



北京高等教育精品教材
BEIJING GAODENG JIAOYU JINGPIN JIAOCAI

普通高等院校
计算机专业(本科)实用教程系列

Java程序设计

案例教程(第3版)

杨树林 胡洁萍 编著

清华大学出版社

普通高等院校计算机专业(本科)实用教程系列

Java 程序设计案例教程(第3版)

杨树林 胡洁萍 编 著

清华大学出版社
北 京

内 容 简 介

Java 具有面向对象、与平台无关、安全、稳定和多线程等特点,是目前最流行的程序设计语言之一。全书共分为 9 章,按 Java 知识的系统性,由浅入深、循序渐进地介绍 Java 语言实用知识与编程技术,内容包括 Java 语言基础、控制结构与异常处理、Java 面向对象编程、数组与集合、GUI 程序设计、Java 数据库编程、流和文件、Java 多线程机制和 Java 网络编程。每部分内容既有示例,又有案例。通过示例帮助学生理解知识,通过案例将各知识点结合起来,指导学生应用知识,达到学以致用。教材中引进一些新知识和新方法,内容实用、重点突出、语言精练、案例典型,既方便学习,又便于应用。配备的课后习题参考了目前市场上常用的企业面试题,针对性强,训练价值大。为便于学习和实践,教材在每章开始给出了内容简介和教学目标,每章结束给出了实验题目。

本书内容丰富,实例典型,知识讲解系统,适合作为大中专院校计算机及其相关专业的教材或参考书,也适合作为软件开发人员及其他有关人员的自学参考书或培训教程。

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 程序设计案例教程/杨树林,胡洁萍编著.--3 版.--北京:清华大学出版社,2016

普通高等院校计算机专业(本科)实用教程系列

ISBN 978-7-302-42018-7

I. ①J… II. ①杨… ②胡… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 263115 号

责任编辑:郑寅堃 薛 阳

封面设计:张 昱

责任校对:焦丽丽

责任印制:沈 露

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 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:20 字 数:480 千字

版 次:2006 年 8 月第 1 版 2016 年 1 月第 3 版 印 次:2016 年 1 月第 1 次印刷

印 数:14501~16000

定 价:39.00 元

第3版前言

解决学生实践应用能力不强的问题,必须更新教学内容、改变教学模式。为了适应应用型本科院校教学的需要,教学内容的选取、设计和组织是至关重要的。案例教学是计算机语言教学的最有效的方法之一,为此,如何将知识和案例有机结合起来,是教材的关键所在。2006年出版的《Java语言最新实用案例教程》(第1版)得到了兄弟院校教师的认可,2010年出版的《Java语言最新实用案例教程》(第2版)更是得到了广大高等学校专家、教师、学生的赞同、支持和厚爱。尽管第2版教材纠正了第1版教学的不足,纠正了第1版教材案例过于复杂,部分内容实用性不大,以及知识讲解不够系统性等问题,但经过使用,发现第2版教材还有改进的余地,如一些知识实用性不强或过于复杂,有些内容还需要引进,案例联系性还不够,内容组织还有待优化等。为此,作者着手编写这一版教材,并将教材更名为《Java程序设计案例教程》,目的是突出基础性、教学性和应用性。新版教材不仅保留了前两版教材的优点,又在如下方面做了改进。

(1) 对教与学的指导性加强。例如,教材在每章增加了知识简介、教学目标和实验题目,教学内容的选取更加体现应用型人才的培养要求,强调基础性。

(2) 教材内容进一步优化。例如,删减了一些不常用(如 Applet 知识)和一些偏难的内容(如线程新特征),调整了一些案例,引进了新内容(如 Properties),强化了对面向对象程序设计的理解(如给出了引例),并优化了内容结构。

(3) 教材更加强案例引导。一些综合性的案例贯穿多个章节,与知识结合得更加紧密。例如,在面向对象编程中,去掉了原来零散的案例,而主要以银行账户和绘图软件两个案例为主;银行账户案例随知识进展不断提升,并且在多线程机制一章进一步设计;绘图软件案例综合应用了接口、继承、绘图等知识,并在 GUI 程序设计一章,围绕界面设计及事件处理得到完整实现,有利于培养学生综合应用能力。

全书共分为9章,内容包括Java语言基础、控制结构与异常处理、Java面向对象编程、数组与集合、GUI程序设计、Java数据库编程、流和文件、Java多线程机制和Java网络编程。除包含许多配合知识学习的示例外,全书还包含40个较完整的案例。

本书得到了北京印刷学院校级教学改革项目和北京印刷学院优秀教学团队项目的资助,在此表示感谢。

由于作者水平有限,书中难免存在疏漏和不足,恳请读者批评指正,使本书得以改进和完善。

作 者

2015年8月于北京

第2版前言

要解决学生实践应用能力不强的问题,不仅要重视软件技术类课程,更要更新教学内容、改变教学模式。案例教学是计算机语言教学的最有效的方法之一。好的案例对学生理解知识,掌握如何应用知识十分重要。《Java 语言最新实用案例教程》(第1版)于2006年出版,已经历了近四年的使用,得到了广大高等学校专家、教师、学生的支持和厚爱。这本教材以指导案例教学为目的,以知识为线索设计案例,将案例有机联系起来,围绕案例讲解知识,教材组织方式新颖,案例丰富,适应了应用型人才培养的要求。但经过几年的使用发现还存在一些不足,如过分强调案例,知识不够系统;有些案例偏难,不适于学习;强调 Swing 过多,部分内容实用性不强;教材中没有习题,不便于教学等。为此,我们及时修订,出版了第2版教材,力求在知识讲解和案例之间找到最佳结合点,既便于教学和学习,又有利于培养学生的应用能力,适应案例教学的要求。第2版教材除保留原教材案例教学的特色外,又结合多年来的教学实践,在以下几个方面做了改进。

(1) 适当加强知识讲解的系统性,调整先讲案例再围绕案例介绍知识的方式,而是先系统、精练地讲解知识,再围绕知识渗透案例。知识内容不强调细而全,但强调系统、实用和精练,突出市场中常用的内容。同时,为了便于教学和学习,增加了许多辅助理解知识的小例子。

(2) 继续体现案例教学的思想。一方面沿用原版教材中好的案例,但对案例重新设计,使其更加优化;另一方面舍弃原版教材中偏难、实用性不大的案例,增加一些新的案例,使案例更接近于实际应用,同时便于教学和学习。新版教材仍然强调案例之间的联系,每部分的案例尽可能是通过大案例的分解而得,将案例穿插到知识讲解中,使案例与知识相辅相成,形成有机的整体,既有利于学生学习知识,又有利于指导学生实践。

(3) 跟踪 Java 新发展,注意适应市场需求,及时引进新内容,如可变参数、枚举、线程新特征等;强调了集合的应用;渗透了一些新思想,如 MVC 设计模式、面向接口编程、分层架构;使用了新的开发模式;配备了每章总结和习题,这些习题参考了常见的企业面试题,更具训练价值。

全书共分为9章,内容包括Java语言概述、流程控制与异常处理、Java面向对象程序设计、数组与集合、GUI程序设计、Java数据库编程、流和文件、多线程与Applet、Java网络编程。除包含许多配合知识学习的例子外,全书还包含68个案例。主要特点如下。

- (1) 精心设计知识结构,讲解精炼,重点突出,便于教学和学习。
- (2) 注意吸收新方法和新技术,强调实用性,重视应用能力的培养。
- (3) 案例系统、典型,将知识内容和案例有机结合,便于指导学生实践。
- (4) 较好地处理具体案例与思想方法、局部知识应用与综合应用的关系。

本书是北京市精品教材立项项目,得到了北京市委的经费支持,在此表示感谢。

由于作者水平有限,书中难免存在疏漏和不足,恳请读者批评指正,使本书得以改进和完善。

作 者

2010年7月于北京

第1版前言

Java 语言具有面向对象、与平台无关、安全、稳定和多线程等特点,不仅可以用来开发大型的应用程序,而且特别适合于 Internet 的应用开发,近年来已逐渐成为一门主流语言。目前无论是高校的计算机专业还是 IT 培训学校都将 Java 语言作为教学内容之一,这对于培养学生的应用能力,提高学生就业素质具有重要的意义。

解决学生实践应用能力不强的问题,不仅要重视软件技术类课程,更要更新教学内容、改变教学模式。案例教学是计算机语言教学的最有效的方法之一。好的案例对学生理解知识,掌握如何应用知识十分重要。目前一些教材类书籍,内容繁杂,常用知识突出不够;新内容没有及时引入,与市场缺乏衔接;例子缺乏实用性和系统性,相互联系不够,对学生的技术指导不利,无法适应案例教学。而一些技术性较强的参考书,又过分强调技术,知识讲解不够系统,不适合于教学。为此,最好的办法就是把案例和知识有机结合起来。一方面,跟踪 Java 发展,适应市场需求,精心选择内容,突出重点,强调实用,使知识讲解系统、精练;另一方面,设计典型的案例,将案例分解、融入到知识讲解中,使知识与案例相辅相成,达到既有利于学生学习知识,又有利于指导学生实践的目的。按照这种思想,本教材吸收了其他教材的优点,克服了其案例过于复杂、部分内容实用性不强、知识讲解不够系统等缺点,重新调整了教材内容的组织方法,精选了教学内容,引进了一些实用内容和程序设计方法,也借鉴了普通教材的优点。

全书共分为 9 章,内容包括 Java 语言概述、流程控制与异常处理、Java 面向对象程序设计、数组与集合、GUI 程序设计、Java 数据库编程、流和文件、多线程与 Applet、Java 网络编程。教材中突出了一些实际应用中常用的内容,引进了可变参数、枚举、线程新特征等一些新内容,强调了集合应用,渗透了一些新思想,如 MVC 设计模式、面向接口编程、分层架构,增加了一些有价值的习题,这些习题参考了常见的企业面试题。本书不同于普通教材的组织方式,它重视学生应用能力培养,精选教学内容,体现案例教学思想,教材中既包含许多辅助理解知识的示例,也包含许多指导应用知识的案例,主要特点如下。

- (1) 精心设计知识结构,知识讲解系统,语言精练,重点突出。
- (2) 注意吸收新方法和新技术,强调实用性,重视应用能力的培养。
- (3) 突出案例教学思想,知识内容和案例有机结合,有利于指导学生实践。
- (4) 较好地处理具体案例与思想方法、局部知识应用与综合应用的关系
- (5) 知识讲解循序渐进,难度适宜,便于教学和学习。

本书是北京市精品教材立项项目,得到了北京市委的经费支持,在此表示感谢。

由于作者水平有限,书中难免存在疏漏和不足,恳请读者批评指正,使本书得以改进和完善。

作者

2006 年 7 月于北京

目 录

第 1 章 Java 语言基础	1
1.1 Java 语言简介	1
1.1.1 Java 语言的发展	1
1.1.2 Java 的运行机制	2
1.1.3 Java 语言的特点	3
1.2 Java 编程环境安装	5
1.2.1 JDK 及其安装	5
1.2.2 Java 开发工具	6
1.3 初识 Java 应用程序	9
1.3.1 Java 应用程序的结构	9
1.3.2 编写和运行 Java 应用程序	10
1.3.3 案例 1-1 包含两个类的程序	13
1.4 Java 语言基本语法	14
1.4.1 基本编码规则	14
1.4.2 案例 1-2 为程序加注释	16
1.4.3 数据类型及其转换	17
1.4.4 常量、变量和表达式	20
1.5 字符串和日期	25
1.5.1 字符串	25
1.5.2 案例 1-3 对输入的字符串进行处理	30
1.5.3 日期和时间	31
1.5.4 案例 1-4 日期工具类	32
小结	35
习题	35
实验	36
第 2 章 控制结构与异常处理	37
2.1 分支结构	37
2.1.1 if 语句	37
2.1.2 案例 2-1 求一元二次方程的根	41
2.1.3 switch 语句	42
2.1.4 案例 2-2 求下一天日期	44
2.2 循环结构	46

2.2.1	for 循环	46
2.2.2	案例 2-3 求素数	47
2.2.3	while 循环	48
2.2.4	循环嵌套	48
2.2.5	案例 2-4 求 $\sin(x)$	49
2.2.6	do...while 循环	50
2.2.7	案例 2-5 进制转换	51
2.2.8	迭代循环	52
2.3	异常处理	53
2.3.1	异常及其体系结构	53
2.3.2	异常处理机制	54
2.3.3	抛出异常	57
2.3.4	案例 2-6 整数的算术计算	58
2.3.5	自定义异常	59
2.3.6	案例 2-7 求三角形面积	60
小结		61
习题		61
实验		62
第3章	Java 面向对象编程	64
3.1	面向对象概述	64
3.1.1	对象和类的概念	64
3.1.2	面向对象程序设计	65
3.1.3	OOP 的关键性理念	67
3.1.4	OOP 的 4 个基本特征	67
3.2	定义类与创建对象	68
3.2.1	定义类	69
3.2.2	创建和使用对象	69
3.2.3	构造方法	70
3.2.4	访问控制与属性	71
3.2.5	案例 3-1 银行账户类	73
3.3	类的方法与重载	75
3.3.1	方法的定义	75
3.3.2	方法的参数类型	76
3.3.3	方法重载	78
3.4	实例成员和类成员	78
3.4.1	实例变量和类变量	78
3.4.2	实例方法和类方法	79
3.4.3	案例 3-2 为银行账户类增加功能	80

3.5 类的继承	82
3.5.1 继承的基本概念	83
3.5.2 定义子类	83
3.5.3 方法覆盖与多态性	86
3.5.4 案例 3-3 完善银行账户类	87
3.6 抽象类与接口	90
3.6.1 抽象类	90
3.6.2 接口	91
3.6.3 案例 3-4 为绘图软件设计一组图形类	93
3.7 内部类与枚举类型	97
3.7.1 内部类	97
3.7.2 枚举类型	99
小结	101
习题	102
实验	103
第 4 章 数组与集合	104
4.1 数组	104
4.1.1 数组的概念	104
4.1.2 数组的定义	104
4.1.3 案例 4-1 成绩排序和统计	106
4.2 集合	108
4.2.1 Java 集合框架	108
4.2.2 Collection 接口常用方法	109
4.2.3 遍历 Collection	109
4.2.4 Collection 的批量操作	110
4.3 集	110
4.3.1 HashSet 类	111
4.3.2 TreeSet 类	113
4.4 列表	115
4.4.1 List 接口	115
4.4.2 ArrayList 类	115
4.4.3 案例 4-2 竞赛评分程序	116
4.4.4 Vector 类	120
4.5 映射	120
4.5.1 Map 接口	120
4.5.2 HashMap 类	121
4.5.3 案例 4-3 网络书城中的购物车类	121
4.5.4 Hashtable 类	124

7.3.3 案例 7-3 用字节流复制文件	238
7.4 数据流和对象流	240
7.4.1 数据流简介	240
7.4.2 对象流简介	241
7.4.3 案例 7-4 为绘图软件增加保存和打开功能	242
小结	245
习题	245
实验	246
第 8 章 Java 多线程机制	247
8.1 线程概述	247
8.1.1 线程与进程	247
8.1.2 线程的优点	248
8.1.3 线程体与线程载体	248
8.2 线程的创建	248
8.2.1 Thread 类	248
8.2.2 创建线程的两种方式	249
8.2.3 案例 8-1 为学生管理系统增加启动界面和状态时钟	250
8.3 线程的状态与优先级	253
8.3.1 线程的状态	253
8.3.2 线程的控制	254
8.3.3 线程组与线程优先级	256
8.3.4 案例 8-2 图片浏览程序	258
8.4 线程同步与通信	262
8.4.1 Java 线程同步机制	262
8.4.2 案例 8-3 取款和存款	262
8.4.3 Java 线程通信机制	264
8.4.4 案例 8-4 哲学家用餐问题	265
8.4.5 “生产者-消费者”问题	267
8.4.6 案例 8-5 吃苹果	267
小结	270
习题	271
实验	271
第 9 章 Java 网络编程	273
9.1 网络编程基础	273
9.1.1 网络基本概念	273
9.1.2 网络协议	274
9.2 获取网络信息与资源	275

9.2.1 获取网络地址信息.....	275
9.2.2 获取网络资源属性.....	276
9.2.3 获取网络资源.....	278
9.2.4 案例 9-1 读取和下载网上文件	280
9.3 基于 TCP 的网络通信.....	282
9.3.1 客户/服务器模式和套接字	282
9.3.2 客户端程序的原理.....	283
9.3.3 案例 9-2 TCP 客户端程序	284
9.3.4 服务器程序的原理.....	288
9.3.5 案例 9-3 TCP 服务器端程序	289
9.4 基于 UDP 的网络通信	293
9.4.1 基于 UDP 网络通信的原理	293
9.4.2 案例 9-4 基于 UDP 的网络通信	295
小结.....	299
习题.....	299
实验	300
参考文献.....	301

第1章 Java 语言基础

【内容简介】

Java 是 1995 年美国 Sun 公司正式推出的完全面向对象的程序设计语言,具有简单、稳定、与平台无关、安全、解释执行、多线程等特点,是目前使用最为广泛的网络编程语言。本章详细介绍 Java 语言的发展、运行机制和特点,Java 编程环境的安装,如何创建和运行 Java 程序,以及 Java 语言的基本语法、字符串和日期等知识。

本章将通过“包含两个类的程序”、“为程序加注释”、“对输入的字符串进行处理”和“日期工具类”4 个案例,帮助读者熟悉集成开发环境,掌握 Java 程序的设计步骤、理解 Java 语言的基本知识。

通过本章的学习,读者将学会利用 NetBeans 开发环境设计简单的 Java 应用程序。

【教学目标】

- ☐ 了解 Java 语言的发展、体系结构;
- ☐ 理解 Java 程序的运行机制及 Java 语言的特点;
- ☐ 掌握 Java 集成开发的安装和使用;
- ☐ 初步理解 Java 应用程序的基本结构;
- ☐ 掌握 Java 语言基本的语法;
- ☐ 掌握 Java 字符串、日期和时间的常用处理方法;
- ☐ 能够用 Java 语言编写简单的应用程序。

1.1 Java 语言简介

1.1.1 Java 语言的发展

Java 语言的前身是 Oak 语言。Sun 公司 1995 年正式发布了 Java 的第一个公开版本。

1991 年,在 Sun 公司由 James Gosling(图 1-1)和 Patrick Naughton 领导的 Green 研究小组,为了能够在消费电子产品上开发应用程序,积极寻找合适的编程语言。消费电子产品种类繁多,包括 PDA、机顶盒、手机等,即使是同一类消费电子产品所采用的处理芯片和操作系统也不相同,也存在着跨平台的问题。起初他们考虑采用 C++ 语言来编写消费电子产品的应用程序,但是研究表明,对于消费电子产品而言,C++ 语言过于复杂和庞大,并不适用,安全性也不令人满意。最后,Green 小组基于 C++ 开发出一种新的语言 Oak,该语言采用了许多 C 语言的语法,提高了安全性,并且是面向对象的语言。但是 Oak 语言在商业上并未获得成功。之后随着互联网在世界上蓬勃发展,Sun 公司发现 Oak 语言所具有的跨平台、面向对象、安全性高等特点,非常符合互

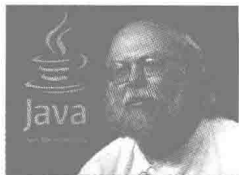


图 1-1 James Gosling

联网的需要,于是转向互联网应用,进一步改进该语言的设计,并最终将这种语言取名为 Java。

1995 年 5 月 23 日, Sun 在 SunWorld'95 上正式发布 Java 和 HotJava 浏览器,并被美国杂志 *PC Magazine* 评为 1995 年十大优秀科技产品,标志 Java 语言的诞生。Java 的出现,正好迎合了当时的互联网市场的需求,这使得 Java 语言犹如雨后的竹节迅速得以发展,之后 Sun 相继推出了 Java 开发工具包 JDK 1.0、JDK 1.1。

1998 年 JDK 1.2 发布, JDK 更名为 J2SDK(Java 2 Software Development Kit),又称 Java 2。此时根据 Java 的应用可以分为三大技术方向:一是 J2SE(Java 2 Standard Edition),桌面应用开发,包括 C/S 结构;二是 J2ME(Java 2 Micro Edition),主要应用在移动应用开发;三是 J2EE(Java 2 Enterprise Edition),企业级应用开发。

2004 年 J2SE 1.5 的发布,是 Java 语言历史发展的又一个里程碑。为了表示该版本的重要性, Sun 公司将此版改名为 Java SE 5。2005 年,在 Sun 公司发布了 Java SE 6 之后,将 Java 的各种版本都进行了更名,正式改为 Java SE 6, Java EE 6, Java ME 6。目前已经发布了 Java SE 8, Java EE 8, Java ME 8。

近年来,移动互联网的发展又使 Java 焕发出了新的生机。Google 发布的 Android 占据了手机操作系统的半壁江山,而 Android 系统正是使用了 Java 语言。这就使得 Java 在移动业务方面占据着不可估量的市场优势。

1.1.2 Java 的运行机制

1. Java 虚拟机

Java 源程序不是编译成可执行文件,而是编译成字节码文件,Java 虚拟机(Java Virtual Machine, JVM)可以解释和运行 Java 字节码文件。

Java 虚拟机由 Java 解释器和运行平台构成,它的作用类似 CPU,负责执行指令,管理内存和存储器,因此可看成是软件模拟的计算机。Java 虚拟机的“机器码”保存在 .class 文件中,有时也可以称之为字节码文件。Java 程序的跨平台主要是指字节码文件可以在任何具有 Java 虚拟机的计算机或者电子设备上运行。Java 虚拟机中的 Java 解释器负责将字节码文件解释成为特定的机器码进行运行。Java 源程序需要通过编译器编译成为 .class 文件(字节码文件)。Java 程序的编译和执行过程如图 1-2 所示。



图 1-2 Java 工作原理

2. 无用内存自动回收机制

在程序的执行过程中,部分内存在使用过后就处于废弃状态,如果不及时进行无用内存

的回收,就会导致内存泄漏,进而导致系统崩溃。在 C++ 语言中是由程序员进行内存回收的,程序员需要在编写程序的时候把不再使用的对象内存释放掉;但是这种人为管理内存释放的方法却往往由于程序员的疏忽而致使内存无法回收,同时也增加了程序员的工作量。而在 Java 运行环境中,始终存在着一个系统级的线程,专门跟踪内存的使用情况,定期检测出不再使用的内存,并进行自动回收,避免了内存的泄漏,也减轻了程序员的工作量。

3. 代码安全性检查机制

字节码的执行需要经过三个步骤,首先由类装载器(Class Loader)负责把类文件(.class 文件)加载到 Java 虚拟机中,在此过程中需要检验该类文件是否符合类文件规范;其次字节码校验器(Bytecode Verifier)检查该类文件的代码中是否存在着某些非法操作,例如,Applet 程序中向本机文件系统进行写操作;如果字节码校验器检验通过,由 Java 解释器负责把该类文件解释成为机器码进行执行。Java 虚拟机采用的是“沙箱”运行模式,即把 Java 程序的代码和数据都限制在一定内存空间里执行,不允许程序访问该内存空间外的内存,如果是 Applet 程序,还不允许访问客户端机器的文件系统。

1.1.3 Java 语言的特点

Java 语言是一种完全面向对象的语言。它充分吸取了 C++ 语言的优点,采用了程序员所熟悉的 C 和 C++ 语言的许多语法,同时又去掉了 C 语言中指针、内存申请和释放等影响程序健壮性的部分,并增加了很多新的特性。Java 语言具有如下特点。

1. 简单

一方面,Java 语言的语法与 C 语言和 C++ 语言很接近,使得大多数程序员很容易学习和使用。另一方面,Java 丢弃了 C++ 中很少使用的、难理解的那些特性,更简单、更精练。在语法规则方面,Java 语言放弃了全程变量、goto 语句、宏定义、全局函数以及结构、联合等;在面向对象方面放弃了操作符重载、多继承、虚基类等。特别地,Java 语言不使用指针,并提供了自动的垃圾收集,使得程序员不必为内存管理而担忧。

2. 面向对象

基于面向对象思想编写的程序结构化程度高,提高了代码的可重用性,增加了程序的可读性和可维护性。与 C++ 不同,Java 对面向对象的要求十分严格,是一种纯粹的面向对象程序设计语言。Java 以类和对象为基础,任何变量和方法都只能包含于某个类的内部,这使得 Java 程序结构更为清晰,并为集成和代码重用带来了便利。为了简单起见,Java 不支持多重继承,但支持实现多接口,并且 Java 语言全面支持动态绑定。

3. 分布式

分布式计算涉及几台计算机在网络上一一起工作。Java 的设计使分布计算变得容易起来。Java 提供了用于网络应用编程的类库,包括 URL、URLConnection、Socket、ServerSocket

等。Java 的 RMI(Remote Method Invocation, 远程方法激活)机制也是开发分布式应用的重要手段。

4. 健壮

Java 的强类型机制、异常处理、垃圾的自动回收等是 Java 程序健壮性的重要保证。对指针的丢弃是 Java 的明智选择。Java 致力于检查程序在编译和运行时的错误, 类型检查帮助检查出许多早期开发出现的错误。Java 自己操纵内存减少了内存出错的可能性。Java 的安全检查机制使得 Java 更具健壮性。

5. 结构中立

Java 将它的程序编译成一种结构中立的中间文件格式。只要有 Java 运行环境的机器都能执行这种中间代码。中间文件格式是一种高层次的与机器无关的字节码格式语言, 这种语言被设计在虚拟机上运行。

6. 安全

Java 通常被用在网络环境中, 为此, Java 提供了一个安全机制以防恶意代码的攻击。除了具有的许多安全特性以外, Java 对通过网络下载类具有一个安全防范机制, 如分配不同的名字空间以防替代本地的同名类, 字节代码检查, 并提供安全管理机制为 Java 应用设置安全哨兵。

7. 可移植性

与体系结构无关的特性使得 Java 应用程序可以在配备了 Java 解释器和运行环境的任何计算机上运行, 这成为 Java 应用软件便于移植的良好基础。通过定义独立于平台的基本数据类型及其运算, Java 数据得以在任何硬件平台上保持一致。

8. 解释性

如前所述, Java 程序在 Java 平台上被编译为字节码格式, 然后可以在实现这个 Java 平台的任何系统中运行。在运行时, Java 平台中的 Java 解释器对这些字节码进行解释执行, 执行过程中需要的类在连接阶段加载到运行环境中。

9. 多线程

Java 提供的多线程功能使得在一个程序里可同时执行多个小任务。Java 比 C 和 C++ 更健壮。多线程带来的更大的好处是更好的交互性能和实时控制性能。

10. 动态

Java 语言的设计目标之一是适应于动态变化的环境。Java 程序需要的类能够动态地被载入到运行环境, 也可以通过网络来载入所需要的类。这也有利于软件的升级。另外, Java 中的类有一个运行时刻的表示, 能进行运行时刻的类型检查。

1.2 Java 编程环境安装

1.2.1 JDK 及其安装

1. JDK 简介

JDK(Java Development Kit)是 Sun 公司最新提供的基础 Java 语言开发工具软件包。其中包含 Java 语言的编译工具、运行工具以及类库。

1) JDK 目录结构

(1) bin 目录: 包含编译器、解释器和一些工具。

(2) lib 目录: 包含类库文件。

(3) demo 目录: 包含各种演示例子。

(4) include 目录: 包含 C 语言头文件,支持 Java 本地接口与 Java 虚拟机调试程序接口的本地编程技术。

(5) jre 目录: 包含 Java 虚拟机、运行时的类包和 Java 应用启动器。

(6) sample 目录: Sun 配带的帮助学习者学习的 Java 例子。

(7) src.zip: 源码压缩文件。

2) bin 目录下的常用工具

(1) javac.exe: Java 语言编译器,输出结果为 Java 字节码。

(2) java.exe: Java 字节码解释器。

(3) javadoc.exe: 帮助文档生成器。

(4) jar.exe: 打包工具。

(5) appletviewer.exe: 小应用程序浏览工具,用于测试并运行 Applet 小程序。—

2. JDK 下载与安装

1) JDK 下载

JDK 是一个开源、免费的工具,可以到 Sun 公司的官方网站上下载 JDK 最新版本,网址为 <http://www.oracle.com/technetwork/java/javase/downloads/index.html>。本书使用的 JDK 版本是 Java SE Development Kit 8u45。

2) JDK 安装

下载后得到 jdk-8u45-windows-i586.exe 文件(如果是 64 位操作系统,下载的文件是 jdk-8u45-windows-x64.exe),直接双击运行即开始安装。在安装过程中可以选择安装路径和安装组件,如果没有特殊要求,使用默认设置即可。默认的安装路径是 C:\Program Files\Java\jdk1.8.0_45。

3) 设置环境变量

设置环境变量是为了更方便地使用 JDK 开发,某些 Java 程序也依赖环境变量来定位。环境变量设置如下。

JAVA_HOME =<JSEDK 安装目录>

CLASSPATH = .; %JAVA_HOME%\lib\dt.jar; %JAVA_HOME%\lib\tools.jar

Path = <原 Path>; %JAVA_HOME%\bin; %JAVA_HOME%\jre\bin

1.2.2 Java 开发工具

Java 程序的编写可以使用任何一种文本编辑器,如 UltraEdit、Notepad、WordPad,甚至 Word。当然最好是使用集成开发环境。目前比较流行的集成开发环境主要有 NetBeans、Eclipse、MyEclipse、JCreator、JBuilder。本书使用的是 NetBeans 8.0.2。

NetBeans 是 Sun 公司推出的开放源码的 Java 集成开发环境(Integrated Development Environment, IDE)。它是使用 Java 语言编写的,具有很好的可移植性,适用于各种客户机和 Web 使用,是业界第一款支持创新型 Java 开发的开放源码 IDE。NetBeans 的下载地址是 <https://netbeans.org/downloads/index.html>。

下载后安装步骤如下。

(1) 运行 netbeans-8.0.2-javase-windows.exe,出现如图 1-3 所示的窗口。在这个对话框中单击【下一步】按钮。



图 1-3 安装起始界面

(2) 出现【许可证协议】窗口,如图 1-4 所示。选中【我接受许可证协议中的条款】复选框,单击【下一步】按钮。

(3) 出现【JUnit 许可证协议】窗口,如图 1-5 所示。选择【我接受许可证协议中的条款。安装 JUnit】单选按钮,单击【下一步】按钮。