

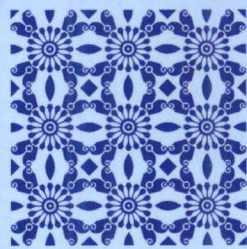
P

PROGRAMMING IN JAVA THIRD EDITION

Java程序设计教程

第3版

施霞萍 王瑾德 史建成 马可幸 张欢欢 编著



机械工业出版社
China Machine Press

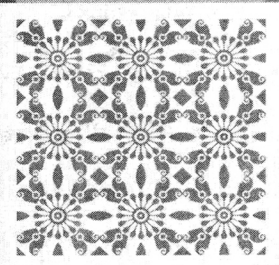
高等院校计算机教材系列

*P*ROGRAMMING IN JAVA THIRD EDITION

Java程序设计教程

第3版

施霞萍 王瑾德 史建成 马可幸 张欢欢 编著



机械工业出版社
China Machine Press

本书以程序设计初学者为对象,以程序设计的基本概念为起点,由浅入深、循序渐进地介绍 Java 程序设计语言的基本概念、方法和应用。内容包括结构化程序设计、面向对象程序设计、图形用户界面开发、异常处理机制、多线程程序设计、输入输出流以及 Java 在网络和多媒体技术中的应用。本书把概念和方法与详细的程序代码紧密地连接起来,并结合图示进行解析,使读者易学易用。针对每章的学习还配有实验和思考练习。

本书面向程序设计零基础的读者,即使没有教师指导,读者也可以自学本书的大部分内容。本书可以作为高等学校各专业学生学习 Java 程序设计语言的教材和参考书,是上海市高等学校计算机二级考试的指定参考书。

封底无防伪标均为盗版

版权所有,侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目(CIP)数据

Java 程序设计教程 / 施霞萍等编著. —3 版. —北京:机械工业出版社, 2012. 11

(高等院校计算机教材系列)

ISBN 978-7-111-40083-7

I. J… II. 施… 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(2012)第 243732 号

机械工业出版社(北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑:余洁

北京市荣盛彩色印刷有限公司印刷

2012 年 11 月第 3 版第 1 次印刷

185mm×260mm·18.5 印张

标准书号:ISBN 978-7-111-40083-7

定价:36.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

客服热线:(010) 88378991; 88361066

购书热线:(010) 68326294; 88376949; 68995259

投稿热线:(010) 88379604

读者信箱:hzjsj@hzbook.com

前 言

面向对象技术被认为是程序设计方法学的革命性突破，它已经被逐步应用于计算机应用开发的各个领域，并成为主流趋势。而 Java 程序设计语言正是面向对象技术成功应用在程序设计中的著名典范。Java 语言及编程技术是随着互联网应用的发展而被推广使用的，是目前国内外广泛应用的程序设计语言，其特有的性质使得它有别于以往其他程序设计语言。

为推动 Java 程序设计语言的学习和应用，并为配合《上海市高等学校计算机等级考试（二级）》Java 程序设计考试科目的设置，在上海市教育委员会高教处、上海市教育考试院的组织领导下，由多位具有多年丰富教学经验的教师集体编著了本书的第 1、2 版。多年来，本书的第 1、2 版受到了广大读者的欢迎，为此我们在前两版的基础上，重新对相关内容进行选择、补充和修改，使本书的内容更加丰富实用，学习更加方便。第 3 版与前两版最大的不同是加强了对本书内容重要知识点的归纳和总结，提供了更加系统和完整的实验练习。同时与本书配套的《Java 程序设计习题精析与实验指导》一书，不仅为读者提供了大量的习题和实验，并对样例进行了详细的讲解，其题型和样例都紧密配合上海市高等学校计算机等级考试，内容包括重点复习、习题解析、实验指导、练习题、综合练习等部分。

本书坚持在上海市高等学校计算机等级考试的 Java 程序设计考试大纲的指导框架下，以初学者为起点，由浅入深、循序渐进地介绍 Java 程序设计语言及应用的基本概念和基本方法。在内容上力争主次分明，避免繁琐细节的罗列，希望在有限的篇幅中让读者比较完整地掌握 Java 程序设计的思想和方法。

全书共分 11 章，前 3 章涵盖了 Java 语言在结构化程序设计中的基本方法和应用。如果你是一个没有学过任何程序设计语言的初学者，通过前 3 章的学习，你将学习所有程序设计语言所共有的概念和知识，同时还将学会最新的 Java 程序设计开发平台的应用。第 4 章全面讨论了面向对象程序设计的思想方法以及在 Java 语言中的实现，通过这部分的学习，你将对面向对象程序设计的思想在 Java 中的应用有比较完整的认识。第 5 章介绍了小应用程序和用 AWT、Swing 组件进行图形用户界面设计的方法，这部分的学习将使你获得专业化图形用户界面的设计能力。第 6、7、8 章分别介绍了 Java 特有的错误处理机制、多线程程序设计以及输入输出流的实现方法，掌握 Java 所具有的这些性能将为你设计出完整、实用的程序提供保证。第 9、10 章讨论了 Java 在网络和多媒体方面的应用，这既是 Java 综合应用的举例，也是 Java 实用性的体现，这将进一步提高你学习 Java 的兴趣。最后一章是实验练习，它是为配合各章的学习而设置的。学习程序设计语言，仅仅看书是不够的，一定要自己动手实践，所以你在使用本书时千万不要忘了上机练习这一步骤。每一章都有本章概要和思考练习。在最后的附录中提供了 2012 年上海市高等学校计算机等级考试——Java 程序设计考试大纲以及样题，便于你学习参考。

如果选用本书作为教材，可根据教学时数和学生基础有选择地使用各个章节。如果已学过其他程序设计语言，可以将前 3 章的内容简化，而加强后面章节的学习。本书建议教学时数为 72 ~ 80 学时。除教材外，我们提供教材中所有的程序代码和讲义。本书思考练习的部分答案见附录 1，实验练习的参考答案见配套教材。例题程序代码和讲义可以通过登录华章网站（www.hzbook.com）下载。

本书第3版由施霞萍、王瑾德、史建成、马可幸、张欢欢编著，由施霞萍通审、修改定稿。机械工业出版社华章分社的多位同志对本书的出版给予了充分的重视和周到的安排，使得本书得以在短时间内完成出版。本书在写作前的酝酿和整个写作过程中，得到了上海市教育委员会高教处和上海市教育考试院相关领导的指导和支持，为本书的定位和内容选择提供了方向。为此对一切曾经鼓励、支持和帮助过我们的组织、领导、朋友，表示真挚的感谢。

尽管作者都有10年以上的大学教龄并积累了许多程序设计方面的教学经验，但由于时间仓促和水平有限，书中难免有不妥之处，竭诚欢迎读者多提宝贵意见。电子邮箱：sxp@dhu.edu.cn。

施霞萍

2012年10月

教学建议

1. 适用对象

财经、理工、医学、农林等专业学生。

2. 建议学时数

72 ~ 80 学时，其中实验课至少为 32 学时。

教学内容	80 学时教学分配			72 学时教学分配		
	课堂教学	实验教学	课外作业	课堂教学	实验教学	课外作业
第 1 章 Java 程序设计概述	2	2		2	2	
第 2 章 Java 程序设计的基本概念	2	2	1	2	2	1
第 3 章 Java 的结构化程序设计	8	8	4	4	4	2
第 4 章 Java 的面向对象程序设计	8	8	4	8	8	4
第 5 章 Java 的图形用户界面	6	6	3	6	6	3
第 6 章 Java 的异常处理	2	2	1	2	2	1
第 7 章 Java 的多线程程序设计	4	4	2	4	4	2
第 8 章 Java 的输入输出流	4	4	2	4	4	2
第 9 章 Java 的网络应用	2	2	1	2	2	1
第 10 章 Java 的多媒体应用	2	2	1	2	2	1
合计	80		19	72		17

3. 实验环境建议

(1) 硬件环境

CPU : Pentium III 550 MGHZ 以上

内存 : 128 MB 以上

硬盘 : 20 GB 以上

(2) 软件环境

操作系统 : Windows 2000 及以上

编程环境 : Eclipse Helios

Java 运行环境 (JRE) : JDK Java SE 6 Update 23

目 录

前 言

教学建议

第 1 章 Java 程序设计概述 1

1.1 Java 与程序 1

1.1.1 概述 1

1.1.2 Java 的发展 2

1.2 Java 语言的特点 3

1.3 Java 程序介绍 4

1.3.1 Java 虚拟机 4

1.3.2 Java 开发工具 4

1.3.3 Java 的 API 类库与 API 文档 5

1.3.4 Java 资源的获取、安装 6

1.3.5 Eclipse 应用平台简介 9

1.4 Java 程序结构及开发过程概述 11

1.4.1 Java 程序结构 11

1.4.2 最简单的 Java 程序 13

1.4.3 Java 程序的开发过程 14

1.5 本章概要 20

1.6 思考练习 21

第 2 章 Java 程序设计的基本概念 22

2.1 数据类型与标识符 22

2.1.1 数据类型 22

2.1.2 标识符 22

2.2 常量 23

2.2.1 整型常量值 23

2.2.2 实型常量值 24

2.2.3 字符型和字符串常量值 24

2.2.4 布尔型常量值 24

2.2.5 常量的定义和使用 24

2.3 变量 26

2.3.1 变量概述 26

2.3.2 变量的定义及赋值 27

2.3.3 变量的作用域 28

2.3.4 变量的默认值 28

2.4 运算符与表达式 29

2.4.1 赋值运算符与赋值表达式 29

2.4.2 算术运算符与算术表达式 29

2.4.3 关系运算符与关系表达式 31

2.4.4 逻辑运算符与逻辑表达式 32

2.4.5 复合赋值运算符 33

2.4.6 其他运算符 34

2.4.7 运算符的优先级与结合性 34

2.5 数据类型的转换 35

2.6 本章概要 37

2.7 思考练习 37

第 3 章 Java 的结构化程序设计 39

3.1 顺序结构及基本语句 39

3.1.1 赋值语句 39

3.1.2 输入语句 40

3.2 选择结构语句 41

3.2.1 if 语句 42

3.2.2 if-else 语句 43

3.2.3 if-else if 语句 44

3.2.4 if 语句的嵌套 45

3.2.5 switch 语句 46

3.3 循环结构语句 48

3.3.1 for 语句 48

3.3.2 while 语句 49

3.3.3 do-while 语句 50

3.3.4 循环结构语句的嵌套 51

3.4 转移语句 52

3.4.1 break 语句	52	4.4.2 访问权限的设置	80
3.4.2 continue 语句	53	4.4.3 类成员（静态成员）	82
3.5 数组	54	4.5 类的继承	85
3.5.1 一维数组的声明与引用	55	4.5.1 继承的基本概念	85
3.5.2 数组的赋值	56	4.5.2 子类的创建	85
3.5.3 一维数组程序举例	56	4.5.3 null、this、super 对象运算符	87
3.5.4 二维数组的声明及引用	58	4.5.4 最终类和抽象类	89
3.5.5 数组的复制	60	4.6 类的多态性	90
3.5.6 字符串处理	60	4.6.1 方法的重载	90
3.6 方法	64	4.6.2 方法的覆盖	92
3.6.1 Java 的程序模块化	64	4.6.3 前期绑定和后期绑定	93
3.6.2 方法的定义及调用	64	4.7 接口	93
3.6.3 参数的传递	65	4.7.1 接口的声明	93
3.6.4 作用域	66	4.7.2 接口的实现	94
3.6.5 return 语句	67	4.8 包	95
3.6.6 方法的嵌套调用	68	4.8.1 Java 的类和包	95
3.6.7 递归	69	4.8.2 引用 Java 定义的包	96
3.7 本章概要	70	4.8.3 自定义包	97
3.8 思考练习	70	4.8.4 包和访问权限	99
第 4 章 Java 的面向对象程序设计	72	4.9 本章概要	99
4.1 面向对象程序设计概述	72	4.10 思考练习	99
4.1.1 面向对象程序设计的目的	72	第 5 章 Java 的图形用户界面	101
4.1.2 类和对象	72	5.1 Applet 概述	101
4.1.3 面向对象程序设计的核心技术	72	5.1.1 一个简单的 Applet 例子	101
4.1.4 Java 的面向对象技术	73	5.1.2 Applet 的安全模型	102
4.2 类的创建	73	5.1.3 java.applet.Applet 类与其他类的关系	102
4.2.1 类的声明格式	73	5.1.4 Applet 的生命周期	103
4.2.2 成员变量	74	5.2 java.awt 与图形用户界面	105
4.2.3 成员方法	74	5.2.1 标签和文本域	105
4.3 对象的创建和使用	76	5.2.2 Java 中的事件处理机制	107
4.3.1 创建对象	76	5.2.3 按钮	113
4.3.2 构造方法和对象的初始化	76	5.2.4 布局	114
4.3.3 对象的使用	78	5.2.5 面板	116
4.3.4 对象的销毁	79	5.2.6 文本区域	118
4.4 类的封装	80	5.2.7 复选框和单选钮	118
4.4.1 封装的目的	80		

5.2.8 下拉列表	120	7.3 线程的状态与控制	184
5.2.9 列表	122	7.3.1 线程的状态	184
5.2.10 窗口与菜单	125	7.3.2 对线程状态的控制	185
5.2.11 对话框	135	7.4 线程的优先级和调度	188
5.3 Swing	137	7.4.1 线程的优先级	188
5.3.1 Swing 的特点	137	7.4.2 线程的调度	188
5.3.2 Swing 类的继承关系	138	7.5 线程组	188
5.3.3 Swing 中的容器	139	7.5.1 线程组概述	188
5.3.4 Swing 中的常用组件	150	7.5.2 ThreadGroup 类	188
5.3.5 Swing 中的事件	161	7.6 线程的同步	189
5.4 二维图形设计	163	7.6.1 线程的同步机制	189
5.4.1 二维图形的坐标系统	163	7.6.2 共享数据的互斥锁定	191
5.4.2 字体	163	7.6.3 数据传送时的同步控制	193
5.4.3 颜色	164	7.6.4 死锁	195
5.4.4 绘图	164	7.7 本章概要	197
5.4.5 Timer 与 TimerTask 类	167	7.8 思考练习	197
5.5 本章概要	170	第 8 章 Java 的输入输出流	198
5.6 思考练习	170	8.1 流的基本概念	198
第 6 章 Java 的异常处理	172	8.1.1 输入输出流与缓冲流	198
6.1 异常和异常对象	172	8.1.2 Java 的标准输入输出	198
6.2 异常的捕获与处理	173	8.1.3 java.io 包中的数据流	199
6.3 try 语句的嵌套	175	8.2 字节流	200
6.4 throw 语句	177	8.2.1 InputStream 和 OutputStream 类	200
6.5 throws 语句	179	8.2.2 文件字节流与文件的读写	201
6.6 使用异常处理的准则	179	8.3 字符流	203
6.7 本章概要	180	8.3.1 Reader 和 Writer 类	203
6.8 思考练习	180	8.3.2 文件字符流与文件的读写	204
第 7 章 Java 的多线程程序设计	181	8.3.3 字符缓冲流与文件的读写	205
7.1 线程的概念	181	8.4 文件类与文件的操作	206
7.1.1 进程和线程	181	8.4.1 文件类 File	206
7.1.2 线程和多任务	181	8.4.2 文件过滤器	207
7.1.3 Java 对多线程的支持	181	8.4.3 文件对话框与文件的操作	209
7.2 线程的创建	181	8.5 文件的随机读写	213
7.2.1 Runnable 接口	182	8.5.1 RandomAccessFile 类	213
7.2.2 Thread 类	182	8.5.2 RandomAccessFile 的 构造方法	213
7.2.3 创建线程的方法	182		

8.5.3 RandomAccessFile 的方法	213	9.5 本章概要	238
8.6 DataInputStream 和 DataOutput- Stream 与文件的操作	214	9.6 思考练习	238
8.6.1 数据流 DataInputStream 和 DataOutputStream 类	214	第 10 章 Java 的多媒体应用	240
8.6.2 使用 DataInputStream 和 Data- OutputStream 类对文件操作	215	10.1 图像显示	240
8.7 本章概要	216	10.2 动画实现	242
8.8 思考练习	216	10.3 声音播放	246
第 9 章 Java 的网络应用	218	10.4 本章概要	249
9.1 网络的基本概念	218	10.5 思考练习	249
9.1.1 IP 地址和端口号	218	第 11 章 实验练习	251
9.1.2 URL	219	实验一 Java 程序的开发过程与 开发环境	251
9.1.3 TCP 与 UDP	219	实验二 Java 程序设计的基本概念	251
9.1.4 Socket	220	实验三 Java 的结构化程序设计	253
9.2 URL 的使用	220	实验四 Java 的面向对象程序设计	256
9.2.1 使用 URL 的方法	220	实验五 Java 的图形用户界面	258
9.2.2 应用举例	222	实验六 Java 的异常处理	264
9.3 Socket 的应用	227	实验七 Java 的多线程程序设计	266
9.3.1 TCP 套接字通信基本步骤	227	实验八 Java 的输入输出流	266
9.3.2 服务器端程序设计举例	229	实验九 Java 的网络应用	270
9.3.3 客户端程序设计举例	231	实验十 Java 的多媒体应用	271
9.4 网络安全管理	233	附录 1 部分参考答案	273
9.4.1 Java 的安全特性	234	附录 2 2012 年上海市高等学校计算机 等级考试(二级)——《Java 程序设计》考试大纲	275
9.4.2 缓存溢出	234	附录 3 上海市高等学校计算机等级 考试试卷(二级)——《Java 程序设计》(样卷)	279
9.4.3 竞争状态	235	参考文献	286
9.4.4 建立安全性策略	236		
9.4.5 安全基本原则	238		

第 1 章 Java 程序设计概述

1.1 Java 与程序

1.1.1 概述

程序设计是伴随着电子计算机的出现而产生的一门技术。简单地说，程序设计就是根据提出的任务，把计算机正确完成该任务而做的工作写成一种能让计算机直接或间接接受的语句的过程，整个任务所对应的一系列语句的集合即被称为一段程序。随着计算机科学的飞速发展，程序设计的方式和水平也在不断地改善与提高。作为计算机软件的一部分，程序设计所用的语言（被称为程序设计语言）与计算机硬件的发展一样，也有几个发展阶段。

1. 机器语言和汇编语言

在计算机问世的初期，人们直接使用计算机能够识别的二进制代码按一定的规则进行程序编写工作。这种用二进制代码表示的规则就是“机器语言”。由于机器语言表示形式不直观，语义单一，因而给编写程序带来很大的麻烦，阻碍了计算机的广泛应用。为此，软件设计者们用一些简单而又形象的符号来替代每一条具体的机器语言，这就形成了“符号语言”，也就是“汇编语言”。但计算机无法直接识别“符号语言”，所以从汇编语言到机器语言，中间要有一个翻译过程，这一过程由翻译程序——“汇编程序”来完成。机器语言和汇编语言是与具体的计算机（确切地说是与计算机指令系统）相关的，是为特定的机器服务的，所以被称为面向机器的语言。

2. 高级语言

人们在汇编语言的基础上，设想能否不考虑具体的机器，用一些接近于自然语言和数学公式的符号来描述自己的解题意图，以便通过各类机器对应的翻译程序就可以在各类机器上运行。这便出现了各种高级语言。20 世纪 80 年代初期，国内外比较通用的计算机语言有十几种，常见的且普遍应用的有 BASIC、FORTRAN、ALGOL、COBOL、PL/1、PASCAL 以及 C 等。在支持 16 位应用程序的 DOS 操作系统下，用上述的计算机语言编写的程序都是按事先设计的流程运行的，因而这些计算机语言被称为面向过程的程序设计语言。与汇编程序作用一样，利用高级语言编写的程序必须被翻译成机器语言才能由计算机执行，完成这一任务的程序称为“语言处理程序”。

“语言处理程序”分为两大类：解释程序和编译程序。解释程序逐句地接收所输入的用程序语言编写的程序（源程序），然后逐句翻译解释并执行源程序，大家所熟知的 BASIC 和 APL 等会话型语言就是采用解释方法运行的。编译程序（有些书上也把它称为编译系统）是把用高级语言编写的面向过程的源程序翻译成目标程序的一种语言处理程序（目标程序即为机器语言构成的程序）。20 世纪 80 年代后期曾经广泛应用的 FORTRAN、PL/1 以及 PASCAL、C 等语言皆采用编译方式实现。

3. 面向对象的程序设计语言

20 世纪 90 年代，Windows 操作系统以其新颖的图形用户界面、卓越的多任务操作系统性能

和高层次的软件开发平台而迅速风靡全球。在与用户的交流过程中，面向过程的程序设计语言的用户界面便显得有些“不合时宜”，随着计算机网络的诞生和发展，面向对象的程序设计语言应运而生。面向对象的程序设计是近年来出现的程序设计技术，它以一种全新的设计和构造软件的思维方法，开拓了程序设计方法史上的新世纪。Java 程序设计语言便是其中的佼佼者。

Java 是美国 Sun Microsystems 公司于 1995 年 5 月正式发布的程序设计语言，它的前身是 Sun Microsystems 公司为智能消费类家用电器研究而开发的项目一部分，但该项目的开发过程却并不顺利，直到 1993 年 Web 开始在 Internet 上盛行，开发小组试着将这一技术转移到 Web 网络上，没想到这一举动使 Java 在 Internet 上获得了空前的成功，使它成为 Web 世界富有创造性的工具。时至今日，尽管 Web 开发人员仍然在利用 Java 使站点更生动、活泼，但它已远远超过了 Web 技术的范围。

1.1.2 Java 的发展

Java 的发展过程如表 1-1 所示。

表 1-1 Java 语言的发展历史

时 间	描 述
1991 年	Sun Microsystems 公司进军消费电子产品（IA）市场
1991 年 4 月	Sun 成立“Green”小组，以 C++ 为基础开发新的程序设计语言，并将其命名为 Oak
1992 年 10 月	Green 小组升级为 First Person 公司，他们将 Oak 的技术转移到 Web 上，并把 Oak 改名为 Java
1993—1994 年	Web 开始在 Internet 上流行，使 Java 得以迅速发展并获得成功
1995 年 5 月	Sun Microsystems 公司正式发布 Java 与 HotJava 产品
1995 年 10 月	Netscape 与 Sun Microsystems 合作，在 Netscape Navigator 中支持 Java
1995 年 12 月	微软（Microsoft）IE 加入支持 Java 的行列
1996 年 2 月	Java Beta 测试版结束，Java 1.0 版正式诞生
1997 年 2 月	Java 发展至 1.1 版。第一个 Java 开发包 JDK（Java Development Kit）发布
1998 年 12 月	Java 升级至 1.2 版
2000 年 5 月	Java 升级至 1.3 版
2002 年 6 月	Java 升级至 1.4 版
2004 年 9 月	Java 升级至 1.5 版，此时其名称改为 Java 5.0 版
2006 年 12 月	Java 升级至 1.6 版，此时其名称改为 Java 6.0 版

严格地说，在 2010 年 Java 7.0 版也已问世，并且我们相信：它的版本仍将不断地更新。Java 的早期版本称为 JDK，1.2 版之后改名为 Java 2，Java 2 平台的发布被称为 Java 发展史的新里程碑，目前流行的很多 Java 教程包括本书都是基于 Java 2 技术的。

1999 年下半年，Sun Microsystems 公司重新组织 Java 平台的集成方法，即重新划分 Java 的应用平台，并将 Java 企业级应用平台作为发展方向。因此，现在 Java 的大家庭中已有三个主要成员：

1) Java ME——即 Java Micro Edition，用于嵌入式 Java 消费电子平台。无线通信、手机、PDA 等小型电子装置都可采用其作为开发工具及应用平台。

2) Java SE——即 Java（Software Development Kit）Standard Edition，是 Java 最通行的版本，是用于工作站、PC 的 Java 标准平台，因此也是本书应用程序的使用版本。

3) Java EE——即 Java Enterprise Edition（可扩展的企业应用 Java 平台），它提供了企业

e-Business 架构及 Web Services 服务，其深受广大企业用户欢迎之处是其开放的标准和优越的跨平台能力。

1.2 Java 语言的特点

Java 程序设计语言是新一代语言的代表，它强调了面向对象的特性，可以用来开发不同种类的软件，它具有支持图形化的用户界面、支持网络以及数据库连接等复杂的功能，Java 语言的主要特点如下：

1. 易于学习

Java 语言很简单，但这里所说的简单，主要是针对于熟悉类似 C++ 语言的程序设计人员来说的，因为它的语法与 C++ 非常相似，但是它摒弃了 C++ 中许多低级、困难、容易混淆、容易出错或不经常使用的功能，例如运算符重载、指针运算、程序的预处理、结构、多重继承以及其他一系列内容，这样便使其比其他许多编程语言更易于学习。利用 Java 语言还能够编制出非常复杂的系统，且运行时占用很少的内存资源。

2. 高效率的执行方式

用 Java 语言编辑的源程序的执行方法是采用先经过编译器编译、再利用解释器解释的方式来运行的。它综合了解释性语言与编译语言的众多优点，使其执行效率较以往的程序设计语言有了大幅度的提高。

3. 与平台无关性

Java 源程序经过编译器编译，会被转换成一种我们称之为“字节码 (byte-codes)”的目标程序。“字节码”的最大特点是可以跨平台运行，即程序设计人员们常说的“编写一次，到处运行”，正是这一特性使得 Java 得到迅速普及。

4. 分布式

所谓分布式主要是指数据分布和操作分布两层意思，数据分布是指数据可以分散在网络中不同的主机上，操作分布是指把一个任务分散在不同的主机上进行处理。Java 从诞生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，支持远程方法调用，便于开发网上应用系统。

5. 安全性

Java 平台采用了域管理方式的安全模型，无论是本地代码还是远程代码都可以通过配置的策略，设定可访问的资源域。当 Java 字节码进入专门处理该内容的程序（即后面所说的解释器）时，首先必须经过字节码校验器的检查，然后 Java 解释器将决定程序中类的内存布局，随后，类装载器负责把来自网络的类装载到单独的内存区域，避免应用程序之间相互干扰破坏。最后，客户端用户还可以限制从网络上装载的类只能访问某些文件系统。上述几种机制结合起来，使得 Java 成为安全的编程语言。

6. 可靠性

Java 程序设计语言要求显式的方法声明，保证了编译器可以发现方法调用错误；Java 不支持指针，杜绝了对内存的非法访问；Java 的垃圾自动收集机制，防止了程序员可能忘记释放原来分配的内存或者释放了其他正在使用的内存而引起的系统严重出错……这些措施保证了 Java 程序的可靠性。

7. 多线程

Java 的多线程机制使应用程序中的线程能够并发执行，且其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，这样就很容易在网络上实现实时交互行为。

8. 丰富的 API（Application Program Interface）类库

Java 开发工具包中的类库包罗万象，应有尽有，程序员的开发工作可以在一个更高的层次上展开，这也是 Java 受欢迎的重要原因之一。Java 同时为用户提供了详尽的 API 文档说明。

1.3 Java 程序介绍

1.3.1 Java 虚拟机

前面已经说过，Java 语言编辑的源程序的执行方法是先经过编译器编译，再利用解释器解释的方式来运行的。Java 程序的开发及运行周期如图 1-1 所示。

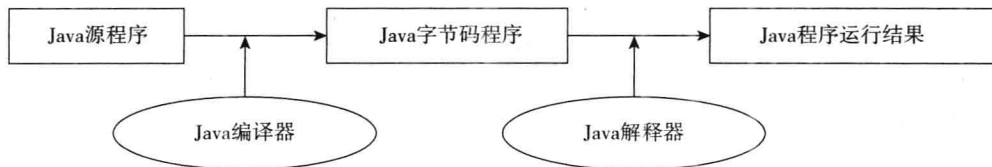


图 1-1 Java 程序的开发及运行周期

基于 Java 运行的平台无关性特点，我们可以直观地理解：在我们的常规计算机运行环境中，一定存在多种类型的 Java 解释程序以帮助我们运行 Java 程序。任何一种可以运行 Java 程序（即可以担任 Java 解释器）的软件都可以称为 Java 虚拟机（Java Virtual Machine, JVM），因此，诸如浏览器与 Java 的一部分开发工具等皆可看做 JVM。当然我们可以把 Java 的字节码看成是 JVM 所运行的机器码。

1.3.2 Java 开发工具

为了正确且顺利地创建、运行和调试 Java 程序，我们不仅仅需要解释器，还需要编译器和其他一系列工具，它们都由一系列程序文件（软件）组成。这些文件的集合称为 Java 开发工具。

1.2 节中介绍过：任何支持编写 Java 程序的环境都会为编写程序的人员提供很多的帮助内容，从这层意义上说，再用“编写”程序这样的词汇便不很妥当了，因此，我们用“开发”两字来代替“编写”是极具现实意义的，随着学习的进一步深入，大家对这一点一定会有更深的体会。由 Java 的开发工具所组成的帮助用户开发 Java 程序的环境称为 Java 的集成开发环境（IDE）。

在 Java 所有的开发工具中，最基本的称为开发工具包（Software Development Kit），简称 SDK，有些参考书称此开发工具包为 JDK。它是由 Sun Microsystems 公司所开发的一套 Java 程序开发软件，该软件中包含一个叫做 JRE 的子软件，称为 Java 的运行环境，因为当前运用很普遍的 Java 程序开发工具——Eclipse 的运行仅需要 JRE 的支持，JRE 可以在 Sun 公司的网站上单独下载。

Java 开发工具包可以在很多网站上免费下载，比较权威的下载可从 <http://www.oracle.com/>

technetwork/java/javase/downloads/index.html（如图 1-2 所示）获得，这是因为在 2010 年 Oracle 公司收购了 Sun 公司，但该公司继续为用户提供 Java 开发工具包的下载服务。

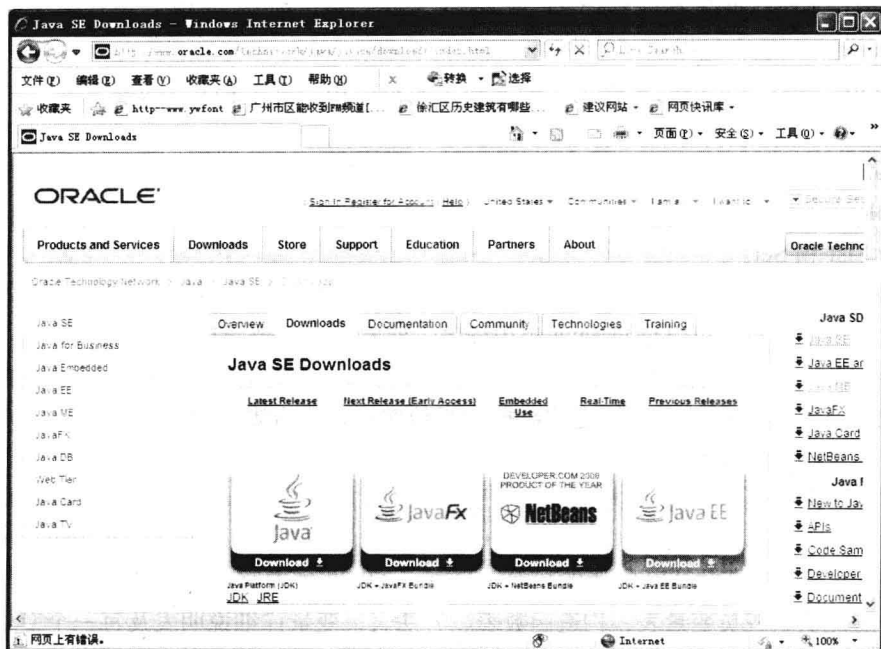


图 1-2 Oracle 公司 Java 开发工具包下载页面

若要安装 Java 开发环境，必须在其下层安装支持该环境的操作系统，Java 产品面向的主流平台有 Solaris、Macintosh、Windows 以及 UNIX 等。由于在大家学习的过程中，所接触的大多是 Windows 操作系统，因此我们向大家介绍的 Java 集成开发环境都以 Windows 作为其操作系统。

1.3.3 Java 的 API 类库与 API 文档

在开发 Java 程序时，需要设计和构造类集合。当程序运行时，对象从那些类进行实例化，并按照需要使用。如果你先前使用过其他程序设计语言进行程序设计，那么你一定知道需要得到类似该程序设计语言的操作手册之类的工具，因为它们可以帮助你获得很多系统支持的、可以直接在你的程序中调用的小程序，例如标准函数等，一切工作并不需要你都从零开始。作为 Java 程序员，其主要任务就是创建正确的类集合，以完成程序需要完成的工作。非常幸运的是，任何 Java 开发工具包中都会给出一套标准的类库，这些类为执行大部分的编程任务提供了方法和接口。类库被组织成许多包，每个包又包含一些子包和多个类，形成树型结构的类层次，其中包括核心包 java、扩展包 javax 和 org 等。

下面简单介绍一些重要的包及其类：

1) java.lang——这个包包含了一些形成语言核心的类，提供了类似 Character、Integer 和 Double 这样的封装类。它还提供了系统标准类，如 String 和 StringBuffer。Java 编辑器总是自动装载这个包。因而一般不必显式导入 java.lang 中的任何类。这个包中的许多类在本书的其他章节都还将叙述。

2) java.applet——这个包提供了创建 Java Applet 的途径，Java Applet 运行在 Web 浏览器下，

通常通过 Internet 下载。

3) java.awt——它是由许多组成 Java 的抽象视窗工具 (AWT) 的类所组成的包, 它提供了基于类的图形用户界面, 可以为 Java Applet 和应用程序编程提供视窗、按钮、对话框及其他控件。

4) java.net——这个包提供了网络、套接字处理器和 Internet 实用工具类。

5) java.io——这个包中的类提供了输入输出服务, 用于读出和写入文件数据, 进行键盘输入和打印输出。

6) java.util——这个包包含为任务设置的实用程序类和集合框架类, 每一个 Java 应用程序和 Java Applet 可能至少会用到这个包中的一个类。另外它还提供了 Collection 接口和它的实现容器类, 如 List 和 Set。

7) java.rmi——远程方法启用包, 在这个包中的类提供了通过远程接口控制的分布式代码的支持。通过该包中的类, 可以创建 Java 应用程序, 使它的不同部分在不同的系统中一起运行。

8) java.sql——这个包提供了结构化查询语言数据库字段类型和方法的实现。根据系统的不同, 这个包的类可能会通过一个特定的数据库系统实现, 或者默认时通过 ODBC (开放数据库连接) 标准的直接映射实现。

Java 的 API 文档是告诉你上述内容的使用方法的文档, 是 Java 程序开发的最好帮手, 当你开始从事 Java 程序开发而需要某些工具时, Oracle 公司的网站 <http://download.oracle.com/javase/6/docs/api/> 下提供了所有有关 API 类库的信息及链接, API 文档内容主要包括: 类层次结构、类及其一般目的的说明、成员变量表、构造函数表、方法表、变量详细说明表及每一个变量使用目的的详细描述、构造方法的详细说明及进一步的描述等。为了方便使用, 我们通常建议大家将该网页地址放置到浏览器的收藏夹中, 以随时查阅。它在浏览器下的界面形式如图 1-3 所示。

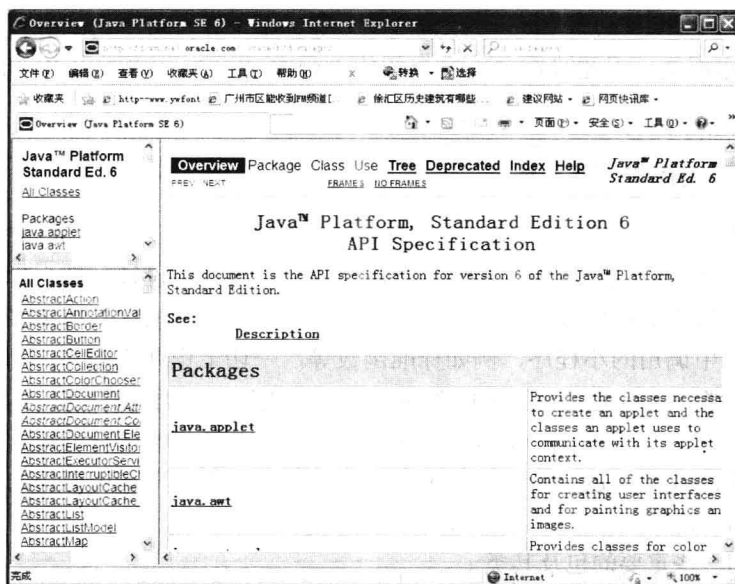


图 1-3 Oracle 公司 Java API 文档浏览页面

1.3.4 Java 资源的获取、安装

开发 Java 应用程序必须以安装 Java 程序开发工具为前提, 支持 Java 程序运行的平台除了前面介绍过的 SDK 之外, 还有很多其他软件环境, 比如 Borland 公司的产品 Jbuilder, 同样是一

个技术相当细腻的产品，支持很多控件，使用 Jbuilder 编写的 Java 软件完全可以脱离该平台运行。Jbuilder 的个人版和企业版均可以在 Borland 公司的网站下载安装。由于 Jbuilder 支持许多的技术，因此该软件在运行时将占用相当多的计算机资源。而我们下面将向大家介绍的是当前较流行，相对比较方便、快捷、功能齐全的 Java 集成开发环境平台软件——Eclipse。

Eclipse 是一个综合开发环境（Integrated Development Environment, IDE），是一个功能完整且成熟的软件，由 IBM 公司于 2001 年首次推出。大家可以在其官方网站 <http://www.eclipse.org> 免费下载该软件的压缩文件，释放到文件夹即可使用。在这里我们介绍的是 Eclipse IDE for Java Developers 版本。本书中的所有程序都是用这个版本的工具测试的。

Eclipse 是一个开放源代码、基于 Java 的可扩展开发平台。“开放源代码”的意思是使用者能够取得软件的原始码，并可适当修改和传播这个软件。有人非常形象地将 Eclipse 比喻成软件开发者的“打铁铺”，它备有火炉、铁砧与铁锤，就像铁匠会用现有的工具打造新的工具，程序员也能用 Eclipse 打造新工具来开发软件——这些新工具可扩充 Eclipse 的功能。

就 Eclipse 本身而言，它只是一个框架和一组称为平台核心的服务程序，用于通过插件和组件构建开发环境。平台核心的任务是让每样东西都动起来，并加载所需的外挂程序。当启动 Eclipse 时，先执行的就是这个服务程序，再由这个程序加载其他外挂程序。Eclipse 附带了一个包括 Java 开发工具（Java Development Tools, JDT）的标准插件集。但 Eclipse 提供下载的压缩包中并不包含 Java 运行环境，因此如果要使用 Eclipse 来开发 Java 程序，需要用户自己另行安装 JRE，并且在操作系统的环境变量中指明 JRE 中 bin 的路径。在这里，我们将向大家推荐的 Java 运行环境为“Java SE Runtime Environment 6u24”版。下面将详细向大家介绍如何从网上获取并安装 Java 开发工具。

1. 下载 JRE

在浏览器窗口的地址栏中输入 Oracle 公司的网址：<http://www.oracle.com/technetwork/java/javase/downloads/index.html> 得到如图 1-2 所示的下载页面。按住该窗体右侧滚动条的滑块往下拉，得到如图 1-4 所示的页面。

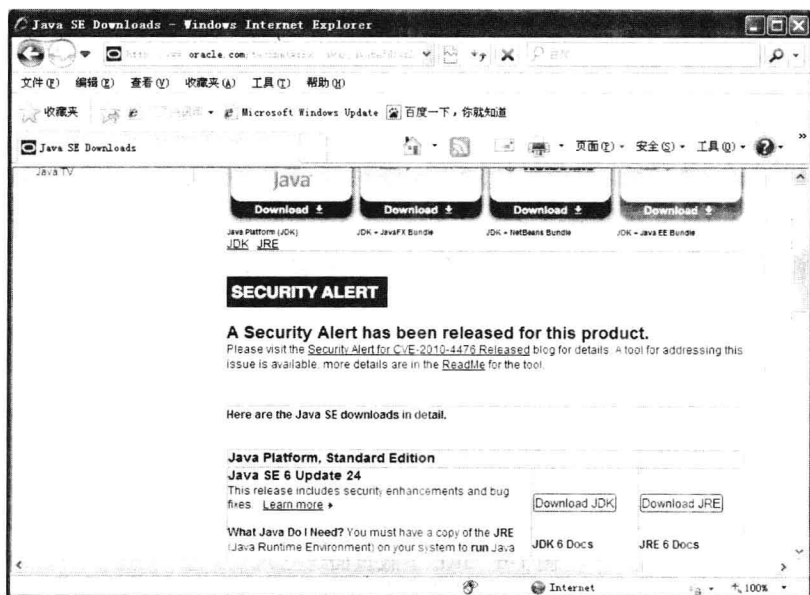


图 1-4 JRE 下载页面（一）