

21世纪高等学校计算机教育实用规划教材

Java程序设计基础

栾颖 主编



清华大学出版社

21世纪高等学校计算机教育实用规划教材

Java程序设计基础

栾颖 主编

清华大学出版社
北京

TP312JA
1650

内 容 简 介

本书共分为10章。首先介绍Java基础、Java的数据类型及程序结构等内容,使读者对Java有个概要的认识和理解;接着详细介绍Java的面向对象编程、图形界面编程、接口、多线程、异常抛出捕获等内容。使读者深入了解Java语言的特点与优势。本书具有结构严谨、层次分明、深入浅出、实例精彩、指导性强等特点;编写风格为:从概念入手,通过相关示例来实践概念,使读者真正理解Java语言的实质,从而让读者更好、更快速地掌握Java。

本书可作为高等院校计算机及相关专业的Java语言程序设计或网络编程基础等课程的教材,也可以作为应用开发人员的自学参考书。

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 程序设计基础/栾颖主编.--北京:清华大学出版社,2014
21世纪高等学校计算机教育实用规划教材
ISBN 978-7-302-38356-7

I. ①J… II. ①栾… III. ①JAVA语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2014)第243366号

责任编辑:魏江江 薛 阳

封面设计:常雪影

责任校对:时翠兰

责任印制:杨 艳

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:21.75 字 数:530千字

版 次:2014年11月第1版 印 次:2014年11月第1次印刷

印 数:1~2000

定 价:39.50元

出版说明

随着我国高等教育规模的扩大以及产业结构调整进一步完善,社会对高层次应用型人才的需求将更加迫切。各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,合理调整和配置教育资源,在改革和改造传统学科专业的基础上,加强工程型和应用型学科专业建设,积极设置主要面向地方支柱产业、高新技术产业、服务业的工程型和应用型学科专业,积极为地方经济建设输送各类应用型人才。各高校加大了使用信息科学等现代科学技术提升、改造传统学科专业的力度,从而实现传统学科专业向工程型和应用型学科专业的发展与转变。在发挥传统学科专业师资力量强、办学经验丰富、教学资源充裕等优势的同时,不断更新教学内容、改革课程体系,使工程型和应用型学科专业教育与经济建设相适应。计算机课程教学在从传统学科向工程型和应用型学科转变中起着至关重要的作用,工程型和应用型学科专业中的计算机课程设置、内容体系和教学手段及方法等也具有不同于传统学科的鲜明特点。

为了配合高校工程型和应用型学科专业的建设和发展,急需出版一批内容新、体系新、方法新、手段新的高水平计算机课程教材。目前,工程型和应用型学科专业计算机课程教材的建设工作仍滞后于教学改革的实践,如现有的计算机教材中有不少内容陈旧(依然用传统专业计算机教材代替工程型和应用型学科专业教材),重理论、轻实践,不能满足新的教学计划、课程设置的需要;一些课程的教材可供选择的品种太少;一些基础课的教材虽然品种较多,但低水平重复严重;有些教材内容庞杂,书越编越厚;专业课教材、教学辅助教材及教学参考书短缺,等等,都不利于学生能力的提高和素质的培养。为此,在教育部相关教学指导委员会专家的指导和建议下,清华大学出版社组织出版本系列教材,以满足工程型和应用型学科专业计算机课程教学的需要。本系列教材在规划过程中体现了如下一些基本原则和特点。

(1) 面向工程型与应用型学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映基本理论和原理的综合应用,强调实践和应用环节。

(2) 反映教学需要,促进教学发展。教材规划以新的工程型和应用型专业目录为依据。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材建设仍然把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现工程型和应用型专业教学内容和课程体系改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材要配套,同一门课程可以有多种具有不同内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材,教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主编。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机教育实用规划教材编委会

联系人:魏江江 weijj@tup.tsinghua.edu.cn

前 言

Java 语言是一种新兴的网络编程语言,其卓越功能和特性为无数开发人员所推崇,越来越多的应用开发采用了基于 Java 技术的解决方案。Java 语言也是一种真正面向对象的编程语言,它提升了应用程序的编程概念和开发思维。Java 语言提供了丰富的开发类库,为用户编程提供了极大的支持和方便。Java 的优势在于它与操作平台的无关性。简单地说,Java 在网络上通行无阻。用 Java 编写的程序,支持 Solaris、Windows、MacOS、HP-UX 和 Linux 等各种平台,既可以在 UNIX 操作系统下运行,也可以在 Linux 和 Windows 操作系统下运行。

笔者比较了多种程序设计语言,认为 Java 有如下突出的优点。

(1) Java 是面向对象的语言,与现代面向对象的设计与分析的软件工程相一致,也是当前的主流程序设计语言之一。

(2) 简单易学。其中的数据类型、数据运算、程序控制结构等基本概念对于任何语言都是一致的;而其语法相对于 C++ 等语言而言更简单,更容易掌握。

(3) Java 语言就其本身而言支持一些高级特性,如多线程、异常处理、自动垃圾回收等,这些特性使 Java 成为极优秀的语言之一。

(4) Java 具有广泛的用途。Java 具有跨平台的特点,在各种平台上都有应用,它还可以有效地进行数据库、多媒体及网络的程序设计。不仅如此,Java 还与 JavaScript 十分相似,而后者可以广泛用于办公软件、网页设计、网络服务程序等方面。

对于学习者而言,选择一本好的教材至关重要。现在市面上有关 Java 的书不少,但适合于教学、自学的书却不多见。笔者基于多年程序设计语言的教学经验,结合个人的软件开发实践,力图使本书突出如下特色。

(1) 对 Java 语言的基础知识,包括数据类型、流程控制、类的封装与继承、多态、虚方法调用、传值调用等进行系统讲解,让学习者知其然,并知其所以然。

(2) 在教材内容编排上,力求由浅到深,循序渐进,举一反三,突出重点,运用口语化的语言,通俗易懂,讲求效率、内容经过多次提炼与升华,突出学习规律和学习技巧,是思维化的直接体现。

(3) 在讲解语言的同时,介绍它所采用的面向对象技术的基础理论、主要原则和思维方法。

(4) 理论与实践相结合,本教材每介绍一个新的概念都有相应的例子来说明。

本书共分为 10 章。

第 1 章介绍 Java 的语言基础,包括 Java 概述、Java 的运行环境、面向对象程序设计、Java 的运行等基础内容。第 2 章介绍 Java 的数据类型及程序结构,包括常量和变量、基本数据类

型、数组等内容。第3章介绍面向对象编程知识,包括对象、类、类的权限封装及成员等内容。第4章介绍类的继承,包括类的继承及其基本操作、类层次结构分析类与类的关系等内容。第5章介绍接口,包括接口的定义、接口的实现、接口的使用等内容。第6章介绍异常处理,包括异常、异常抛出、异常捕获等内容。第7章介绍图形界面设计及工具类,包括用户界面组件、AWT基本组件、AWT容器等内容。第8章介绍数据库JDBC的应用,包括JDBC简介、JDBC对象与接口等内容。第9章介绍流和文件,包括流、数据流的输入与输出、文件输入与输出流等内容。第10章介绍多线程,包括线程相关的概念、线程的使用、库等内容。

本书具有知识全面、实例精彩、指导性强的特点,力求以全面的知识性及丰富的实例来指导读者透彻学习Java各方面的技术。本书可以帮助Java初级、中级开发人员提高开发技能,掌握Java的新特性,书中的实例对高级开发人员也有一定的启发意义。

本书主要由栾颖编写,参加编写的还有刘超、邓俊辉、梁朗星、李旭波、张棣华、刘泳、邓耀隆、何正风和周品。

由于作者水平有限,所以错误和疏漏之处在所难免,期望得到各领域的专家和广大读者的批评指正。

编者

2014年8月

目 录

第 1 章 Java 的语言基础	1
1.1 Java 的定义	1
1.1.1 Java 概述	1
1.1.2 Java 带来的影响	1
1.1.3 Java 的特点	2
1.1.4 不同版本的 Java	3
1.1.5 Java Application 与 Java Applet	3
1.1.6 Java 与 C、C++ 的不同之处	4
1.1.7 丰富的 Java 类库	5
1.1.8 Java 的前景	5
1.2 Java 的运行环境	5
1.2.1 J2SE 的下载与安装	5
1.2.2 环境变量与测试	9
1.3 面向对象程序设计	13
1.3.1 面向对象的讲述	14
1.3.2 对象、类与实体	15
1.3.3 对象的状态与行为	15
1.3.4 对象的关系	15
1.3.5 对象的软件开发过程	16
1.4 Java 的运行	17
1.4.1 Java 程序	17
1.4.2 Java 程序的书写和保存	19
1.5 Applet 的编辑、编译与运行	19
1.5.1 Applet 源程序的编辑与编译: AppletWelcome.java	19
1.5.2 在 HTML 文件中嵌入 Applet	20
1.6 Java 程序代码的输入与输出	21
1.6.1 字符串的输入与输出	21
1.6.2 Java Application 图形的输入与输出	23
1.6.3 Applet 图形的输入与输出	24
1.6.4 Java Application 与 Applet 的共同程序	25
练习与思考题	26

第 2 章 Java 的数据类型及程序结构	27
2.1 常量和变量	27
2.1.1 常量	27
2.1.2 变量	27
2.2 基本数据类型	32
2.2.1 整型	33
2.2.2 字符型	34
2.2.3 浮点型	35
2.2.4 逻辑布尔型	35
2.3 数组	36
2.3.1 一维数组	36
2.3.2 多维数组	38
2.4 Java 的运算符	42
2.4.1 算术运算符	43
2.4.2 关系运算符	45
2.4.3 逻辑运算符	46
2.4.4 位运算符	48
2.4.5 赋值运算符	50
2.4.6 条件运算符	51
2.4.7 表达式及运算符的优先级	51
2.5 控制语句	52
2.5.1 选择语句	52
2.5.2 循环语句	55
2.5.3 跳转语句	58
练习与思考题	61
第 3 章 面向对象编程知识	63
3.1 概率	63
3.2 对象	63
3.2.1 对象的概念	63
3.2.2 对象句柄	64
3.2.3 初始化对象变量	64
3.3 类	64
3.3.1 类构造	65
3.3.2 术语	69
3.4 类的权限封装及成员	70
3.4.1 类的权限封装	70
3.4.2 类的成员	70
3.5 Time 类	76

3.5.1	Time 类的研究	76
3.5.2	this 的使用	79
3.5.3	Time 类的重载构造函数	80
3.6	使用已有类	84
3.6.1	构造器	84
3.6.2	Date 类	85
3.6.3	Math 类	91
3.6.4	GregorianCalendar 类	94
3.7	包	100
3.7.1	包的建立	100
3.7.2	包的使用	101
3.7.3	Java 包	102
	练习与思考题	103
第 4 章	类的继承	104
4.1	简介	104
4.2	类的继承	104
4.2.1	派生子类	105
4.2.2	继承及其相关操作	105
4.2.3	构造器的调用	106
4.2.4	super 的使用	109
4.2.5	父类与子类之间的转换	111
4.3	类层次结构分析	113
4.4	类与类的关系	116
4.4.1	依赖	116
4.4.2	聚合	117
4.4.3	继承	117
4.5	final 和 abstract 的属性	117
4.6	类的多态性	117
4.6.1	多态的实质	118
4.6.2	多态性设计	120
4.6.3	绑定	121
4.7	多态性的实现	122
4.7.1	由继承实现多态	122
4.7.2	接口实现多态	128
4.8	虚函数的调用	129
4.8.1	上溯造型	129
4.8.2	虚函数方法的调用	130
4.8.3	构造方法内部调用方法的多态性	132
4.9	创建和使用接口	133

4.9.1	Payable 类继承层次	134
4.9.2	程序的实现	134
	练习与思考题	139
第 5 章	接口	140
5.1	接口的定义	140
5.2	接口的实现	141
5.3	接口的使用	142
5.4	接口的多重实现	145
5.5	利用接口进行对象类的转换	149
5.6	Comparable 接口与 Iterator 接口	151
5.7	匿名类和内部类	151
5.7.1	匿名类	151
5.7.2	内部类	152
5.8	抽象	153
5.8.1	抽象行为	153
5.8.2	抽象类	153
5.9	抽象与接口的区别	157
	练习与思考题	157
第 6 章	异常处理	159
6.1	异常	159
6.1.1	异常的基本概念	159
6.1.2	异常类及其类层次结构	160
6.1.3	异常的类型	161
6.2	异常抛出	165
6.2.1	怎样抛出异常	166
6.2.2	怎样创建异常类	167
6.3	异常捕获	168
6.3.1	捕获多个异常	169
6.3.2	重新抛出异常	169
6.3.3	finally 子句	170
6.4	创建用户异常	172
6.4.1	用户自定义异常类	172
6.4.2	重抛异常及异常链接	173
6.5	异常传递	175
6.6	I/O 异常	177
	练习与思考题	179

第 7 章 图形界面设计及工具类	181
7.1 用户界面组件	181
7.1.1 JFC 简介	181
7.1.2 AWT	181
7.1.3 Swing	183
7.2 AWT 基本组件	185
7.2.1 标签(Label)	185
7.2.2 单选按钮(Button)	186
7.2.3 文本输入框(TextField)	186
7.2.4 多行文本输入框(TextArea)	186
7.2.5 复选框(Checkbox)	188
7.2.6 单选按钮(Radio Buttons)	190
7.2.7 画布(Canvas)	191
7.2.8 选择菜单(Choice)	193
7.2.9 菜单项(MenuItem)	194
7.2.10 列表框(Scrolling List)	194
7.2.11 文件对话框(FileDialog)	196
7.2.12 滚动条(Scrollbar)	197
7.3 AWT 容器	197
7.3.1 Frame 类	197
7.3.2 Panel 类	199
7.4 Dialog 类	200
7.4.1 非模态对话框	200
7.4.2 模态对话框	202
7.5 Swing 组件的使用	203
7.5.1 Swing 简介	203
7.5.2 容器类	205
7.5.3 组件	210
7.6 布局管理器	218
7.6.1 FlowLayout 布局管理器	218
7.6.2 BorderLayout 布局管理器	219
7.6.3 GridLayout 布局管理器	220
7.6.4 GridBagLayout 布局管理器	223
7.6.5 CardLayout 布局管理器	225
7.6.6 容器的嵌套	227
练习与思考题	228
第 8 章 数据库 JDBC 的应用	229
8.1 JDBC 简介	229

8.1.1	JDBC 技术概述	229
8.1.2	JDBC 驱动管理器和驱动程序类型	229
8.1.3	JDBC-ODBC 桥	231
8.1.4	ODBC 数据源设定	231
8.2	JDBC 对象与接口	234
8.2.1	数据库连接对象与接口	234
8.2.2	数据库查询接口	239
8.2.3	ResultSet 对象与接口	243
8.2.4	元数据	244
8.3	Java 访问数据库的基本步骤	248
	练习与思考题	251
第 9 章	流和文件	252
9.1	流	252
9.1.1	流概述	252
9.1.2	I/O 流	252
9.1.3	筛选流	253
9.1.4	其他 I/O 类和接口	254
9.2	数据流的输入与输出	254
9.2.1	标准输入和标准输出	254
9.2.2	字节流与处理流	257
9.3	文件输入与输出流	260
9.3.1	FileInputStream 与 FileOutputStream 类	260
9.3.2	FileReader 与 FileWriter 类	262
9.3.3	随机访问文件的输入和输出	265
9.3.4	File 类	270
9.4	对象流	273
9.4.1	存储对象	273
9.4.2	对象流应用示例	274
9.5	文件对话框	275
9.5.1	打开对话框	276
9.5.2	存储对话框	278
9.5.3	菜单对话框	280
9.5.4	运行可执行文件	283
	练习与思考题	284
第 10 章	多线程	286
10.1	线程相关的概念	286
10.1.1	线程	286
10.1.2	并发性	286

10.1.3	线程的状态	287
10.2	线程的使用	288
10.2.1	线程的创建方法	289
10.2.2	多线程的创建	293
10.2.3	线程的优先级	295
10.2.4	线程的基本控制	296
10.2.5	线程的等待和中断	299
10.3	库	302
10.3.1	库的安全性	302
10.3.2	多线程内核程序	304
10.3.3	多线程垃圾收集程序	305
10.4	线程的共享与控制	306
10.4.1	同步线程的共享	306
10.4.2	线程间的同步控制	310
10.5	线程的同步问题	314
10.5.1	死锁	314
10.5.2	恢复死锁	315
10.5.3	竞争	316
10.5.4	丢失唤醒	317
10.5.5	同步变量	317
	练习与思考题	331
	参考文献	333

第 1 章

Java 的语言基础

1.1 Java 的定义

Java 语言是当今流行的网络编程语言,它的面向对象、跨平台、分布应用等特点给编程人员带来一种崭新的计算机概念,使 WWW 从最初的单纯提供静态信息发展到现在的提供各种各样的动态服务,产生了巨大的变化。Java 不仅能够编写小应用程序实现嵌入网页的声音和动画功能,而且还能够应用于独立的大中型应用程序,其强大的网络功能能够把整个 Internet 作为一个统一的运行平台,极大地拓展了传统单机或客户机/服务器(Client/Server,C/S)模式应用程序的外延和内涵。

1.1.1 Java 概述

Java 语言是一种平台无关的、可移植的高级程序设计语言。Java 语言是 Sun 公司于 1995 年推出的类似于 C++ 语言的程序设计语言,它与 Internet 结合,最大的特点是可以在不同的平台环境中运行。Java 语言推出以后,计算机界对其好评如潮,Microsoft 的总裁 Bill Gates 对 Java 语言的评价是“Java 是长时间以来最为卓越的程序设计语言”,计算机界的许多有识之士预言:“Java 语言将会成为未来程序设计的主流语言”,全世界已经达成了一个共识:“Java 将会带去一场软件革命”。2000 年,Java 语言更是被评为 20 世纪十大科技成果之一。在 20 世纪堪称人类伟大的科学技术成果有原子物理与核物理、生物技术、航空航天技术、电子技术、信息通信技术、工程应用等,与它们相比较,Java 语言能够获得如此殊荣,是十分了不起的。或许 Java 语言的价值到目前还没有全部体现出来,真正地了解和认识 Java 语言恐怕还需要一段时间。

Java 这个词从其狭义上说指的是作为程序设计语言的 Java,从广义上说指的是 Java 开发工具和开发环境。

1.1.2 Java 带来的影响

Java 虽出现的时间不长,但已被业界接受,IBM、Apple、DEC、Adobe、SiliconGraphics、HP、Oracle、Toshiba、Netscap 和 Microsoft 等大公司已经购买了 Java 的许可证。Microsoft 还在其 Web 浏览器 Explorer 3.0 版中增加了对 Java 的支持。

另外,众多的软件开发商也开发了许多支持 Java 的软件产品。如 Borland 公司的基于 Java 的快速应用程序开发环境 Latte; Metrowerks 公司和 NaturalIntelligence 公司分别开发的基于 Machintosh 的 Java 开发工具; Sun 公司的 Java 开发环境 JavaWorkshop; Microsoft 也开发出系列 Java 产品。数据库厂商如 Illustra、Sybase、Versant、Oracle 都在开发支持 HTML 和 Java 的 CGI(Common Gateway Interface)。在以网络为中心的计算机时代,不支持 HTML 和 Java,就意味着应用程序的应用范围只能限于同质的环境。

Intranet 正在成为企业信息系统最佳的解决方案。它的优点表现在:便宜、易于使用和管

理。用户不管使用何种类型的机器和操作系统,界面是统一的 Web 浏览器,而数据库、Web 页面、应用程序(用 Java 编写的 Applet)则存在 WWW 服务器上。开发人员只需维护一个软件版本,管理人员省去了为用户安装、升级、培训之苦,用户则只需一个操作系统,一个 Internet 浏览器足矣。

我们可以设想未来的计算方式,如每个 HomePage 的实质是一个多媒体应用程序,这些程序用 Java 来开发。Java 应用程序运行在异质的机器、异质的操作系统之上,甚至于电冰箱、烤面包箱、防盗电子设备之中,用 Internet 把所有的电子设备连接起来,通过 TCP/IP 进行信息的交流。Java 应用程序之间既可以交换消息,也可以交换程序(一个 Java 的小应用程序 applet)。或许有一天,我们可以在 Netscape 浏览器里查看电冰箱的温度,向烤面包箱发一个电子邮件。

Java 的出现是计算机信息交换的一个重要里程碑。在单机时代,程序进程之间靠共享存储进行变量交换;在网络时代,运行在不同宿主机上的程序按网络协议进行无格式的消息(二进制字符流)交换,消息的语义由交换程序双方维护;在 Java 时代,运行在网络上的程序进程交换的是小应用程序(Applet)。小应用程序是什么?它是一个对象,由一组属性和方法构成,是一个可执行的实体;不仅有数据的状态,而且有定义在数据上的操作。未来可能进行代理(Agent)交换,代理有一定的智能性,那便是信息交换的更高级阶段。

随着 Internet 的崛起、Java 的诞生,巨型、臃肿的应用软件开始向小型化发展,由众多“生活”在 Internet 上的小应用程序(Applet)相互协作完成信息的处理与传递。Java 会加速应用软件的小型化、网络化的趋势。随着 Internet 的发展,软件必然是面向“大众”,而不是“少数人”的奢侈品,薄利多销将是网络应用软件的重要特点。分布式对象技术保证了用多少下载多少。

Java 连同 Internet,WWW 正在改变应用软件的开发和使用方式,一切都要围绕着网络,围绕着平台无关。很多人认为,Word、Excel 等传统的信息处理工具都必然走向萎缩,因为它们是单机时代的产物。信息的价值在于使用和共享,Internet 和 Web 是信息的使用和共享最快捷、最便宜的方式,Word 将演化成为 Web 写作工具,Excel 则将演化成 Web 上的电子表格。

1.1.3 Java 的特点

Java 语言自产生以来,Sun 公司致力于使 Java 成为一个简单的(Simple)、面向对象的(Object Oriented)、分布式的(Distributed)、解释的(Interpreted)、健壮的(Robust)、安全的(Secure)、结构中立的(Architecture Neutral)、可移植的(Portable)、高效能的(High Performance)、多线程的(Multithreaded)、动态的(Dynamic)程序语言。正是由于它的这些特点的存在,使得 Java 成为目前使用最为广泛的网络编程语言。

(1) Java 是编写 Web 页面的一种语言,就如同 HTML 和 VRML 一样。

事实上,Java 并不像是 HTML 一类的描述语言(Description Language),而是一种编程语言(Programming Language)。描述语言是标明内容和位置,而编程语言则是描述一种产生结果的过程。

(2) Java 代码是可移植的,但 C、C++ 就不是。

Java 源代码(Source Code)的可移植性是比较 C 语言好一些,差别在于 Java 的字节码。Java 源码编译后,就能在所有的机器上执行,条件是机器上要有 Java 解释器。

(3) Java 不适合于建立大型的应用程序。

如果说 Java 适合做大型程序,则 Java 就不适合应用于 Web 浏览器了。第一个商业用的

Java Applets 并不是全部用 Java 开发的,Java 只是用来处理界面部分。所以说 Java 对于开发大型的应用程序有一定的难度。

(4) Java 语言容易学习和使用,不像 C、C++ 和其他程序语言。

Java 是一种编程语言。Java 比起 C、C++ 或许容易学习,但它也是一种编程语言,要想真正掌握它,也必须付出相当的努力。

(5) Java 能被拓展从而能在机器上执行任何事情。

理论上说,Java Applet(Java 小应用程序)能做任何事情,如模拟 3D VRML 模型、播放音频、视频等。但事实上,一个小应用程序(Applet)仅能在那一个页面上被执行,而没有办法在除此以外的其他页面上执行。同时,Java 的拓展也受限于 Java 类库的功能。

(6) Java 是解释执行的,Basic 也是解释执行的,因此 Java=Basic。

虽然 Java 的确是使用解释器,但事实上,Java 与 C 或 C++ 等完全编译语言较为相近,只不过编译所产生的文件不一致而已。

1.1.4 不同版本的 Java

下面对 Java 目前不同的版本做一个简单介绍。

1. Java Developers Kits(JDK)

Java 开发工具箱,是 Sun 公司发布的 Java 最初版本。自 Java 1.2 发布后,Java 改名为 Java 2,相应的 JDK 也就改名为 J2SE。

2. Java SDK Standard Edition(J2SE)

J2SE 主要用于开发一般台式机应用程序。我们平时所说的 JDK 就是指 J2SE,而我们学 Java 就是从学习 J2SE 开始的,这也就是本书主要讲述的内容。

3. Java SDK Enterprise Edition(J2EE)

用于开发分布式的企业级大型应用程序。其中的核心是 Enterprise Java Beans(EJB,分布式 Java 组件,又称为企业 Java Bean)的开发。

4. Java SDK Micro Edition(J2ME)

用于开发掌上电脑、手机等移动通信设备上使用的应用程序。现阶段并不是所有的移动设备都支持 Java,只有具备 J2ME 运行环境(JVM+J2ME API)的设备才能运行 Java 程序。J2ME 的集成开发工具(通常都带有一些仿真器)有 Sun 的 J2ME Wireless Toolkit、IBM 的 Visual Age Micro Edition 等。

1.1.5 Java Application 与 Java Applet

在 J2SE 中可以编写两种 Java 程序:Java Application 与 Java Applet。标准的 Java 语言程序称为 Java Application,是可以独立运行的完整的应用程序,它运行在 Java 运行时系统中,有自己固定的运行入口点和出口点,与传统的高级语言程序比较接近。Java Applet 是 Java 语言为了丰富 WWW 页面设计而开发的一种应用,Applet 是 Java 首创的一个英文单词,由 Application 的前三个字母和表示“碎片”的-let 词尾构成,其字面意思是“Java 小应用程序”,可以被嵌入到 HTML 中,成为主页的一部分发布到 WWW 页面上,由 Java 兼容浏览器控制运行。目前还是一种 Java 所独有的 Internet 网络应用程序,这也是 Java 的一大特色。Java Application 与 Java Applet 都遵循相同的语法,只是由于 Java Applet 的特定用途而在运行环境和运行步骤上与 Java Application 有所不同。Java Applet 通常用来实现 WWW 页面的图像、声音和动画等功能。