

本书第二版被评为
普通高等教育“十一五”国家级规划教材

新世纪计算机基础教育丛书

丛书主编 谭浩强

Java程序设计

(第三版)

辛运伟 饶一梅 马素霞 编著



清华大学出版社

013028386

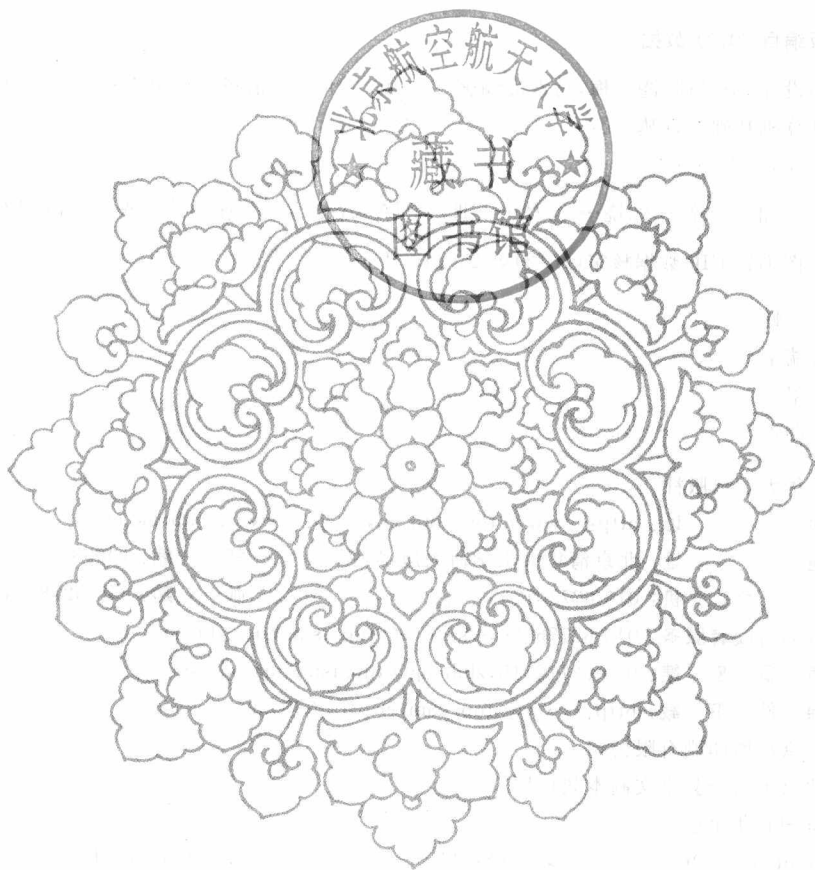
TP312JA
133-3

新世纪计算机基础教育丛书

Java程序设计

(第三版)

辛运伟 饶一梅 马素霞 编著



清华大学出版社



北航

C1635058

TP312JA

133-3

883830017

内 容 简 介

本书从 Java 语言的基本特点入手,全面介绍 Java 语言的基本概念和编程方法,并深入介绍 Java 语言的高级特性。书中内容涉及 Java 语言中的基本语法、数据类型、类、异常、界面设计、小应用程序、I/O 数据流、线程及网络功能等,基本覆盖了 Java 语言的大部分实用技术,是进一步使用 Java 语言进行技术开发的基础。

本书内容详尽,实例丰富,在每章最后均列出了若干习题,以方便教学。本书适合作为高等学校教材,也可作为专业人员的参考书。

与本书配套的《Java 程序设计题解与上机指导》(第三版)中提供了本书的习题解答及有关实验。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java 程序设计/辛运伟,饶一梅,马素霞编著. --3 版. --北京:清华大学出版社,2013.4

新世纪计算机基础教育丛书

ISBN 978-7-302-30807-2

I. ①J… II. ①辛… ②饶… ③马… III. ①JAVA 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 287443 号

责任编辑:焦 虹

封面设计:傅瑞学

责任校对:时翠兰

责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:22.25

字 数:513 千字

版 次:2001 年 9 月第 1 版 2013 年 4 月第 3 版

印 次:2013 年 4 月第 1 次印刷

印 数:1~3000

定 价:35.00 元

产品编号:050526-01

现

代科学技术的飞速发展,改变了世界,也改变了人类的生活。作为新世纪的大学生,应当站在时代发展的前列,掌握现代科学技术知识,调整自己的知识结构和能力结构,以适应社会发展的要求。新世纪需要具有丰富的现代科学知识,能够独立完成面临的任務,充满活力,有创新意识的新型人才。

掌握计算机知识和应用,无疑是培养新型人才的一个重要环节。现在计算机技术已深入到人类生活的各个角落,与其他学科紧密结合,成为推动各学科飞速发展的有力的催化剂。无论学什么专业的学生,都必须具备计算机的基础知识和应用能力。计算机既是现代科学技术的结晶,又是大众化的工具。学习计算机知识,不仅能够掌握有关知识,而且能培养人们的信息素养。这是高等学校全面素质教育中极为重要的一部分。

高校计算机基础教育应当遵循的理念是:面向应用需要;采用多种模式;启发自主学习;重视实践训练;加强创新意识;树立团队精神,培养信息素养。

计算机应用人才队伍由两部分人组成:一部分是计算机专业出身的计算机专业人才,他们是计算机应用人才队伍中的骨干力量;另一部分是各行各业中应用计算机的人员。这后一部分人一般并非计算机专业毕业,他们人数众多,既熟悉自己所从事的专业,又掌握计算机的应用知识,善于用计算机作为工具解决本领域中的任务。他们是计算机应用人才队伍中的基本力量。事实上,大部分应用软件都是由非计算机专业出身的计算机应用人员研制的,他们具有的这个优势是其他人难以代替的。从这个事实可以看到在非计算机专业中深入进行计算机教育的必要性。

非计算机专业中的计算机教育,无论目的、内容、教学体系、教材、教学方法等各方面都与计算机专业有很大的不同,绝不能照搬计算机专业的模式和做法。全国高等院校计算机基础教育研究会自1984年成立以来,始终不渝地探索高校计算机基础教育的特点和规律。2004年,全国高等院校计算机基础教育研究会与清华大学出版社共同推出了《中国高等院校计算机基础教育课程体系2004》(简称CFC2004);2006年、2008年又共同推出了《中国高等院校计算机基础教育课程体系2006》(简称CFC2006)及《中国高等院校计算机基础教育课程体系2008》(简称CFC2008),由清华大学出版社正式出版发行。

1988年起,我们根据教学实际的需要,组织编写了《计算机基础教育丛书》,邀请有丰富教学经验的专家、学者先后编写了多种教材,由清华大

学出版社出版。丛书出版后,迅速受到广大高校师生的欢迎,对高等学校的计算机基础教育起了积极的推动作用。广大读者反映这套教材定位准确,内容丰富,通俗易懂,符合大学生的特点。

1999年,根据新世纪的需要,在原有基础上组织出版了《新世纪计算机基础教育丛书》。由于内容符合需要,质量较高,被许多高校选为教材。丛书总发行量1000多万册,这在国内是罕见的。最近,我们又对丛书作了进一步的修订,根据发展的需要,增加了新的书目和内容。本丛书有以下特点:

(1) 内容新颖。根据21世纪的需要,重新确定丛书的内容,以符合计算机科学技术的发展和教学改革的要求。本丛书除保留了原丛书中经过实践考验且深受群众欢迎的优秀教材外,还编写了许多新的教材。在这些教材中反映了近年来迅速得到推广应用的一些计算机新技术,以后还将根据发展不断补充新的内容。

(2) 适合不同学校组织教学的需要。本丛书采用模块形式,提供了各种课程的教材,内容覆盖了高校计算机基础教育的各个方面。丛书中既有理工类专业的教材,也有文科和经济类专业的教材;既有必修课的教材,也包括一些选修课的教材。各类学校都可以从中选择到合适的教材。

(3) 符合初学者的特点。本丛书针对初学者的特点,以应用为目的,以应用为出发点,强调实用性。本丛书的作者都是长期在第一线从事高校计算机基础教育的教师,对学生的基础、特点和认识规律有深入的研究,在教学实践中积累了丰富的经验。可以说,每一本教材都是他们长期教学经验的总结。在教材的写法上,既注意概念的严谨和清晰,又特别注意采用读者容易理解的方法阐明看似深奥难懂的问题,做到例题丰富,通俗易懂,便于自学。这一点是本丛书一个十分重要的特点。

(4) 采用多样化的形式。除了教材这一基本形式外,有些教材还配有习题解答和上机指导,并提供电子教案。

总之,本丛书的指导思想是内容新颖、概念清晰、实用性强、通俗易懂、教材配套。简单概括为:“新颖、清晰、实用、通俗、配套”。我们经过多年实践形成的这一套行之有效的创作风格,相信会受到广大读者的欢迎。

本丛书多年来得到了各方面人士的指导、支持和帮助,尤其是得到了全国高等院校计算机基础教育研究会的各位专家和各高校老师们的支持和帮助,我们在此表示由衷的感谢。本丛书肯定有不足之处,希望得到广大读者的批评指正。

欢迎访问谭浩强网站: <http://www.tanhoqiang.com>。

丛 书 主 编
全国高等院校计算机基础教育研究会荣誉会长
谭 浩 强

前言

本书自 2001 年第一版问世,到现在已经超过了 10 年。从这一点可以看出,Java 语言仍具有足够的吸引力,其生命力并没有随时间的推移而有丝毫的减弱,学习 Java 语言的人越来越多。同时,由于技术的发展,Java 语言的适用环境更加广泛,新的应用层出不穷。

Java 语言之所以这样长盛不衰,一是因为其众多技术特点与现今的应用十分合拍,可以覆盖大部分需求;二是因为不断有新版本问世,完善了 Java 语言自身的功能。有了这两点,Java 语言成为程序员首选的程序设计开发工具就顺理成章了。

现在越来越多的高等学校开设了 Java 语言这门课程,国内外已经出版了较多的教材。针对 Java 语言的教学目的,结合国内教学的特点,我们重新编写了《Java 程序设计》一书,并配套编写了《Java 程序设计题解与上机指导》。承蒙读者的厚爱,这本书的前几版已经被很多所大学选作教材。

第三版在结构上与第二版一致,为了方便初学者调试程序,增加了简单输入输出功能的介绍。另外,为了增加图形用户界面的观赏性,介绍了绘图功能及颜色的选择。有了这些内容,读者编写的程序将会更加美观。

本书从 Java 语言的基本特点入手,详细介绍 Java 语言的基本概念和编程方法,帮助读者深入了解 Java 的高级特性。本书共分为 12 章,涉及 Java 语言中的基本语法、数据类型、类、异常、界面设计、小应用程序、I/O 数据流、线程及网络功能等内容。这些内容基本覆盖了 Java 语言的大部分实用技术,是进一步使用 Java 语言进行技术开发的基础。愿本书能成为读者进入 Java 殿堂的铺路石。

作为教材,在每章的最后列出了若干习题,供读者参考。本书不仅适合 Java 语言的初学者使用,也可作为专业人员的参考书。

计算机技术是不断发展、不断完善的技术,Java 语言也是如此。在本书出版的过程中,Java 语言仍没有停止它的完善过程。本书中有些内容是以目前的版本为标准,当推出更高版本的 Java 时,读者应参考这些新标准。

在本书的编写过程中,我们得到了南开大学信息技术科学学院卢桂章教授、陈有祺教授、刘璟教授、周玉龙教授、朱耀庭教授等的亲切关怀和

悉心指导,在此表示深深的感谢;我们也非常感谢清华大学出版社为我们提供了这个机会,同时也感谢读者在众多的 Java 语言的参考书中选中了本书。

本书由辛运伟、饶一梅、马素霞编写。由于作者水平有限,书中难免有错误和不妥之处,恳请广大读者特别是同行专家批评指正。

编者

于南开园

目 录

1

概述

1.1	什么是 Java 语言	1
1.1.1	Java 语言的特点	2
1.1.2	Java 的三层架构	3
1.1.3	Java 语言的目标	4
1.1.4	Java 虚拟机	4
1.1.5	垃圾收集	5
1.1.6	代码安全	5
1.2	一个基本的 Java 应用程序	8
1.2.1	开发环境的安装	8
1.2.2	Java 应用程序	9
1.3	程序的编译和运行	11
1.3.1	编译	11
1.3.2	运行	11
1.4	常见错误	12
1.4.1	编译时错误	12
1.4.2	运行时错误	13
1.5	使用 Java 核心 API 文档	14
	习题	17

2

标识符和数据类型

2.1	Java 的基本语法单位	18
2.1.1	空白、注释及语句	18
2.1.2	关键字	20
2.1.3	标识符	20
2.2	Java 编码体例	21
2.3	Java 的基本数据类型	22
2.3.1	基本数据类型	22

2.3.2	类型转换	24
2.3.3	变量、说明和赋值	25
2.4	复合数据类型	26
2.4.1	概述	26
2.4.2	Java 的复合数据类型	27
2.5	类和对象的初步介绍	27
2.5.1	Java 中的面向对象技术	27
2.5.2	Java 中的类定义	29
2.5.3	Java 中与 OOP 有关的关键字	30
2.5.4	类定义示例	33
2.5.5	创建一个对象	35
2.5.6	引用变量的赋值	38
2.5.7	自动初始化和 null 引用值	39
2.5.8	术语概述	39
	习题	40



表达式和流程控制语句

3.1	表达式	42
3.1.1	操作数	42
3.1.2	运算符	46
3.1.3	表达式的提升和转换	48
3.1.4	数学函数	51
3.2	流控制	51
3.2.1	表达式语句	51
3.2.2	块	52
3.2.3	分支语句	52
3.2.4	循环语句	57
3.2.5	break 与 continue 语句	60
3.2.6	注释语句	62
3.3	简单的输入输出	63
	习题	66



数组、向量和字符串

4.1	数组	69
4.1.1	数组说明	69

4.1.2	创建数组	70
4.1.3	数组边界	73
4.1.4	数组元素的引用	73
4.1.5	多维数组	74
4.1.6	数组复制	79
4.2	Vector 类	80
4.2.1	概述	81
4.2.2	Vector 类的构造方法	81
4.2.3	Vector 类对象的操作	82
4.2.4	Vector 类中的其他方法	83
4.2.5	Vector 类的使用举例	84
4.3	字符串类型	86
4.3.1	字符串简述	86
4.3.2	字符串说明及初始化	86
4.3.3	字符串处理	86
4.3.4	几个特殊处理	88
	习题	90



进一步讨论对象和类

5.1	抽象数据类型	92
5.1.1	概述	92
5.1.2	定义方法	93
5.1.3	按值传送	96
5.1.4	重载方法名	98
5.2	对象的构造和初始化	99
5.2.1	显式成员初始化	99
5.2.2	构造方法	99
5.2.3	默认的构造方法	101
5.2.4	构造方法重载	101
5.2.5	finalize() 方法	102
5.3	this 引用	103
5.4	子类	103
5.4.1	“is a”关系	104
5.4.2	extends 关键字	105
5.4.3	单重继承	106

5.4.4	多态性	108
5.4.5	方法自变量和异类集合	108
5.4.6	instanceof 运算符	109
5.4.7	转换对象	110
5.5	方法重写	111
5.5.1	方法重写示例	111
5.5.2	应用重写的规则	115
5.5.3	父类构造方法调用	117
5.6	Java 包	118
5.6.1	Java 包的概念	118
5.6.2	import 语句	119
5.6.3	目录层次关系及 classpath 环境变量	120
5.6.4	访问权限与数据隐藏	122
5.6.5	封装	123
5.7	类成员	124
5.7.1	类变量	124
5.7.2	类方法	126
5.8	关键字 final	128
5.8.1	终极类	128
5.8.2	终极方法	129
5.8.3	终极变量	129
5.9	抽象类	131
5.10	接口	133
5.10.1	接口的定义	133
5.10.2	接口的实现	134
5.11	内部类	137
5.11.1	内部类的概念	137
5.11.2	匿名类	139
5.11.3	内部类的工作方式	140
5.12	包装类	141
5.13	过时信息	142
	习题	145



Java 语言中的异常

6.1	异常	147
6.2	异常示例	150

6.3	异常处理	150
6.3.1	try, catch 和 finally 语句	151
6.3.2	再讨论前面的示例	151
6.3.3	公共异常	152
6.4	异常分类	153
6.5	抛出	154
6.6	创建自己的异常	154
	习题	156



Java 的图形用户界面设计

7.1	AWT 与 Swing	157
7.2	容器	158
7.2.1	顶层容器	158
7.2.2	使用 JFrame 创建一个应用程序	159
7.2.3	内容窗格	159
7.2.4	面板	161
7.3	布局	162
7.3.1	一个简单的例子	162
7.3.2	FlowLayout 布局管理器	164
7.3.3	BorderLayout 布局管理器	166
7.3.4	GridLayout 布局管理器	167
7.3.5	CardLayout 布局管理器	169
7.3.6	BoxLayout 布局管理器	171
7.3.7	其他布局管理器	176
7.4	事件处理	177
7.4.1	事件处理模型	177
7.4.2	事件的种类	179
7.4.3	一个较复杂的例子	181
7.4.4	多监听程序	183
7.4.5	事件适配器	183
	习题	185



Swing 组件

8.1	按钮	187
8.1.1	普通按钮	187

8.1.2	切换按钮、复选按钮及单选按钮	192
8.2	标签	197
8.3	组合框	200
8.4	列表	203
8.5	文本组件	208
8.5.1	文本域	209
8.5.2	文本区	213
8.6	菜单组件	216
8.6.1	菜单栏	216
8.6.2	菜单	217
8.6.3	菜单项	217
8.6.4	复选菜单项和单选菜单项	219
8.6.5	弹出式菜单	222
8.7	对话框、标准对话框与文件对话框	225
8.7.1	对话框	225
8.7.2	标准对话框	226
8.7.3	文件对话框	231
8.8	控制组件外观	234
8.8.1	颜色	234
8.8.2	字体	235
8.8.3	绘图	236
习题		240



Java Applet

9.1	编写 Applet	244
9.2	Applet 的方法和 Applet 的生命周期	246
9.3	Applet 的运行	247
9.3.1	用于显示 Applet 的方法	247
9.3.2	appletviewer	249
9.3.3	HTML 与<applet>标记	250
9.3.4	Applet 参数的读取	251
9.3.5	Applet 与 URL	253
9.4	在 Applet 中的多媒体处理	254
9.4.1	在 Applet 中显示图像	254
9.4.2	在 Applet 中播放声音	255

9.5	Applet 的事件处理	257
9.6	Applet 与普通应用程序的结合	258
	习题	260



Java 数据流

10.1	数据流的基本概念	261
10.1.1	输入数据流	262
10.1.2	输出数据流	263
10.2	基本字节数据流类	264
10.2.1	文件数据流	264
10.2.2	过滤流	266
10.2.3	管道数据流	268
10.2.4	对象流	269
10.2.5	可持久化	270
10.3	基本字符流	273
10.3.1	读者和写者	273
10.3.2	缓冲区读者和缓冲区写者	275
10.4	文件的处理	279
10.4.1	File 类	279
10.4.2	随机访问文件	281
	习题	282



线程

11.1	线程和多线程	284
11.1.1	线程的概念	284
11.1.2	线程的结构	285
11.2	线程的状态	286
11.3	创建线程	287
11.3.1	创建线程的方法一——继承 Thread 类	288
11.3.2	创建线程的方法二——实现 Runnable 接口	290
11.3.3	关于两种创建线程方法的讨论	292
11.4	线程的启动	293
11.5	线程的调度	293

11.6	线程的基本控制	295
11.6.1	结束线程	295
11.6.2	检查线程	296
11.6.3	挂起线程	296
11.7	同步问题	298
11.7.1	线程间的通信	298
11.7.2	线程间的资源互斥共享	300
11.7.3	对象的锁定标志	302
11.7.4	同步方法	303
11.8	死锁	304
11.9	线程交互——wait()和 notify()	306
11.9.1	问题的提出	306
11.9.2	解决方法	307
11.9.3	守护线程	308
11.9.4	综合应用	309
	习题	314



Java 的网络功能

12.1	概述	316
12.2	使用 InetAddress	318
12.3	统一资源定位器	319
12.3.1	URL 的概念	319
12.3.2	URL 的构造方法	320
12.3.3	与 URL 相关的异常	321
12.3.4	获取 URL 对象属性	321
12.3.5	读入 URL 数据	322
12.4	Socket 接口	325
12.4.1	Socket 的基本概念	326
12.4.2	Socket 通信的基本步骤	328
12.4.3	Socket 通信的程序设计	328
12.5	Java 编程综合实例	330
	习题	338
	参考文献	339

第1章 概 述

1.1 什么是 Java 语言

1991 年,美国 Sun Microsystems 公司的 Jame Gosling、Bill Joe 等人为在电视、烤箱等家用消费类电子产品上进行交互式操作而开发了 Oak 语言,之后 Sun 的开发人员将 Oak 改为了 Java。互联网的出现给 Java 带来了生机。因此,在 1995 年,Sun 向公众推出 Java 时曾引起业界的巨大轰动。Java 语言面向网络应用,其类库不断丰富,性能不断提高,应用领域也不断拓展,已成为当今最通用、流行的软件开发语言之一,是许多专业人员首选的开发语言。Java 从诞生之日起历经多次修改,最新的 Java 技术一般称为 Java 2 平台,它由以下三部分组成:

- Java 2 平台标准版(J2SE);
- Java 2 平台企业版(J2EE);
- Java 2 平台微型版(J2ME)。

本书主要使用标准版。

Sun 公司为开发人员提供了软件开发工具包(Software Development Kit,JDK),并不断更新。目前针对 Java 2 平台的工具包版本是 J2SDK,读者可以在 Sun 公司的网站 <http://java.sun.com/j2se> 上查询当前最新的版本。

那么,Java 到底是什么?为什么它一问世就引起计算机界如此强烈的反响呢?实际上,Java 是一种功能强大的程序设计语言,它既是开发环境,又是应用环境,代表了一种新的计算模式。特别是从 1993 年开始互联网的流行,为 Java 提供了发挥潜能的机会。图 1-1 说明了 Java 语言的基本概念。

Java 语言	面向对象的程序设计语言
	与机器无关的二进制格式类文件
	Java 虚拟机(用来执行类文件)
	完整的软件程序包(跨平台的 API 和库)

图 1-1 Java 语言的基本概念

Java 是简单的、面向对象的语言,它具有分布性、安全性和健壮性等特点。它的最初版本是解释执行,现在的版本中增加了编译执行,所以它具有高性能;它是多线程的、动态的语言;最主要的特点是它与平台无关,解决了困扰软件界多年的软件移植问题。Java 语言既具有 C++ 语言的功能,又具备对类型进行严格检查的安全特性。

1.1.1 Java 语言的特点

Java 语言自诞生之日起,就受到了全世界的关注。Java 的出现标志着一个新的计算时代的到来,即 Java 计算时代。Java 的众多突出特点使得它受到了大众的欢迎。实际上,Java 符合目前面向对象程序设计的主流,它与 Web 及 Internet 的结合紧密,具有动画、声音等功能,能实时处理信息。它有如下显著的特点:

1. 语法简单,功能强大

Java 是一种类似于 C++ 的语言,熟悉 C++ 的程序设计人员不需花费太多的精力就可以掌握 Java。Java 去掉了 C++ 中不常用且容易出错的地方。例如,Java 中没有指针、结构和类型定义等概念,不再有全局变量,没有 #include 和 #define 等预处理器,也没有多重继承的机制。程序员不用自己释放占用的内存空间,因此不会引起因内存混乱而导致的系统崩溃。

Java 强调了面向对象的特性,是一个完全的面向对象的语言,对软件工程技术有很强支持。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以及动态的接口模型。Java 的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。另一方面,Java 的类提供了一类对象的原型,通过继承机制,子类可以使用父类所提供的方法,从而实现了代码的复用。Java 的解释器只占用很少的内存,适合在绝大多数类型的机器上运行。因此,使用 Java 可以编制出非常复杂的系统。

2. 分布式与安全性

Java 从诞生之日起就与网络联系在一起。它强调网络特性,内置 TCP/IP、HTTP、FTP 协议类库,便于开发网上应用系统。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。

3. 与平台无关

提到 Java,有一句著名的口号:“一次编写,到处运行”。这反映了 Java 的平台无关性,实现了编程人员多年的梦想。Java 可以跨平台使用,因此更适合于网络应用。Java 语言规定了统一的数据类型,有严格的语言定义,为 Java 程序跨平台的无缝移植提供了很大的便利。Java 编译器将 Java 程序编译成二进制代码,即字节码(bytecode)。不同的操作系统有不同的虚拟机,它类似于一个小巧而高效的 CPU。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 运行环境的系统上,都可以执行这些代码。运行时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“到处运行”。

4. 解释、编译两种运行方式

Java 程序可以经解释器得到字节码,所生成的字节码经过了精心设计,并进行了优