器代码程序。翻译过程如果不出现错误, 就一直进行到全部执行完毕, 否则将在错误处停止执行。同一个程序, 如果是解释执行的, 那么它的运行速度通常会比被编译方式的运行速度慢一些。但是, 对 Java 来说, 两者的差别不太大, Java 的字节码经过仔细设计, 很容易便能使用 JIT (Just In Time, 即时编译) 技术将字节码直接转化成高性能的本地机器码。Sun 公司在 Java 2 发行版中提供了一个字节码编译器 JIT, 它是 Java 运行环境平台的一部分。Java 运行系统在提供 JIT 的同时仍具有平台独立性, 因而“高效且跨平台”对 Java 来说不再矛盾。

Sun 公司提供的各种 Java 编译器总是按照统一的标准把源程序编译成字节码文件, 而且 Sun 公司已经为广大的硬件提供了各种带 JVM 的 Java 运行环境平台, 这些平台都可以解释这样的字节码文件。如果把 Java 的源文件比作“汉语”的话, 字节码就相当于“世界语”, 世界语不和具体的“国家”有关, 只要这个“国家”提供了“翻译”, 就可以快速地把世界语翻成本地语言。

2. 面向对象

面向对象编程是一种先进的编程思想, 更加容易解决复杂的问题。面向对象编程主要体现在下列三个特性上。

(1) 封装

面向对象编程的核心思想之一就是数据和对数据的操作封装在一起。通过抽象, 即从具体的实例中抽取共同的性质形成一般的概念, 如类的概念。人们经常谈到的机动车类就是从具体的实例中抽取共同的属性和功能形成的一个概念, 那么一个具体的轿车就是机动车类的一个实例, 即对象。一个对象将自己的数据和对这些数据的操作合理、有效地封装在一起, 如每辆轿车调用“加大油门”改变的都是自己的运行速度。

(2) 继承

继承体现了一种先进的编程模式。子类可以继承父类的属性和功能, 即继承了父类的数据和数据上的操作, 同时又可以增加子类独有的数据和数据上的操作。比如, “人类”自然继