

高等院校计算机教材系列

第2版

Java程序设计教程

施霞萍 张欢欢 编著
王瑾德 马可幸

为教师提供教辅



机械工业出版社
China Machine Press

TP312
1423=2

2006

高等院校计算机教材系列

本书是“高等院校计算机教材系列”中的一本，主要介绍Java语言及其在Web开发中的应用。全书共分10章，主要内容包括：Java语言概述、Java语言基础、Java类与接口、Java异常处理、Java多线程、Java网络编程、Java数据库编程、Java图形用户界面、Java Swing、Java Applet等。本书可作为高等院校计算机专业及相关专业的教材，也可供从事Java开发的工程技术人员参考。

第2版

Java程序设计教程

施霞萍 张欢欢 编著
王瑾德 马可幸

图书在版编目(CIP)数据

Java语言程序设计教程 / 施霞萍, 张欢欢编著. — 北京: 机械工业出版社, 2006.9

(高等学校计算机专业系列)

ISBN 7-111-19917-2

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(2006)第113991号

机械工业出版社CIP数据核字(2006)第113991号

2006年10月第1版CIP数据核字(2006)第113991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

北京: 机械工业出版社



机械工业出版社
China Machine Press

本书以基本概念为起点,由浅入深、循序渐进地介绍 Java 程序设计的基本概念、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、错误处理机制、多线程设计、输入输出流等 Java 多种技术和应用。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,使读者易学易用。本书提供了适量课后练习题以夯实基础,在最后附录中提供了上海市高等学校计算机等级考试大纲和 2005 年的试卷,以方便备考。本书还提供了学习和教学的配套材料,可以到下列网址下载: www.hzbook.com。

本书面向没有程序设计基础的读者,即使没有教师教授,也可以自学本书的大部分内容。可以作为高等院校各专业学生学习 Java 程序设计基础的教材和参考书。

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 第2版/施霞萍等编著. -北京:机械工业出版社,2006.9
(高等院校计算机教材系列)
ISBN 7-111-19971-5

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 (2006) 第 113691 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:杨庆燕

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2006 年 9 月第 2 版第 1 次印刷

184mm×260mm·20.75 印张

定价:30.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线:(010) 68326294

前 言

Java 程序设计语言及编程技术是随着互联网的发展而被推广应用的,它是目前国内外广泛使用的程序设计语言。Java 语言是面向对象技术成功应用在程序设计语言中的著名典范,它的内容与特点与以往的其他程序设计语言有很大的不同。面向对象技术被认为是程序设计方法学的一场革命,它已经逐步替代面向过程的程序设计技术,成为计算机应用开发领域的主流趋势。

为推动 Java 程序设计语言的应用普及,同时为配合《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 版,并于 2004 年 9 月出版。两年以来,本书第 1 版受到了广大读者的欢迎。为此我们在第 1 版的基础上,重新对有关的内容进行选择、补充和编排,使本书的内容更加丰富实用,学习更加易用流畅。本版与第 1 版最主要的不同是添加和补充了与目前最新的考题形式相一致的练习题以及编程题,同时提供了部分参考答案。但不论怎么改变,本书仍然坚持在上海市高等学校计算机等级考试的 Java 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,避免烦琐细节的罗列,但又把 Java 的一些先进内容和思想方法介绍给大家,希望在有限的篇幅中使读者比较完整地掌握 Java 程序设计的思想和方法。

当然,学好程序设计语言仅仅看书是不够的,另一个重要步骤是动手实践。为此,本书在讲解概念和方法的同时,配有大量具有说服力的、浅显易懂的例子进行演示,读者在使用本书时千万不要忘了上机练习这一过程。您可以直接从网上下载书本上的例子来加深对知识点的理解和认识,也可以使用结合本书的内容所出的练习题进行实践。除直接上机调试外,我们还提供编程题的参考答案帮助您学习。

全书共分 10 章,前 3 章主要涵盖了 Java 在结构化程序设计中的基本概念和应用。如果您是一个没有学过任何程序设计语言的初学者,您将学会程序设计语言所共有的概念和知识,同时还会学习和了解到最新的 Java 开发平台。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,您对面向对象程序设计的思想在 Java 中的应用就会有比较完整的认识。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,其中 swing 组件在技术上要比 awt 组件先进。完成这部分的学习将使您达到能够设计出专业化图形用户界面的能力。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您设计出完整、实用的程序提供保证。最后两章讨论了 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。这将进一步提高您学习 Java 的兴趣。在最后的附录中,提供了上海市高等学校计算机等级考试——Java 程序设计考试大纲及最新的考题,便于您学习参考。

为了使您更好、更方便地学习和掌握 Java 程序设计方法,除教材外,我们提供教材中所有的程序代码和部分练习题答案,其中选择题和填空题的答案见附录 1,例题程序代码和编程题

答案,您可以到下面的网址下载: www.hzbook.com。为帮助教师的教学,我们将提供教学课件演示,需要者可直接到上面的网址获取。

本书第2版由施霞萍、张欢欢、王瑾德、马可幸编著,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重视和周到的安排,使本书得以在短时间内完成出版。对一切曾经鼓励、支持和帮助过我们的领导、组织、朋友,在此表示真挚的感谢。

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

2006年8月

作者介绍



施霞萍 1982年和1984年先后毕业于东华大学(原中国纺织大学)电气自动化专业(本科)和工业自动化专业(研究生)。现为东华大学计算机学院副教授、硕士生导师、计算机基础教学部副主任。主要研究方向包括网络及数据库系统集成、数据挖掘、数字图像处理等。主讲数字信号处理、Java 程序设计与应用、VB 程序设计、计算机组成与网络等课程。作为项目主要参加者,承担和完成了多项国家级和上海市级的科研项目。曾获国家纺织工业局三等奖。目前兼任上海市高等学校计算机等级考试命题组成员。



张欢欢 1990年毕业于中国人民解放军国防科技大学三系,同年到上海华东理工大学计算中心任教。1994年开始在华东理工大学计算机系攻读在职研究生并于1997年毕业,获得硕士学位。2002年开始攻读在职博士。主要研究方向是形式化方法与验证技术、通用程序设计。在教学上主要从事程序设计语言的教学。



王瑾德 1983年毕业于上海复旦大学数学系。现任上海中医药大学现代教育技术中心、计算机教学部教师。曾经开设高等数学、数理统计学、模糊数学等课程。1992年转向计算机应用教学,曾经开设的课程有:计算机文化基础(初级、中级)、Pascal 程序设计、C 语言程序设计、Visual Basic 程序设计、Java 程序设计以及 Web 设计与应用等。曾经参与《现代医学信息学》、《高校计算机等级考试实用指南》等教材与辅导书的编写工作。



马可幸 1987年毕业于同济大学。现为上海东华大学计算机科学与技术学院讲师,从事计算机基础教学,讲授程序设计和多媒体技术等课程。感兴趣的研究方向是面向对象程序设计技术和网络多媒体教学。曾参加教育部多媒体教学课件项目的研制工作。

目 录

前言	
作者介绍	
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 SDK 目录结构	5
1.3.4 Java 2 开发工具简介	5
1.3.5 Java 的 API 类库与 API 文档	6
1.3.6 Java 程序结构	7
1.3.7 最简单的 Java 程序	9
1.4 Java 程序的开发过程	10
1.4.1 编辑源程序	10
1.4.2 编译源程序	10
1.4.3 运行 Java 程序	11
1.4.4 编写第一个 Applet	11
1.5 Java 资源的获取与安装	13
1.5.1 下载 SDK	13
1.5.2 安装 SDK	13
1.5.3 设置 SDK 的操作环境	15
1.6 Eclipse 简介	17
1.6.1 Eclipse 环境界面	17
1.6.2 Java 开发环境(JDE)	19
1.7 本章概要	24
1.8 课后练习	25
第 2 章 Java 程序设计的基本概念	27
2.1 数据类型与标识符	27
2.1.1 数据类型	27
2.1.2 标识符	28
2.2 常量	28
2.2.1 整型常量值	29
2.2.2 实型常量值	29
2.2.3 字符型和字符串常量值	29
2.2.4 布尔型常量值	29
2.2.5 常量的定义和使用	30
2.3 变量	31
2.3.1 变量概述	31
2.3.2 变量的定义及赋值	32
2.3.3 变量的作用域	33
2.3.4 变量的默认值	33
2.4 运算符与表达式	34
2.4.1 赋值运算符与赋值表达式	34
2.4.2 算术运算符与算术表达式	34
2.4.3 关系运算符与关系表达式	36
2.4.4 逻辑运算符与逻辑表达式	37
2.4.5 复合赋值运算符	38
2.4.6 其他运算符	38
2.4.7 运算符的优先级与结合性	39
2.5 数据类型的转换	40
2.6 本章概要	42
2.7 课后练习	42
第 3 章 Java 的结构化程序设计	45
3.1 顺序结构及基本语句	45
3.1.1 赋值语句	45
3.1.2 输入语句	46
3.2 选择结构语句	48
3.2.1 if 语句	48
3.2.2 if-else 语句	50
3.2.3 if-else if 语句	51
3.2.4 if 语句的嵌套	53
3.2.5 switch 语句	53
3.3 循环结构语句	56
3.3.1 for 循环语句	56
3.3.2 while 循环结构语句	58
3.3.3 do while 循环结构语句	58
3.3.4 循环结构语句的嵌套	59
3.4 转移语句	61
3.4.1 break 语句	61
3.4.2 continue 语句	63

3.5 数组	64	4.6.2 方法的覆盖	111
3.5.1 一维数组的声明与引用	64	4.6.3 前期绑定和后期绑定	112
3.5.2 数组的赋值	65	4.7 接口	113
3.5.3 一维数组程序举例	66	4.7.1 接口的声明	113
3.5.4 二维数组的声明及引用	68	4.7.2 接口的实现	114
3.5.5 数组的复制	70	4.8 包	115
3.5.6 字符串处理	71	4.8.1 Java 的类和包	115
3.6 方法	75	4.8.2 引用 Java 定义的包	117
3.6.1 Java 的程序模块化	75	4.8.3 自定义包	117
3.6.2 方法的定义及调用	75	4.8.4 包和访问权限	119
3.6.3 参数的传递	76	4.9 本章概要	120
3.6.4 作用域	79	4.10 课后练习	120
3.6.5 return 语句	79	第 5 章 Java 的图形用户界面	125
3.6.6 方法的嵌套调用	80	5.1 Applet 概述	125
3.6.7 递归	82	5.1.1 一个简单的 Applet 例子	125
3.7 本章概要	83	5.1.2 Applet 的安全模型	128
3.8 课后练习	83	5.1.3 java.applet.Applet 类与其他类的关系	128
第 4 章 Java 的面向对象程序设计	89	5.1.4 Applet 的生命期	128
4.1 面向对象程序设计概述	89	5.2 java.awt 与图形用户界面	130
4.1.1 面向对象程序设计的目的	89	5.2.1 标签和文本域	131
4.1.2 类和对象	89	5.2.2 Java 中的事件处理机制	133
4.1.3 面向对象设计的核心技术	89	5.2.3 按钮	138
4.1.4 Java 的面向对象技术	90	5.2.4 布局	139
4.2 类的创建	90	5.2.5 面板	141
4.2.1 类的声明格式	90	5.2.6 文本区域	142
4.2.2 成员变量	91	5.2.7 复选框和单选按钮	143
4.2.3 成员方法	91	5.2.8 下拉列表	146
4.3 对象的创建和使用	93	5.2.9 列表	148
4.3.1 创建对象	93	5.2.10 窗口与菜单	151
4.3.2 构造方法和对象的初始化	94	5.2.11 对话框	163
4.3.3 对象的使用	96	5.3 swing	166
4.3.4 对象的销毁	96	5.3.1 swing 的优点	166
4.4 类的封装	98	5.3.2 swing 类的继承关系	166
4.4.1 封装的目的	98	5.3.3 swing 中的容器	167
4.4.2 访问权限的设置	98	5.3.4 swing 中的常用组件	180
4.4.3 类成员(静态成员)	100	5.3.5 综述 swing 中的事件	196
4.5 类的继承	103	5.4 二维图形设计	197
4.5.1 继承的基本概念	103	5.4.1 二维图形的坐标系统	198
4.5.2 子类的创建	104	5.4.2 字体	198
4.5.3 null、this、super 对象运算符	107	5.4.3 颜色	198
4.5.4 最终类和抽象类	109	5.4.4 绘图	199
4.6 类的多态性	110	5.5 本章概要	203
4.6.1 方法的重载	110		

5.6 课后练习	203	Stream 类	243
第 6 章 Java 的异常处理	207	8.2.2 文件字节输入输出流	244
6.1 异常和异常对象	207	8.2.3 创建文件输入输出流对象 并打开文件	244
6.2 异常的捕获与处理	208	8.2.4 对文件进行读写的方法	245
6.3 try 语句的嵌套	211	8.2.5 字节文件流的关闭	245
6.4 throw 语句	213	8.3 字符流	247
6.5 throws 语句	214	8.3.1 Reader 和 Writer 类	247
6.6 使用异常处理的准则	215	8.3.2 字符文件流 FileReader 和 FileWriter 类	248
6.7 本章概要	215	8.3.3 创建字符流文件对象并 打开文件	248
6.8 课后练习	216	8.3.4 字符文件流的读写	248
第 7 章 Java 的多线程程序设计	219	8.3.5 字符文件流的关闭	248
7.1 线程的概念	219	8.3.6 字符缓冲流 BufferedReader 和 BufferedWriter	249
7.1.1 进程和线程	219	8.4 文件类	250
7.1.2 线程和多任务	219	8.4.1 文件类 File	250
7.1.3 Java 对多线程的支持	219	8.4.2 File 类的方法	251
7.2 线程的创建	219	8.4.3 文件过滤	252
7.2.1 Runnable 接口	220	8.5 文件的随机读写	254
7.2.2 Thread 类	220	8.5.1 RandomAccessFile 类	254
7.2.3 创建线程的方法	220	8.5.2 RandomAccessFile 的构造方法	254
7.3 线程的状态与控制	222	8.5.3 RandomAccessFile 的方法	254
7.3.1 线程的状态	222	8.6 本章概要	256
7.3.2 对线程状态的控制	224	8.7 课后练习	256
7.4 线程的优先级和调度	227	第 9 章 Java 的网络应用	259
7.4.1 线程的优先级	227	9.1 网络的基本概念	259
7.4.2 线程的调度	227	9.1.1 IP 地址和端口号	259
7.5 线程组	227	9.1.2 URL	260
7.5.1 线程组	227	9.1.3 TCP 与 UDP	260
7.5.2 ThreadGroup 类	228	9.1.4 Socket	261
7.6 线程的同步	228	9.2 URL 的使用	262
7.6.1 线程的同步机制	228	9.2.1 使用 URL 的方法	262
7.6.2 共享数据的互斥锁定	230	9.2.2 应用举例	264
7.6.3 数据传送时的同步控制	233	9.3 Socket 的应用	270
7.6.4 死锁	235	9.3.1 TCP 套接字通信基本步骤	271
7.7 本章概要	238	9.3.2 服务器端程序设计举例	272
7.8 课后练习	238	9.3.3 客户端程序设计举例	275
第 8 章 Java 的输入输出流	241	9.4 网络安全管理	278
8.1 流的基本概念	241	9.4.1 Java 的安全特性	279
8.1.1 数据流	241	9.4.2 缓存溢出	279
8.1.2 Java 的标准输入输出	241	9.4.3 竞争状态	280
8.1.3 java.io 包中的数据流	243		
8.2 字节流和顺序文件的读写	243		
8.2.1 InputStream 和 Output-			

9.4.4 建立安全性策略	281	10.5 课后练习	299
9.4.5 安全基本原则	283	附录 1 部分参考答案	301
9.5 本章概要	283	附录 2 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	307
9.6 课后练习	284	附录 3 2005 年上海市高等学校计算机 等级考试试卷(二级) ——《Java 程序设计》	311
第 10 章 Java 的多媒体应用	287		
10.1 图像显示	287		
10.2 动画设计	289		
10.3 声音播放	295		
10.4 本章概要	299		

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(称为程序设计语言)与计算机硬件的发展一样,也经历了几个发展阶段。

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即可以在各类机器上运行,于是各种高级语言便应运而生。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,用如上所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,因此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设计语言编写的程序(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, Windows9.x 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纪元。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的一部分, 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, 没想到这一举动竟使 Java 在 Internet 上获得了空前的成功, 使它成为了 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 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(Java Development Kit, JDK)发布
1999 年 7 月	Java 升级至 1.2 版
2000 年 9 月	Java 升级至 1.3 版
2001 年 7 月	Java 升级至 1.4 版

由上表可知, Java 的最新版本为 1.4 版。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, 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) J2ME——即 Java 2 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) J2SE——即 Java 2 (Software Development Kit) Standard Edition, 是 Java 最通行的版

本, 是用于工作站、PC 机的 Java 标准平台, 因此也是本书应用程序的使用版本。

3) J2EE——即 Java 2 Enterprise Edition(可扩展的企业应用 Java 2 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 语言主要有以下特点:

1. 易于学习

Java 语言很简单。这里所说的简单, 主要是针对熟悉类似 C++ 语言的程序设计人员来说的, 因为它的语法和 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有利于有程序设计语言基础的 Java 语言初学者的学习。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”的最大特点便是可以跨平台运行, 即程序设计人员们常说的“编写一次, 到处运行”, 正是这一特性成为 Java 得以迅速普及的重要原因。

4. 分布式

所谓的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, 便于开发网上应用系统。

5. 安全性

Java 2 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. 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。

7. 丰富的 API(Application Program Interface)文档和类库

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,这也正是 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。

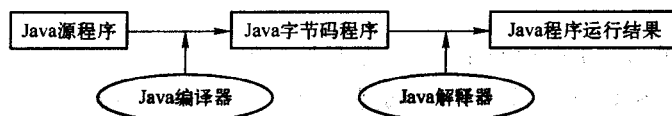


图 1-1 Java 程序的开发及运行周期

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作 JVM。当然我们可以把 Java 的字节码(byte-codes)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,这些内容都由一系列程序文件(软件)组成。这些文件的集合称为 Java 的开发工具。从 Java 的特点部分可以看到:任何支持用于编写 Java 程序的环境都会给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”,就相当有意思了,随着大家学习的进一步深入,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)。

Java 所有开发工具中,我们要特别介绍开发工具包(Software Development Kit)Standard Edition 1.4.2,简称 SDK1.4,有些参考书称此开发工具包为 JDK1.4。本书中的所有程序都是用这个版本的工具测试的。它是由 Sun Microsystems 公司开发的一套 Java 程序开发软件,可从 Sun Microsystems 公司的网站免费获得,在本章的第 5 节我们将向大家详细介绍 SDK1.4 的下载与安装。

若要安装 Java 开发环境,必须在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 以及 Windows 等,Java 开发工具包可从 <http://java.sun.com/j2se/1.4/download.html> 下载。由于在大家学习的过程中,所接触的大多是 Windows 操作系统,因此我们现在就以 SDK1.4 Windows 版本为例,下载程序“j2sdk-1_4_2_03-windows-i586-p.exe”,解压安装后会出现如图 1-2 所示的目录结构。

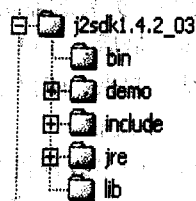


图 1-2 SDK 目录结构

现在来介绍一下 SDK1.4 目录以及所包含的文件。

1.3.3 SDK 目录结构

1. 软件安装的根目录

包含版权、许可以及 README 文件。还包括: src.zip 构成 Java 平台核心 API 的所有类的源文件。

2. bin 目录

包含 SDK 开发工具的可执行文件。

3. lib 目录

开发工具使用的归档包文件。其中 tools.jar 包含支持 SDK 的工具和实用程序的非核心类, dt.jar 是 BeanInfo 文件的 DesignTime 归档, BeanInfo 文件用来告诉集成开发环境 (IDE) 如何显示 Java 组件以及如何让开发人员根据应用程序自定义它们。

4. jre 目录

Java 运行时环境的根目录。其子目录 \jre\bin 中包含 Java 平台使用的工具和库的可执行文件及 DLL。其中 DLL 文件是指经典虚拟机使用的 DLL 文件。经典虚拟机是 Java 虚拟机的语言注释版本。当新虚拟机可用时, 它们的 DLL 文件将被安装在 jre/bin 的某个新子目录中。

子目录 jre\lib 是 Java 运行时环境使用的代码库、属性设置和资源文件。包括 rt.jar 为自举类(构成 Java 平台核心 API 的 RunTime 类)。charsets.jar 为字符转换类及其他与国际化和本地化有关的类。

子目录 jre\lib\ext 是 Java 平台扩展的默认安装目录。

子目录 jre\lib\security 包含用于安全管理文件。这些文件包括安全策略(java.policy)和安全属性(java.security)文件。

5. demo 目录

含有资源代码的程序示例。

6. include 目录

包含 C 语言头文件, 支持 Java 本地接口和 Java 虚拟机调试程序接口的本地代码编程技术。

1.3.4 Java 2 开发工具简介

在 SDK 的 bin 目录下, 存放着 Java 2 提供的一些可执行应用程序, 为我们开发和测试自己的程序提供了工具:

1) javac: Java 语言的编译器, 将 Java 源代码转换成字节码。

2) java: Java 解释器, 直接从类(字节码)文件执行 Java 应用程序代码, 类名后面可带一个或多个字符串参数。

3) javadoc: API 生成器, 根据 Java 源代码及其说明语句生成 HTML 格式的 API 文档。

4) appletviewer: 小应用程序浏览器。

5) jar: Java Archive 文件归档工具, 用它可以把类文件以及其他格式的文件进行压缩, 以便于程序中引入以及网络传输等。

6) jdb: Java 调试工具 Debugger。

- 7) javap: 类文件的分解器。
- 8) extcheck: 判断和检测 jar 冲突。
- 9) rmic: 为远程对象调用产生存根和框架。
- 10) rmiregistry: 提供远程对象注册服务。
- 11) rmid: 激活远程方法的系统后台服务。
- 12) serialver: 返回类的序列化版本号。
- 13) native2ascii: 将本地编码的文本文件转化成 Unicode 编码文件。
- 14) keytool: 按键和键盘识别的管理工具。
- 15) jarsigner: 产生和识别 jar 文件签名。
- 16) policytool: 图形化用户界面工具的管理策略文件。
- 17) tnameserv: 提供可访问的 CORBA 命名服务。
- 18) idlj: 生成 .java 文件使它映射到一个 OMG RDL 接口并且使一个应用可以利用 CORBA 功能写入 Java 程序。

1.3.5 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的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、可以直接供你在你的程序中调用的小程序,例如标准函数等等,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,SDK 给出了一套标准的类库,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。如图 1-3 所示。

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。因而一般不必显示导入 java.lang 中的任何类。这个包中的许多类在本书的许多其他章节还将叙述。

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——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。

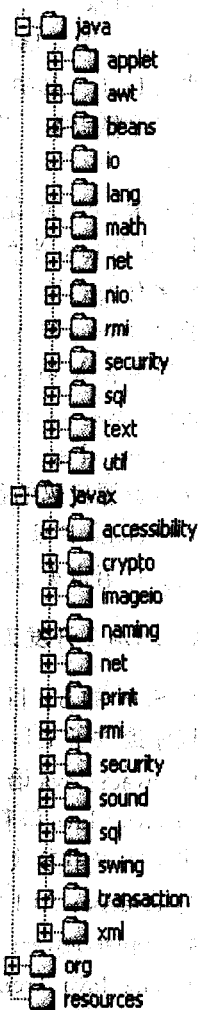


图 1-3 Java 类库结构

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,访问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,或者缺省时通过 ODBC(开放数据库连接)标准的直接映射实现。

Java 的 API 文档是 SDK 说明文件的一部分,是具体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。当你从 Sun Microsystems 公司的网站 <http://java.sun.com/j2se/1.4.2/download.html> 下载 SDK 参考文件“j2sdk-1_4_2-doc.zip”,解压缩后,可以在名为“j2sdk-1_4_2-doc”的文件夹下看到如图 1-3 所示的目录结构,其中的文本是 HTML 格式文件,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详细描述、构造方法的详细说明及进一步的描述等等。为了方便使用,我们通常将 C:\j2sdk-1_4_2-doc\docs\api\index.html 放置到浏览器的收藏夹中,以供随时查阅。它在浏览器下的界面形式如图 1-4 所示。

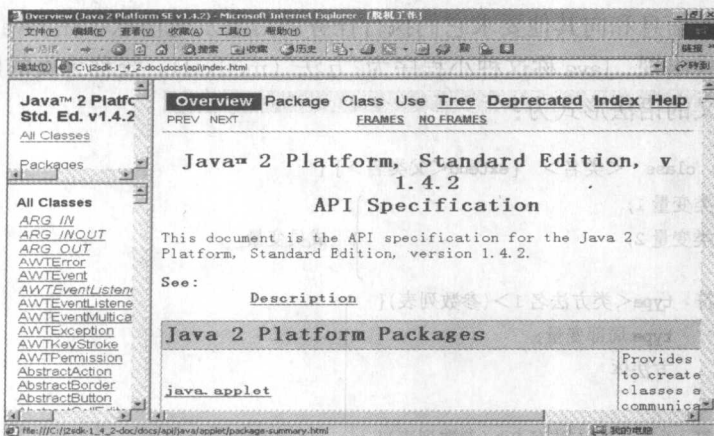


图 1-4 API 文档索引页

1.3.6 Java 程序结构

1. 源程序文件的构成

要编写 Java 程序,首先应该知道 Java 程序文件中必须包括什么内容,Java 程序的源程序文件结构如下:

- package 语句, 0~1 句, 必须放在文件开始,作用是把当前文件放入所指向的包中。