

JAVA

技术实用教程

AVA

许晓宁 主编

JISHUSHIYONGJIAOCHENG



东南大学出版社
SOUTHEAST UNIVERSITY PRESS

全国职业教育计算机类系列教材

Java 技术实用教程

主 编 许晓宁

参 编 (按姓氏笔画排序)

朱 沙 邢素萍

戎 戟 周学全

东南大学出版社

内 容 提 要

Java 语言是当今最流行的计算机高级编程语言之一,围绕网络编程的设计定位和代表新技术的面向对象特性使它在当今网络时代的计算机应用中占有非常重要的地位。

本书介绍了 Java 语言的特点、基本语法,面向对象的程序设计方法,Java 技术的应用。第一部分重点介绍了 Java 语言的运算符、流程控制、数组、异常、I/O 及文件操作、多线程机制、图形化界面设计、Applet 小程序等;第二部分介绍了多线程编程、网络编程、多媒体编程和数据库编程等 Java 编程技术。

本书是一本介绍 Java 技术的实用教程,配备了大量实例,由浅入深,通俗易懂,突出 Java 技术的应用。本书可作为高职、高专院校计算机专业、通信专业、电子商务专业及相关专业的教材,也可作为 Java 技术开发人员和广大用户的自学或参考用书。

图书在版编目(CIP)数据

Java 技术实用教程/许晓宁主编. —南京:东南
大学出版社,2005. 1

ISBN 7-81089-820-5

I. J... II. 许... III. Java 语言-程序设计
IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 131133 号

东南大学出版社出版发行
(南京四牌楼 2 号 邮编 210096)

出版人:宋增民

江苏省新华书店经销 扬中市印刷有限公司印刷
开本:787mm×1092mm 1/16 印张:18.5 字数:468 千字
2005 年 1 月第 1 版 2005 年 1 月第 1 次印刷
印数:1—4000 册 定价:28.00 元

(凡有印装质量问题,可直接向发行部调换。电话:025-83795801)

出版说明

全国职业教育计算机专业建设研讨会于2004年7月18日在湖北三峡职业技术学院召开,来自上海、江苏、山西、辽宁、贵州、黑龙江等40多位职业技术学院的代表参加了会议。

在本次会议上,与会专家学者对目前职业教育的现状进行了深刻地分析,特别对计算机专业建设提出了独到的见解。他们一致认为:计算机专业建设要与教学改革相结合,以市场需求为导向,以教材建设为基础。因此,会议决定为配合计算机专业建设,编写一套适合职业教育的计算机系列教材,要求突出职业特点,有创新思想,以“考证”为切入点,加强实践环节。

根据各校计算机专业建设和课程设置情况,本次会议由全国职业教育计算机类教材建设委员会秘书长孔繁华组织各院校计算机专业教师确定了首批教材建设的选题,以后还将随着专业建设的深入及计算机技术的发展,逐步形成一套完善的、切合实际的计算机职业教育系列教材。

全国职业教育电子信息类教材编委会总要求:坚决贯彻职业教育的要求,即基础适度够用、加强实践环节、突出职能教育,把握职业教育电子信息类专业课程建设的特点;立足当前学生现状,面向用人单位(市场),打破条条框框,少一些理论,多一些技能教育;采取逆向思维的方式编写,即从市场需要什么技能来决定学生需要什么知识结构,并由此决定编写什么教材。

全国职业教育电子信息类教材编委会会员单位:

南京信息职业技术学院

本溪电子工业学校

扬州电子信息学校

河南信息工程学校

大连电子工业学校

黑龙江信息技术职业学院

本溪财贸学校

山西工程职业技术学院

四川省电子工业学校

锦州铁路运输学校

内蒙古电子信息职业技术学院

江苏海事职业技术学院

黑龙江农业经济职业技术学院

南通纺织职业技术学院

湖北三峡职业技术学院

长沙市电子工业学校

山西综合职业技术学院

北京信息职业技术学院

福建省电子工业学校

山西省邮电学校

新疆机械电子职业技术学院

山东信息职业技术学院

哈尔滨机电工程学校

上海机电工业学校

贵州省电子工业学校

南京交通职业技术学院

扬州职业大学

南通航运职业技术学院

全国职业教育电子信息类教材编委会

2005年1月

前 言

Java 是 Sun 公司推出的新一代面向对象程序设计语言,特别适用于 Internet 应用程序开发,它的跨平台可移植性优势使得 Java 语言正如火如荼,风靡计算机世界。Java 语言采用面向对象程序设计技术,功能强大且简单易学,特别适用于 Internet 程序设计,目前已经成为应用广泛的 Web 程序设计语言。

Java 既是一种程序设计语言,又是一种技术体系。本书是一本内容丰富、实践性强的程序设计教材。本书在编写上力图通过介绍 Java 语言的基本知识,引导学生进入 Java 程序设计世界,通过众多实例认识 Java 强大的设计能力,逐步掌握 Java 基本应用技术,为今后进一步学习 Java 其他高级技术奠定基础。

为实现对高等职业技术人才的培养,重点掌握本专业领域实际工作的基本知识和职业技能,本书编写以知识和技能为基点,围绕各基点完成课堂教学、课后自学、思考与练习、上机操作等教学环节,将课堂教学与实践教学紧密地联系起来,最终达到使学生能够比较熟练地编写中档规模的 Java 应用程序的目标。

本书在知识体系的构造上,以简明实用、便于学习、注重知识的起步和连贯、反映 Java 技术的最新发展和应用为特色;以 Java 语言的应用程序和小程序设计为主线,采用通俗易懂的方法讲述大量实例;注重实践练习,每章都安排了各种类型的思考和练习题;突出实训过程,每章都有上机操作要求,培养学生实际动手开发程序的能力。

本书内容包括两个部分,第一部分 Java 语言应用基础,包括 1 至 8 章。1 至 3 章介绍 Java 语言的基本特性及基本语法,包括 Java 语言概述、数据类型、运算符与表达式以及流程控制语句;4 至 6 章介绍 Java 语言基本的面向对象程序设计技术,包括 Java 类与对象、数组与字符串操作、异常处理;7、8 两章介绍 Java 输入输出及文件操作、Java 图形用户界面设计技术、Applet 小程序设计和应用。第二部分 Java 语言高级应用,包括 9 至 12 章。第 9 章介绍 Java 多线程编程;第 10 章介绍 Java 网络编程的基本概念和网络编程技术;第 11 章介绍多媒体技术在 Java 程序中的实现;第 12 章介绍 Java 数据库程序设计、JDBC 体系结构及工作机制,以实例说明怎样应用 Java 类与接口进行 Java 数据库程序设计。

本书授课的参考学时数为 72。其中,理论教学的参考学时数为 42,上机实验的参考学时数为 30。课程结束之后还可以安排 1~2 周的课程设计或实训。

全书由许晓宁主编,第 1 章、第 4 章、第 11 章和第 12 章由许晓宁编写,第 2 章和第 7 章由朱沙编写,第 3 章和第 8 章由邢素萍编写,第 5 章和第 9 章由戎戟编写,第 6 章和第 10 章由周学全编写。

由于作者水平有限,编写时间仓促,错误之处在所难免,敬请广大读者指正。欢迎提出宝贵意见(E-mail: yzddxxn@163.com)。

编 者

2004 年 10 月

目 录

1 Java 语言概述	(1)
1.1 Java 语言简介	(1)
1.1.1 Java 语言的发展历史	(1)
1.1.2 Java 语言与 Internet	(2)
1.1.3 Java 语言的特点	(3)
1.2 Java 程序运行环境	(4)
1.2.1 安装 J2SDK 开发环境	(4)
1.2.2 集成开发环境 JBuilder	(5)
1.2.3 JPad Pro 开发工具	(5)
1.3 编译和运行 Java 程序	(9)
1.3.1 J2SDK 环境下开发 Java Application 程序	(9)
1.3.2 J2SDK 环境下开发 Java Applet 程序	(9)
1.3.3 使用 JPad Pro 开发 Java 应用程序	(11)
本章小结	(14)
思考与练习	(14)
上机操作	(15)
实验 1 运行环境的安装和配置	(15)
实验 2 Java 程序的编辑、编译和运行	(17)
2 Java 语言基础	(19)
2.1 数据类型概述	(19)
2.1.1 整型数据	(20)
2.1.2 浮点型数据	(20)
2.1.3 布尔型数据	(20)
2.1.4 字符型数据	(20)
2.1.5 各类型数据之间的转换	(20)
2.2 变量和常量	(21)
2.2.1 标识符	(21)
2.2.2 常量	(22)
2.2.3 变量	(24)
2.3 运算符与表达式	(25)
2.3.1 算术运算符和算术表达式	(25)
2.3.2 关系运算符和关系表达式	(26)
2.3.3 布尔运算符和布尔表达式	(27)
2.3.4 位运算符和位运算表达式	(29)
2.3.5 赋值运算符和赋值表达式	(29)

2.3.6 其他运算符	(30)
2.3.7 运算符的优先级	(30)
本章小结	(32)
思考与练习	(32)
上机操作	(33)
实验 数据类型及表达式	(33)
3 流程控制语句	(36)
3.1 条件控制语句	(36)
3.1.1 if 语句	(36)
3.1.2 switch 语句	(40)
3.2 循环控制语句	(41)
3.2.1 while 语句	(41)
3.2.2 do-while 语句	(42)
3.2.3 for 语句	(44)
3.2.4 循环语句的嵌套	(45)
3.3 转移控制语句	(46)
3.3.1 break 语句	(47)
3.3.2 continue 语句	(48)
3.3.3 return 语句	(49)
本章小结	(50)
思考与练习	(50)
上机操作	(53)
实验 流程控制语句的应用	(53)
4 Java 类与对象	(56)
4.1 面向对象编程的基本概念	(56)
4.1.1 对象	(56)
4.1.2 类	(56)
4.1.3 封装性	(57)
4.1.4 继承性	(57)
4.1.5 多态性	(57)
4.2 类的定义和使用	(57)
4.2.1 类定义	(57)
4.2.2 方法	(59)
4.2.3 对象的创建与使用	(62)
4.2.4 构造函数	(64)
4.3 类的继承	(66)
4.3.1 子类的创建	(66)
4.3.2 this 与 super 关键字	(69)
4.4 多态	(71)
4.4.1 方法的重载	(71)

4.4.2	方法的覆盖	(72)
4.5	接口	(73)
4.5.1	接口的定义	(73)
4.5.2	接口的实现	(74)
4.6	包	(76)
4.6.1	Java 预定义包	(76)
4.6.2	包的定义	(80)
4.6.3	包的引用	(81)
本章小结		(81)
思考与练习		(82)
上机操作		(83)
实验 1 类的定义、对象的创建及使用		(83)
实验 2 方法和构造函数		(85)
实验 3 类的继承、方法的覆盖与重载		(87)
5	数组与字符串操作	(90)
5.1	数组	(90)
5.1.1	一维数组	(90)
5.1.2	多维数组	(92)
5.2	字符串操作	(94)
5.2.1	字符串的表示和生成方法	(95)
5.2.2	字符串的访问	(95)
5.2.3	字符串的修改	(96)
5.2.4	字符串的比较	(97)
5.2.5	可变字符串的处理	(97)
本章小结		(100)
思考与练习		(101)
上机操作		(103)
实验 1 数组操作		(103)
实验 2 字符串操作		(104)
6	异常处理	(105)
6.1	异常与异常类	(105)
6.1.1	异常处理机制	(105)
6.1.2	异常类的继承结构	(107)
6.2	异常的处理	(108)
6.2.1	捕获异常	(109)
6.2.2	多异常的处理	(112)
6.2.3	异常的抛出	(113)
本章小结		(116)
思考与练习		(116)
上机操作		(118)

实验 异常处理语句的使用	(118)
7 Java 输入输出及文件操作	(120)
7.1 文件	(120)
7.1.1 创建文件对象	(120)
7.1.2 文件类提供的方法	(121)
7.2 Java 输入输出流	(123)
7.2.1 字节输入流	(123)
7.2.2 字节输出流	(123)
7.2.3 字符类输入流	(124)
7.2.4 字符类输出流	(124)
7.3 过滤流	(125)
7.4 管道流	(126)
7.5 对象的串行化	(127)
本章小结	(129)
思考与练习	(129)
上机操作	(130)
实验 文件创建、读写操作	(130)
8 图形用户界面和 Applet	(133)
8.1 Java 的 AWT 图形界面	(133)
8.1.1 AWT 简介	(133)
8.1.2 AWT 图形界面组件	(134)
8.2 事件机制	(144)
8.2.1 委派事件模型	(144)
8.2.2 事件处理类概述	(147)
8.2.3 应用 ActionEvent 类	(149)
8.2.4 应用 ItemEvent 类	(151)
8.2.5 应用 TextEvent 类	(152)
8.2.6 应用 KeyEvent 类	(154)
8.2.7 应用 MouseEvent 类	(155)
8.2.8 应用 WindowEvent 类	(158)
8.3 AWT 布局管理器	(159)
8.3.1 应用 BorderLayout 类处理边界页面设置	(160)
8.3.2 应用 CardLayout 类进行多层页面设置	(162)
8.3.3 应用 FlowLayout 类进行流动式页面设置	(163)
8.3.4 应用 GridLayout 类进行方格式页面设置	(165)
8.4 应用 Swing 创建用户界面	(166)
8.4.1 Swing 基本组件	(166)
8.4.2 Swing 的页面布局	(172)
8.4.3 Swing 的组件	(175)
8.5 Java Applet	(177)

8.5.1	Applet 简介	(177)
8.5.2	Applet 生命周期及主要方法	(178)
8.5.3	Applet 的 HTML 标记及属性	(179)
8.5.4	Applet 的安全机制	(181)
	本章小结	(181)
	思考与练习	(182)
	上机操作	(183)
实验 1	组件的创建及安放	(183)
实验 2	布局管理器的应用	(185)
实验 3	Applet 小程序	(186)
9	多线程编程	(188)
9.1	线程的概念	(188)
9.2	线程的使用	(189)
9.2.1	线程基本结构	(189)
9.2.2	线程的状态及调度	(194)
9.2.3	线程的控制	(196)
9.3	线程的管理	(200)
9.3.1	同步	(200)
9.3.2	优先级	(200)
	本章小结	(201)
	思考与练习	(202)
	上机操作	(203)
实验	多线程编程	(203)
10	Java 网络编程	(208)
10.1	网络基础	(208)
10.1.1	服务器和客户机	(209)
10.1.2	使用 URL	(209)
10.1.3	套接字概述	(210)
10.1.4	网络类和接口	(210)
10.2	网络编程	(211)
10.2.1	解析网站 URL	(211)
10.2.2	使用 TCP 协议进行服务器/客户端编程	(215)
	本章小结	(224)
	上机操作	(224)
实验 1	利用 URL 访问网站	(224)
实验 2	利用 URL 读取服务器上的文本文件	(225)
实验 3	利用 Socket 与服务器通信	(227)
11	Java 多媒体编程	(230)
11.1	图形绘制	(230)
11.1.1	颜色的使用	(230)

11.1.2	绘制直线	(231)
11.1.3	绘制矩形	(232)
11.1.4	绘制椭圆和圆	(233)
11.1.5	绘制圆弧	(234)
11.1.6	绘制多边形	(235)
11.1.7	Graphics2D 绘图的实例	(236)
11.1.8	Java Application 绘图	(237)
11.2	图像处理	(239)
11.2.1	载入图像	(239)
11.2.2	显示图像	(240)
11.3	动画处理	(243)
11.3.1	动画程序设计	(243)
11.3.2	消除闪烁	(245)
11.4	声音播放	(246)
	本章小结	(249)
	上机操作	(249)
	实验 Java 多媒体应用	(249)
12	Java 数据库编程	(255)
12.1	JDBC 概述	(255)
12.1.1	JDBC API 结构	(255)
12.1.2	JDBC 驱动程序类型	(256)
12.1.3	JDBC-ODBC 桥	(256)
12.1.4	使用 JDBC-ODBC 桥接器与 Access 数据库相连	(256)
12.2	使用 JDBC 对数据库进行操作	(260)
12.2.1	对数据库进行查询	(260)
12.2.2	修改数据库中的记录	(269)
12.2.3	向数据库中插入记录	(271)
12.2.4	删除数据库中的记录	(273)
12.3	使用 JDBC 进行数据库定义性操作	(273)
12.3.1	创建数据表	(273)
12.3.2	删除数据表	(275)
12.3.3	创建视图	(276)
12.3.4	删除视图	(278)
	本章小结	(279)
	上机操作	(279)
	实验 1 创建数据源	(279)
	实验 2 数据库操作	(280)
	参考文献	(286)

1 Java 语言概述

【本章学习目标】

本章主要讲述 Java 语言产生的历史及其基本特性,Java 语言作为一门面向对象编程语言的基本编程思路,Java 程序的两种方式:Java Application 和 Java Applet,安装和配置 Java 开发环境的方法。通过本章的学习应掌握以下主要内容:

- (1) Java 语言的发展历史;
- (2) Java 语言的基本特性;
- (3) 安装和配置 J2SDK 开发环境;
- (4) 安装 JPad Pro 开发工具;
- (5) 编译和运行 Java 程序。

1.1 Java 语言简介

Java 语言是 Sun 公司开发的一种编程语言。它具有平台独立性、安全性、面向对象、动态性、分布性等卓越的特性,具备强大的网络功能。更广义地说,Java 是一个普遍适用的软件平台,可用于 Internet(互联网)、Intranet(企业内部网)及各种设备、系统、计算平台。它是一种基本的、结构紧凑的先进技术,一经产生就引起了广泛的关注,并在很短的时间内蓬勃发展起来。

1.1.1 Java 语言的发展历史

Java 来自于 Sun 公司 1991 年的一个叫 Green 的项目,这个项目需设计一种小型的计算机语言,用于机顶盒、家电控制芯片等消费类设备。由于这些设备功能有限、内存较小,所以这种语言必须尽可能地小,并能生成非常紧凑的代码。而且最重要的一点是,这些设备所采用的处理芯片多种多样,要使这个语言有强大的生命力,就绝不能受限于任何一种体系结构。

开始准备采用 C++,但 C++太复杂,安全性差。1992 年基于 C++开发了一种新的语言 Oak(Java 的前身),Oak 是一种用于网络的精巧而安全的语言,后因语言本身和市场的问题,使得消费性电子产品的发展无法达到当初预期的目标。

随着 Internet 上的 WWW(Word Wide Web,万维网)服务越来越普遍,人们需要一个好用的浏览器,以便更好地阅读 Web 页面。Oak 项目组成员用 Java 编制了 HotJava 浏览器,得到了 Sun 公司首席执行官 Scott McNealy 的支持,触发了 Java 进军 Internet。HotJava 这个试验品在 1995 年 5 月召开的 Sun World 大会上首次亮相,从而引发了遍及全球、至今未衰的 Java 热。

1995 年秋,Netscape 决定让其浏览器支持 Java,并在 1996 年初发布了支持 Java 的版本。这意味着 Java 语言开始大流行。后来,IBM、Symantec、Inprise 等许多著名 IT 公司都注册了 Java 使用许可证。微软公司也在 IE 中提供了对 Java 的支持功能,并在 Windows 中提供了 Java 虚拟机。

Java 语言的初露锋芒,让 Sun 公司意识到它的价值,于是在 1996 年初发布了 Java 的第一个版本。但这个版本的 Java 语言并不是十分完备,不适合用于正规的应用程序的开发。

1996 年 5 月,在美国旧金山召开了 JavaOne 大会,在这次大会上,Sun 公司为人们绘制了一个美好的蓝图。之后 Sun 公司就发布了令人满意的 Java 1.1 版,它实现了绝大部分的目标。

在 1998 年召开的 JavaOne 大会上,Sun 公司发布了 Java 1.2 版,这就是 Java 2,它以功能全面、具有高度扩展能力的新版本换掉了原来的 GUI 界面和图形工具包。朝着“一次编写,到处运行”的宏伟目标又前进了一大步。目前,Java 语言已经成为最流行的网络编程语言,许多院校纷纷开设 Java 课程,Java 正逐步成为世界上程序员使用最多的编程语言。

1.1.2 Java 语言与 Internet

Java 是一种面向对象的程序设计语言,除了可以解决传统的程序设计问题以外,还能解决 WWW 上的编程问题。

Java 通过“小应用程序”(Applet)巧妙地解决了客户端编程的问题。通过小应用程序,一旦用户需要客户软件,软件就可从服务器被自动下载回来。它们能自动取得客户软件的最新版本,不会出错,也没有重新安装的麻烦。由于 Java 的设计原理,程序员只需要创建程序的一个版本,那个程序就能在几乎所有计算机以及安装了 Java 解释器的浏览器中运行。由于 Java 是一种全功能的编程语言,所以在向服务器发出一个请求之前,我们能先在客户端做完尽可能多的工作。例如,再也不必通过互联网传送一个请求表单,再由服务器确定其中是否存在一个拼写或者其他参数错误。大多数数据校验工作均可在客户端完成,没有必要坐在计算机前面焦急地等待服务器的响应。这样一来,不仅速度和响应的灵敏度得到了极大的提高,而且对网络和服务端造成的负担也可以明显减轻,这对保障互联网的畅通是至关重要的。

在客户端的编程中,安全问题是让人头痛的问题。点击一个 Web 站点的时候,可能会随同 HTML 网页传回任何东西,如 GIF 文件、脚本代码、编译好的 Java 代码以及 ActiveX 组件。Java 解释器内嵌于本地的 Web 浏览器中,在小应用程序装载时会检查有嫌疑的指令。而且小应用程序根本没有权力将文件写进磁盘或者删除文件,因此小应用程序是安全的。

在服务器端的编程中,向服务器发出的较复杂的请求时通常涉及对一个数据库进行操作(事务处理)。其中最常见的是发出一个数据库检索命令,得到结果后,服务器会把它格式化成 HTML 页,并作为结果传送回来。另外,有时需要在数据库中注册自己的名字,或者向服务器发出一份订单,这就涉及对那个数据库的修改。这类服务器请求必须通过服务器端的一些代码进行,Java 可以实现所有服务器端编程,写出的程序就称做“小服务程序”(Servlet)。

Java 是最流行的网络编程语言,主要用于开发网络应用程序和 Web 浏览,其主要的应用方面如下:

- (1) 所有面向对象的应用开发,包括面向对象的事件描述、处理、综合等。
- (2) 计算过程的可视化、可操作化的软件的开发。
- (3) 动态画面的设计,包括图形图像的调用。
- (4) 设计选择交互、定向交互、控制流程等。
- (5) Internet 的系统管理功能模块的设计,包括 Web 页面的动态设计、管理和交互操作设计等。
- (6) Intranet 上的软件开发。
- (7) 与各类数据库连接查询的 SQL 语句的实现。
- (8) 其他应用类型的程序。

1.1.3 Java 语言的特点

Java 语言具有鲜明的特点,因此使它在分布式网络应用、多平台应用、图形用户界面、Web 应用、多线程应用等软件的开发中成为方便高效的工具。

1) 开发和使用简单性

Java 的语法风格非常近似于 C++ 语言,但删改了 C++ 中的指针、操作符重载等一些易混淆的地方。在内存管理方面又提供了垃圾收集,这使程序员可以在实现程序功能方面投入更多的精力,而无需考虑诸如内存释放等枝节问题。C++ 中复杂而灵活的指针操作往往导致严重的错误,一向是开发调试人员深感棘手的问题,而这在 Java 中却不存在。Java 虚拟机还能为程序链接本地甚至远程的类库,用户不需要开发已经提供的功能,可以在任何应用程序中使用包中所提供的源代码,因此用户可以快速运行 Java 程序。

2) 分布式

Java 对 TCP/IP 协议(或 HTTP、FTP 协议)的支持使 Java 程序可以轻易地建立网络连接,并通过统一资源定位器 URL(Uniform Resource Locator)访问远程文件如同访问本地文件一样方便,非常适合分布环境的应用。

3) 面向对象性

Java 语言的设计是完全面向对象的,不允许定义独立于类的变量和方法(函数)。Java 以类和对象为基础,在对象中封装了该对象的状态变量以及相应的方法,实现了模块化和信息的隐藏。这就使程序的结构更为清晰,为继承和重用带来便利。

4) 安全性

Java 语言里删除了 C++ 中对象指针和释放内存等功能,避免出现非法内存操作的现象。当 Java 用来创建浏览器应用程序时,将语言功能和一些浏览器本身提供的功能结合起来,使它更安全。Java 运行系统将进行字节码检验,并记录对象的存储情况,将访问限制在安全范围之内。将本地的类与远程的类分开运行,阻止远程系统对本地系统的破坏。支持 Java 的浏览器还允许用户控制 Java 软件对本地系统的访问。

5) 健壮性

Java 在编译和运行程序时,都要对可能出现的问题进行检查,以消除错误的产生。Java 提供自动垃圾收集来进行内存管理,防止程序员在管理内存时容易产生的错误。通过集成的面向对象的例外处理机制防止系统的崩溃。

6) 平台无关性

Java 为支持它的各种操作系统提供了一致的应用程序接口(API)。在 API 界面上,所有 Java 程序都将不依赖于平台。程序开发人员无需考虑应用环境的硬件条件和操作系统结构,Java 的源程序被编译成与机器无关的字节码。执行时,Java 的运行系统将字节码转化为当前机器的机器码。

7) 解释执行

Java 程序经过编译形成字节码,然后在虚拟机上解释执行。这是 Java 程序能够独立于平台运行的基础。这也使程序有利于增量链接,从而加快开发过程。与由编译语言生成的针对平台的机器码相比,Java 字节码的执行速度要慢一些。

8) 多线程

Java 提供了内置的多线程支持,程序中 can 以方便地创建多个线程,使得在一个程序里可

以同时执行多个任务。比如,用不同的线程分别控制声音和图像,可以轻易构筑声像交织的复杂效果,而编程时只要分别安排各线程的工作,不必关心它们的合作,这也大大促进了程序的动态交互性和实时性。

1.2 Java 程序运行环境

Java 语言有两种开发环境:一种是命令行方式的 J2SDK(Java 2 Software Development Kit),其前身是 JDK(Java Developers Kits);另一种是集成开发环境,如 JBuilder、JPad Pro、WorkShop、Visual Age 等。两种开发环境都提供 Java 程序编译、运行、调试、存档等基本功能,不同的开发环境在使用的方便性上相差很大。

1.2.1 安装 J2SDK 开发环境

1) J2SDK

Java 不仅提供了一个丰富的语言和运行环境,而且还提供了一个免费的 Java 开发工具集 J2SDK。编程人员和最终用户可以利用这些工具来开发 Java 程序或调用 Java 内容。

为了学习和编写 Java 程序,通常需要下载两个包: J2SDK 和 J2SDK 文档(<http://java.sun.com/j2se/1.4.2/download.html>)。J2SDK 包含了 Java 开发工具(Developer Kit)和 Java 运行环境(Runtime Environment),以及 Java API 源码等,可以根据需要选择 Windows 版或者 Linux 版。而 J2SDK 文档是一个压缩包,以网页的形式提供了 Java API 的非常详尽的开发文档,是 Java 程序开发的最佳参考资料。

J2SDK 包括以下主要工具:

- (1) javac Java 语言编译器,输出结果为 Java 字节码。
- (2) java Java 字节码解释器,直接从类文件执行 Java 应用程序字节码。
- (3) javadoc 根据 Java 源代码及说明语句生成 HTML 格式的 API 文档。
- (4) appletviewer 小应用程序浏览工具,用于测试并运行 Java 小应用程序。
- (5) jar Java Archive 文件归档工具。
- (6) jdb Java 调试工具 Debugger。
- (7) javah C 代码处理工具,用于从 Java 类调用 C++ 代码。
- (8) javap Java 反汇编器,返回 Java 程序的成员变量及方法等信息。

2) 安装 J2SDK

下载的 Windows 版本的 J2SDK 安装程序是一个文件名中含有 windows 字样并以 .exe 结尾的可执行文件,如 J2SDK 1.4.2 的 Windows 安装程序是 j2sdk-1_4_2_05-windows-i586-p.exe。直接运行该安装程序,并根据向导的提示就可以非常简单地安装好 J2SDK 1.4.2 开发环境。假设 J2SDK 安装在 C:\j2sdk1.4.2_05 目录下。

3) 开发环境的配置

下面以 J2SDK1.4.2_05 为例说明开发环境的配置。

(1) 平台为 Windows 9X 时,需修改系统根目录下的 autoexec.bat 文件的 path 和 class-path 变量,例如:

```
set Java_HOME=C:\j2sdk1.4.2_05
set PATH=%Java_HOME%\bin
```

set CLASSPATH=%Java_HOME%\lib

path 环境变量指定操作系统到 C:\j2sdk1.4.2_05\bin 目录中查找 Java 工具,classpath 环境变量指定 Java 工具到 C:\j2sdk1.4.2_05\lib 目录下查找类文件。重新启动操作系统,上述设置生效。

(2) 平台为 Windows 2000 或 Windows XP 时,需修改环境变量,在环境变量中增加 classpath,并修改原有的环境变量 path。设置方法为依次进入“控制面板”→“系统”→“高级”→“环境变量”。在系统变量处,找到 Path,单击“编辑”,将 c:\j2sdk1.4.2_05\bin 加到后面,再单击“新建”,加入新的环境变量 classpath,其值设为 C:\j2sdk1.4.2_05\lib,单击“确定”按钮后设置生效。

1.2.2 集成开发环境 JBuilder

命令行方式的开发工具 JDK 对于程序开发不太方便。许多公司为编程人员提供了更方便的 Java 编程工具。JBuilder 提供了一个可视化的集成开发环境,包含了 Java 集成开发环境中的所有内容,可以方便地开发纯 Java Application 程序、Applet 程序、JavaBeans 程序以及兼容 J2EE 的分布式企业应用。JBuilder 是一个开发大型 Java 应用系统的首选工具。

1.2.3 JPad Pro 开发工具

JBuilder 的 Java 编程是立足于工程的,对计算机的硬件配置要求也比较高。JPad Pro 是一个小巧而又功能强大的 Java 编程工具,是一个完整的 Java Application 和 Java Applet 的开发环境,并且支持 HTML 及其他类型的文件。JPad Pro 使编写、编译、运行和测试复杂程序变得相对简单了;使用这个程序可以很容易地查找语法错误;可以在 Sun 的 J2SDK 工具和 Microsoft 的 Java SDK 工具中进行转换;可以使用内置的浏览器查找类、方法和包;JPad Pro 中还可以使用 JavaScript 或者 VBScript 编写工具,是一个开发小型 Java 应用系统的有效工具。

JPad Pro 的安装步骤:

(1) 在 Windows 平台下,直接运行 JPad Pro 的安装程序,出现欢迎和版权警告信息,如图 1.1 所示。

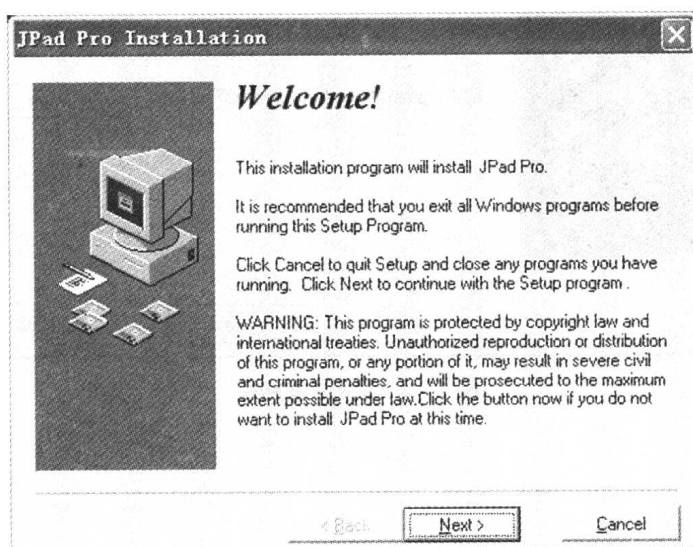


图 1.1 欢迎和版权警告信息

(2) 点击“Next”，将出现关于 JPad Pro 的版本信息，如图 1.2 所示。

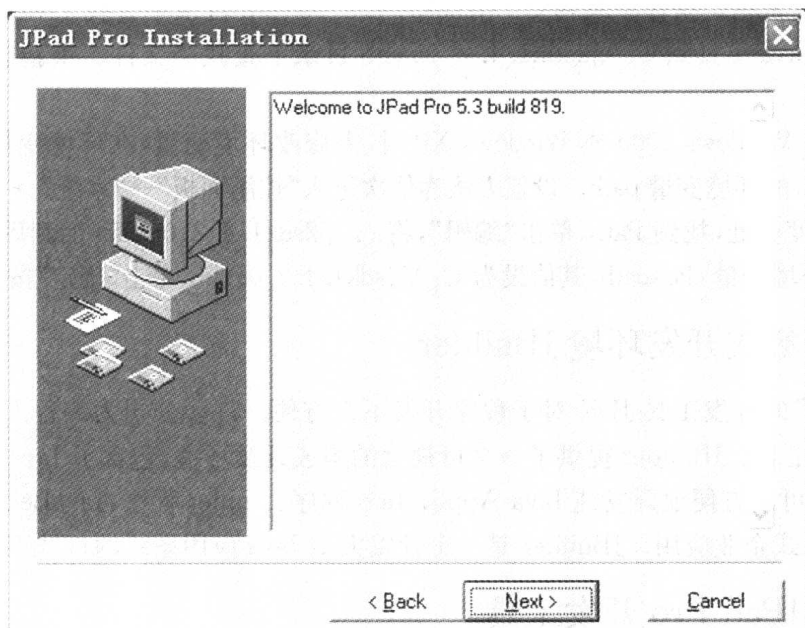


图 1.2 版本信息

(3) 点击“Next”，出现设置安装路径的窗口，选择“Browse”可以改变默认安装路径，如图 1.3和图 1.4 所示。

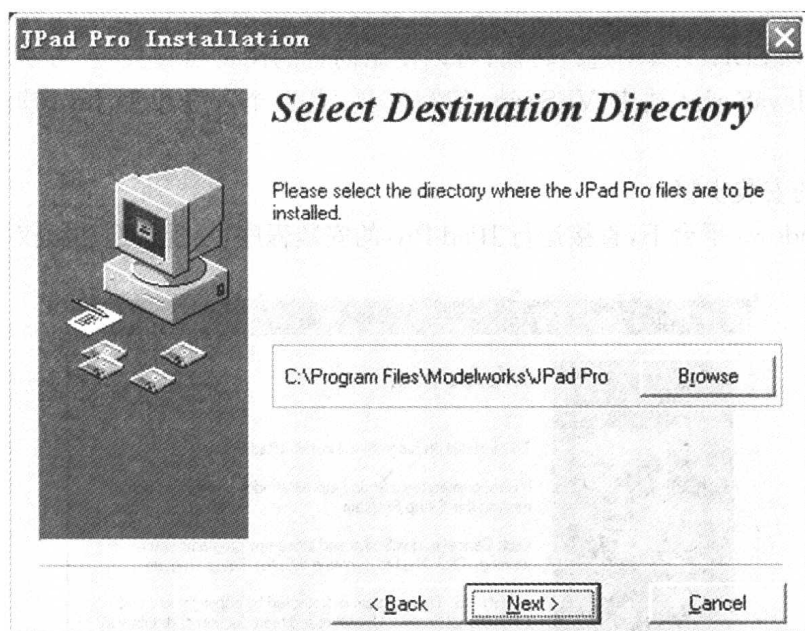


图 1.3 选择安装路径