

软件工程系列教材

Java

Java

程序设计

<http://www.tup.com.cn>

雍俊海
编著



清华大学出版社

软件工程系列教材

Java

程序设计

雍俊海 编著

清华大学出版社
北京

内 容 简 介

本书讲解学习 Java 语言的基本方法,介绍 Java 程序设计基础知识及其编程思想,内容包括 Java 语言的基础语法、面向对象的基本概念、数组、字符串、异常处理、文件和数据流、图形用户界面设计、小应用程序、线程、编程规范、网络程序设计、多媒体与图形学程序设计以及数据库程序设计等。本书的章节编排与书写以人们学习与认知过程为基础,与实际开发的需求相匹配。本书内容力求简洁,每章都附有习题,使得读者在轻松和欢乐之中迅速地了解与掌握 Java 程序设计的知识和方法,并应用到实践中去。

本书内容丰富易学,既可以作为计算机专业和非计算机专业的基础教材,也可以作为需要学习 Java 语言的工程人员和科技工作者的自学参考书。

版权所有,翻印必究。举报电话:010-62782989 13901104297 13801310933

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Java 程序设计/雍俊海编著. —北京:清华大学出版社,2004.8

(软件工程系列教材)

ISBN 7-302-08910-8

I. J… II. 雍… III. Java 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 060941 号

出 版 者:清华大学出版社

地 址:北京清华大学学研大厦

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

客户服务:010-62776969

责任编辑:龙啟铭

印 刷 者:北京市清华园胶印厂

装 订 者:三河市新茂装订有限公司

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:24.25 字数:556 千字

版 次:2004 年 8 月第 1 版 2004 年 8 月第 1 次印刷

书 号:ISBN 7-302-08910-8/TP·6308

印 数:1~5000

定 价:29.80 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或(010)62795704

序

当今社会是信息社会。以信息产业为主要标志的综合国力竞争日益激烈,软件产业作为信息产业的核心和国民经济信息化的基础,越来越受到世界各国的高度重视。整个世界的软件业正以极其迅猛的速度在向前发展。我国的软件人才要适应这种高速变化的时代变革,必须打破一些传统的思想观念,不断提高创新能力,才能在竞争日益激烈的国内外环境中取得卓越的成就。

要很好地迎接迅猛发展的科学技术浪潮对教育提出的巨大挑战,关键在于如何提高效率,包括软件开发、软件维护以及软件学习的效率。这种效率指的不仅是在很短的时间内完成特定的程序,而且还要求程序具有较高的执行效率和良好的稳定性。这就需要良好的程序设计与编写风格,以满足大规模软件生产的需求。本书能够在这一方面起到非常积极的作用。

本书作为计算机软件的基础性教材,其编排和设计很好地适应了信息产业迅猛变革所带来的各种需求,是教学新观念的产物。本书非常简洁,利用浅显的方法揭示各种基本概念的本质之处,通过翔实的示例展现 Java 语言的特点和编程的思想,使得学生通过学习本书在较短的时间内掌握较多的知识。本书注重实用,课后的习题贴切而又充满趣味,营造了一种轻松而快乐的学习氛围,培养和加强了学生的学习乐趣与学习成就感。

本书在内容编排上紧紧把握教育、产业与市场的实际需求。本书的出版将有助于培养我国高层次、实用型和复合式的软件人才,为我国的软件业走向世界作出积极的贡献。

中国工程院院士

2004 年 4 月

前 言

现代科学技术正在迅猛地发展着,计算机信息技术在其中发挥着巨大的作用。计算机技术已经渗透到各行各业,并推动着这些行业的迅速发展。因此,如何尽快地掌握计算机知识,学好一门计算机语言,已经成为现代人才普遍面临的一个基本问题。本书就是在这种背景下编写的。

首先,Java 语言本身就是可以满足这种需求的一种计算机语言。它远比计算机语言 C++ 简单,去掉了 C++ 语言中一些不易理解或容易出错的概念和语法。此外,因为 Java 是一种较新的计算机语言,所以它在面向对象和多线程特性上可以设计得比其他现有计算机语言更为纯粹一些。同时,Java 语言在网络、平台无关性和安全性方面的优点也比大部分计算机语言(如 C++ 语言)更为突出。这样,学习 Java 程序设计,应用 Java 语言实现算法就比较容易且节省时间。同时,编写出来的 Java 代码也就比较容易得到复用和移植。

其次,本书在编写之前分析了人们认知和学习的过程,借鉴了各种教材的优点,并加以补充和进一步完善。在本书编写过程中自始至终都要求能够体现这些特点,表达力求严谨准确,使得读者在阅读本书时既能轻松愉悦,又能高效地学到 Java 程序设计的知识。本书非常注重实用性,其内容的选取是建立在软件公司对 Java 程序设计的实际需求上。本书在各章之后还附有习题,以适应学习计算机语言实践性强的特点。这既方便读者自学,多做练习,以便更深刻地理解和掌握这门计算机语言;同时也方便授课教师检查学生对 Java 程序设计的掌握情况。本书的授课时间为 48~80 学时。在学时少的情况下,对于从第 8 章到最后一章的内容,授课教师可以自行选择部分章节讲解。在编写本书的过程中已经考虑了这种情况,所以这几章内容的相关性已经减到最弱的程度。这几章的内容在编写时也考虑了如何便于自学。

本书非常注重 Java 程序设计思想的建立,贯穿全书各个章节。本书系统地介绍了如何从实际的问题到用 Java 程序来解决这些问题;包括 Java 程序模型的建立、程序的分解机制、程序的编程规范和代码的编写。这些程序设计方法可以用来解决工程和科研中常见的实际问题。灵活地利用这些程序设计方法可以使得 Java 语言成为读者日常学习和工作中的一种有益的而且是非常方便的工具。

本书既可以作为计算机专业和非计算机专业的基础教材,也可以作为需要使用计算机的工程人员和科技工作者的自学参考书。清华大学的宿华同学参与了本书最后一章的编写。本书在编写与出版的过程中得到了许多朋友的帮助,这里一并表示诚挚的谢意。由于水平与时间所限,书中可能出现一些错误和缺陷,恳切希望广大读者特别是讲授此课程的老师批评指正。作者个人主页: <http://cgcad.thss.tsinghua.edu.cn/~junhai/chn/index.html>。

雍俊海 于清华园

2004 年 4 月 5 日

目 录

第 1 章 绪论	1
1.1 历史简介	1
1.2 特点	2
1.3 开发环境的建立	3
1.4 Java 程序及其执行过程	11
1.4.1 开发 Java 程序的工作流程	11
1.4.2 Java 程序的工作原理	17
1.5 本章小结	18
习题	19
第 2 章 结构化程序设计	20
2.1 标识符和关键字	20
2.2 基本数据类型、直接量和变量	24
2.2.1 基本数据类型	24
2.2.2 直接量	25
2.2.3 变量	26
2.3 运算符	29
2.3.1 算术运算符	30
2.3.2 关系运算符	31
2.3.3 布尔逻辑运算符	32
2.3.4 位运算符	33
2.3.5 赋值类运算符	35
2.3.6 条件运算符	36
2.3.7 其他运算符	36
2.4 控制结构	36
2.4.1 if 语句和 if-else 语句	37
2.4.2 switch 语句	39
2.4.3 for 语句	41
2.4.4 while 语句	42
2.4.5 do-while 语句	43
2.4.6 break 语句	43
2.4.7 continue 语句	45
2.5 结构化程序设计	46

2.6 本章小结	48
习题	49
第3章 面向对象程序设计	52
3.1 面向对象程序设计基本思想	52
3.2 类、实例、域和方法	53
3.2.1 类、域和方法的定义	54
3.2.2 引用数据类型、实例生成和几类特殊的方法	58
3.3 继承性	61
3.4 多态性	64
3.4.1 静态多态性	64
3.4.2 动态多态性	66
3.5 包	70
3.6 封装性	73
3.7 修饰词	76
3.8 接口	79
3.9 内部类	81
3.10 变量作用域范围与方法调用	86
3.10.1 变量作用域范围	86
3.10.2 方法调用	89
3.11 递归方法与压缩方法	93
3.12 本章小结	95
习题	95
第4章 数组、字符串与异常处理	98
4.1 数组	98
4.1.1 一维数组	98
4.1.2 多维数组	101
4.2 字符串	108
4.2.1 String	108
4.2.2 StringBuffer	113
4.3 异常处理	117
4.3.1 异常及其种类	117
4.3.2 异常产生	119
4.3.3 异常处理示例	119
4.3.4 自定义异常类型	124
4.4 本章小结	126
习题	126

第 5 章 文件与数据流	131
5.1 输入流与输出流	131
5.1.1 InputStream 和 FileInputStream	131
5.1.2 OutputStream 和 FileOutputStream	138
5.1.3 PrintStream	141
5.1.4 标准输入输出流的重定向	143
5.2 数据的输入流和输出流	145
5.3 带缓存的输入流和输出流	149
5.4 随机访问文件	152
5.5 读写器	155
5.5.1 Reader 和 Writer	156
5.5.2 FileReader 和 FileWriter	156
5.5.3 BufferedReader 和 BufferedWriter	158
5.5.4 PrintWriter	162
5.6 对象串行化	164
5.7 文件	169
5.8 本章小结	171
习题	172
 第 6 章 图形用户界面设计	173
6.1 基本 GUI 组件和容器	173
6.1.1 概述	173
6.1.2 JLabel、JTextField 和 JPasswordField	174
6.1.3 JButton、JCheckBox 和 JRadioButton	179
6.1.4 JComboBox 和 JList	184
6.1.5 JTextArea 和 JScrollPane	187
6.1.6 JSlider	189
6.1.7 JFrame 和 JPanel	191
6.2 布局管理器	193
6.2.1 FlowLayout	193
6.2.2 BorderLayout	194
6.2.3 GridLayout	196
6.2.4 BoxLayout	197
6.2.5 GridBagLayout	199
6.2.6 CardLayout	201
6.2.7 组合布局方式	203
6.2.8 自定义的布局管理器	205
6.3 事件处理模型	211

6.3.1	事件处理模型概述	211
6.3.2	鼠标事件处理	215
6.3.3	键盘事件处理	221
6.4	高级图形用户界面	223
6.4.1	菜单	224
6.4.2	多文档界面	230
6.5	本章小结	234
	习题	235
第 7 章	小应用程序	236
7.1	编写小应用程序	236
7.1.1	生命周期	236
7.1.2	小应用程序	239
7.2	编写网页	243
7.2.1	基本 HTML 语句	244
7.2.2	归档文件	245
7.2.3	网页中的小应用程序参数	247
7.3	应用程序与小应用程序	250
7.4	本章小结	254
	习题	254
第 8 章	多线程程序设计	256
8.1	编写线程程序	256
8.1.1	通过类 Thread 构造线程类	256
8.1.2	通过接口 Runnable 构造线程类	259
8.2	线程的生命周期	261
8.3	多线程的同步处理	266
8.3.1	同步方法	266
8.3.2	同步语句	271
8.3.3	成员方法 wait/notify/notifyAll	274
8.3.4	死锁问题	280
8.4	本章小结	284
	习题	284
第 9 章	编程规范	286
9.1	程序编写规范	286
9.1.1	命名规范	286
9.1.2	排版规范	288

9.1.3 语句.....	290
9.1.4 文件组织.....	291
9.2 程序调试	293
9.3 文档注释	298
9.4 本章小结	306
习题.....	307
第 10 章 网络程序设计	308
10.1 网络资源定位器.....	308
10.1.1 网络地址.....	308
10.1.2 网络资源定位器.....	310
10.1.3 从网络资源定位器获取网络资源.....	312
10.2 基于 TCP 的网络程序设计	313
10.3 基于 UDP 的网络程序设计	322
10.4 本章小结.....	326
习题.....	326
第 11 章 多媒体与图形学程序设计	328
11.1 声音播放.....	328
11.2 图像显示.....	331
11.3 图形显示.....	333
11.4 计算机动画.....	338
11.4.1 通过定时器控制动画速度.....	338
11.4.2 动画制作.....	341
11.4.3 提高动画质量.....	345
11.5 本章小结.....	349
习题.....	349
第 12 章 数据库程序设计	351
12.1 基本原理.....	351
12.1.1 JDBC 驱动程序类型	351
12.1.2 JNDI 原理	353
12.2 开发环境的建立.....	354
12.2.1 安装数据库和建立测试数据.....	354
12.2.2 选择、获取和安装 JDBC 驱动程序	357
12.2.3 安装与设置 JNDI	358
12.3 数据库程序设计.....	358
12.4 JDBC 的 API 简介	364

12.4.1	数据源.....	366
12.4.2	数据库连接.....	366
12.4.3	SQL 语句执行器	367
12.4.4	SQL 查询结果集	368
12.5	性能优化.....	370
12.5.1	预编译语句.....	370
12.5.2	SQL 语句批处理机制	372
12.5.3	连接池.....	372
12.6	本章小结.....	373
	习题.....	373
	参考文献.....	375

第 1 章 绪 论

自从 1946 年第一台 ENIAC 计算机在美国宾夕法尼亚州(Pennsylvania)诞生以来,计算机产业的发展速度以及计算机向其他领域渗透的速度已经远远出乎人们的意料。现在,它已经成为各行各业的基本工具。在这期间,计算机语言本身也在飞速发展。其发展方向之一就是使得计算机语言越来越接近于人们的思维习惯。按照这种发展方向来分,计算机语言可以分为第一代(机器)语言、第二代(低级)语言和第三代(高级)语言。这种发展方向使得程序越来越容易编写、阅读、维护、复用和移植。Java 语言就是这样发展起来的一种高级语言:易学易用,并迅速受到推崇。目前,Java 语言已经成为最常用的计算机语言之一。本章将简单介绍 Java 的历史和特点,以及从建立 Java 环境到运行 Java 程序的整个流程。

1.1 历史简介

Java 语言是一种很新的计算机语言,它的历史很短。Java 语言的前身是 Oak 计算机语言。1991 年,Sun 公司为了占领智能消费型电子产品的市场,资助了一个“绿色项目”。这个项目是由 James Gosling 负责的,主要是开发用于智能消费型电子产品的语言,即 Oak 语言。Oak 语言是在 C 和 C++ 计算机语言的基础上进行简化和改进的一种语言。项目进行不久,Sun 公司意识到已经存在一种叫做 Oak 的计算机语言。于是,Sun 公司重新给这种语言命名为 Java 计算机语言。这样,James Gosling 也就成为了 Java 语言的创始人。

但 Java 语言很快就遇到了一些困难,因为 Sun 公司发现智能消费型电子产品发展没有预想的那样快,而且当时 Sun 公司在竞争一个大项目时失败了。Sun 公司差一点就要取消这个“绿色项目”。到 1993 年,Sun 公司重新分析市场需求,认为网络具有很好的发展前景,而且 Java 语言似乎非常适合网络编程。于是 Sun 公司将 Java 语言的应用背景转向网络市场,为网页增加了“动态的内容”。

Sun 公司的这次市场策略转变是非常成功的。1995 年,当 Sun 公司在“Sun World 95”大会上第一次正式公布 Java 语言时,立即引起了巨大的轰动,因为那时正是网络“泡沫经济”的时代,网络处于“狂热”的时期。Java 语言为网络的发展开辟了一个新纪元。同年,Java 语言就被计算机杂志《PC Magazine》评为 1995 年十大优秀科技产品(当年计算机产品就此一项入选)。微软公司总裁比尔·盖茨当时的一句话“Java 语言是有史以来最卓越的计算机程序设计语言”也反映了当时人们对 Java 语言的普遍评价。许多计算机公司都开始支持和开发 Java 产品,其中包括 IBM 公司、Apple 公司和 Oracle 公司等。1996 年,Sun 公司专门成立 Javasoft 分公司来发展 Java。Java 从此得到了迅猛的发展和

广泛的应用。这种速度也是前所未有的。

1999 年, Sun 重新组织 Java 平台的集成方法, 加强 Java 企业级应用平台的功能。目前, Java 程序可以支持智能消费型电子产品的开发, 各种应用程序的开发(包括个人应用程序和企业级的应用程序), 尤其是网络程序的开发。目前, Java 语言拥有“互联网上的世界语”的美称。

1.2 特 点

Java 语言的特点与其历史发展是相关的。它之所以能够受到如此众多的好评以及拥有如此迅猛的发展速度, 与其语言本身的特点是分不开的。其主要特点总结如下。

(1) **简单性**: 从 Java 语言的发展史可以了解到, Java 语言是在 C 和 C++ 计算机语言的基础上进行简化和改进的一种新型计算机语言。它去掉了 C 和 C++ 中最难正确应用的指针和最难理解的多重继承技术等内容, 通过垃圾自动回收机制简化了程序内存管理, 统一了各种数据类型在不同操作系统平台上所占用的内存大小。Java 程序的简单性是其得以迅速普及的最重要原因之一。

(2) **网络特性**: Java 语言正是因为其对互联网络的良好支持而受到推崇并迅速被广泛应用的。Java 语言是目前对网络支持最全面、与网络关系最密切的一种计算机语言。

(3) **面向对象**: 由于 Java 语言是一种新型计算机语言, 没有兼容过程式计算机语言的负担, 所以 Java 语言在面向对象的特性上就比 C++ 语言更为彻底。面向对象模型是一种模拟人类社会和人解决实际问题的模型, 因此更符合人们的思维习惯。与其他程序设计方式相比, 采用这种方式进行程序设计似乎显得更为合理。它的优点是使得程序与人类的行为更一致, 而且容易扩充和维护。它的缺点是程序在开发的过程中常常变得越来越庞大。

(4) **平台无关性/可移植性**: Java 语言的设计目标是让其程序不用修改就可以在任何一种计算机平台上运行。解决异构操作系统兼容性问题是一个很艰巨的任务。Sun 公司提供的 Java 语言也没有完全做到这一点。在 Java 语言的说明书中, Sun 公司用权重(weight)的轻重来表示其提供的类或成员方法与计算机平台的相关性大小。不过, 总的来说, Java 语言在这一方面是做得最好的计算机语言。

(5) **鲁棒性**: 鲁棒性指的是程序执行的稳定性, 也常常称为健壮性。Java 语言设计者在设计 Java 语言的过程中就一直考虑如何减少编写程序的过程中可能产生的错误。Java 在编译和执行的过程中都会进行严格的检查, 以减少错误的发生。Java 语言中的垃圾自动回收机制和严格的异常处理机制在很大程度上提高了程序的鲁棒性。其实, Java 语言的简单性同时就在一定程度上保证了程序的鲁棒性。

(6) **安全性**: 在网络上运行的 Java 程序是符合网络安全协议的。另外, 在执行 Java 程序的过程中, Java 虚拟机也会对程序的安全性进行检测。所以, 一般说来, Java 程序是安全的, 它不会访问或修改未经允许的内存或文件。

(7) **多线程性**: 这主要用来处理复杂事务或需要并行的事务。Java 虚拟机本身就是

一个多线程的程序。采用多线程机制是提高程序运行效率的一种方法,当然也增加了程序的设计难度。

(8) **解释性**: Java 语言是一种解释执行的语言。这是 Java 语言的一个缺点,因为解释执行的语言一般会比编译执行的语言(如 C 和 C++ 语言)的执行效率要低。

总而言之,Java 语言是一种易学好用,健壮性高,但执行效率相对较低的计算机语言。它适合于各种对执行时间要求不是很苛刻的应用程序。用 Java 语言编写程序一般会比用其他计算机语言编写程序花费更少的时间,而且调试所需的时间一般也会较短。对于计算机初学者或正打算开始学习一门计算机语言的工程师或教学科研工作者来说,选择 Java 程序设计进行学习是一个很好的方案。

1.3 开发环境的建立

要学好任何一门计算机语言,都必须加强练习,即编写程序解决各种实际问题。只有多练习,勇于尝试,并善于总结,才能真正掌握和精通一门计算机语言。要练习,首先就需要建立起 Java 的开发环境。

要在一台计算机上编写和运行 Java 程序,首先应当在这台计算机上建立起 Java 开发环境。建立 Java 开发环境就是要在计算机上安装 Java 开发工具包并在计算机中设置相应参数,使得 Java 开发工具包可以在计算机中顺利地得到正确运行。Sun 公司免费提供的开发工具包的早期版本简称为 JDK (Java Developer's Kit)。现在推出的开发工具包分成三种版本:J2SE(Java™ 2 Platform, Standard Edition)、J2EE(Java™ 2 Platform, Enterprise Edition)和 J2ME(Java™ 2 Platform, Micro Edition)。J2SE 是用于工作站和个人计算机(简称 PC 机)的标准开发工具包,J2EE 是应用于企业级开发的工具包,J2ME 主要是用于开发智能消费型电子产品(如移动电话和汽车导航系统等)。本书介绍的是基于 J2SE 的 Java 程序设计。建立基于 J2SE 的 Java 开发环境的步骤如下:

- (1) 下载 J2SE 安装程序;
- (2) 运行 J2SE 安装程序,安装 J2SE;
- (3) 设置环境变量运行路径(path)和类路径(classpath);
- (4) 下载 J2SE 的在线帮助文档。

J2SE 安装程序可以从 Sun 公司的 Javasoft 分公司网站(<http://java.sun.com>)下载。下载时要注意自己计算机的操作系统。下载的安装程序应当与自己计算机的操作系统相匹配,而且一般选择最新的版本。安装程序下载完了,就可以运行安装程序。安装过程只要遵循安装程序提供的指示进行就可以了。安装完成之后就进入了步骤(3)。这个步骤的目标是给计算机设置 Java 工具包运行的环境变量:运行路径(path)和类路径(classpath)。其中,运行路径(path)变量记录的是各个运行程序所在的路径。系统根据这个变量的值来查找运行程序。所以在运行路径(path)中加上 J2SE 运行程序所在的路径,就会使得在运行 J2SE 程序时不必输入全路径名。类路径(classpath)环境变量是用来记录 J2SE 类库所在的路径,这是 J2SE 需要的一个环境变量。通常将类路径(classpath)

的值设为当前路径(用一个点“.”表示)和 J2SE 类库所在的路径(两个路径间用分号隔开)。假设 J2SE 的安装路径是 C:\j2sdk (如果实际的安装路径不是 C:\j2sdk, 则用实际的安装路径替代 C:\j2sdk), 则需要设置的环境变量及其值分别为:

```
path=%path%;C:\j2sdk\bin
```

```
classpath=.;C:\j2sdk\lib
```

其中“%path%”表示原有路径。实际的具体设置步骤在不同的操作系统中会有些不同, 而且有多种实现方法。无论是在什么操作系统下或采用什么方法, 只要给计算机系统正确地设置上面的两个环境变量就行了。下面分别介绍在操作系统 Microsoft Windows NT、2000、XP、Me 和 98 下设置这两个环境变量的方法。

1. 在 Microsoft Windows NT、2000 或 XP 下设置运行路径(path)和类路径(classpath)的步骤

如图 1.1 所示, 用鼠标的左键依次单击桌面菜单项“开始”、“设置”和“控制面板”。

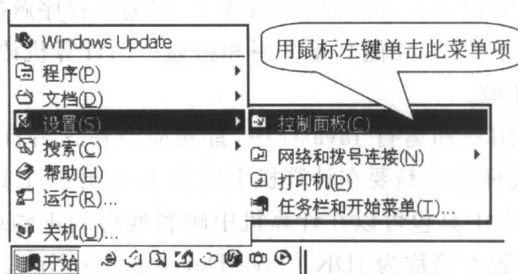


图 1.1 操作系统的桌面菜单

这样就会弹出如图 1.2 所示的对话框。

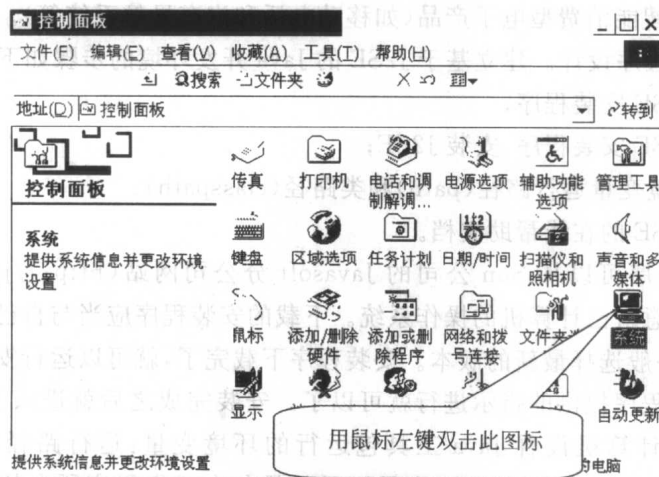


图 1.2 控制面板

用鼠标的左键双击控制面板上的“系统”图标, 就会弹出一个对话框。这个对话框在

Microsoft Windows NT、2000 或 XP 中略有所不同。在 Microsoft Windows NT 下选取该对话框的“环境”选项卡,在 Microsoft Windows 2000 或 XP 下选取“高级”选项卡。Microsoft Windows 2000 的“高级”选项卡如图 1.3 所示。

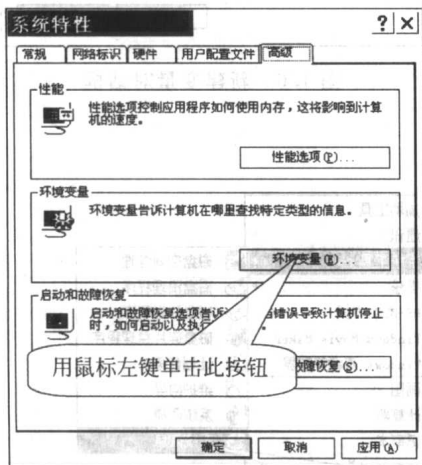


图 1.3 系统特性对话框



图 1.4 环境变量对话框

下面只以 Microsoft Windows 2000 操作系统为例来阐述设置环境变量的后继步骤,在 Microsoft Windows NT 或 XP 操作系统下的操作步骤是相似的。用鼠标左键单击图 1.3 所示的“系统特性”对话框中“高级”选项卡的“环境变量”按钮,计算机显示出“环境变量”对话框,如图 1.4 所示。在该对话框中,分别给用户变量表和系统变量表设置或添加(如果该变量表中没有该变量)运行路径(path)和类路径(classpath)这两个变量。一般来说,这两个变量表均会有运行路径(path)这个变量。先在变量表中选中该变量(即 path),再用鼠标左键单击按钮“编辑”(如果没有 path 变量,就用鼠标左键单击“新建”按钮),这时出现如图 1.5 所示的对话框。在变量值的编辑框中,在原有值的末尾加入“;C:\j2sdk\bin”。其中,分号用来分隔原来的路径和新加入的 J2SE 运行路径。输入完成之后,用鼠标左键单击“确定”按钮,关闭对话框,同时回到图 1.4 所示的对话框。通常在用户变量列表和系统变量列表中都不会有类路径(classpath)这个变量,需要用鼠标左键分别单击在图 1.4 所示的两个变量列表下面的“新建”按钮,创建这个变量(如果已经有 classpath 变量,就用鼠标左键单击“编辑”按钮,进行编辑)。这时系统出现“新建变量”的对话框,如图 1.6 所示,输入相应变量名及其值。输入完成之后,用鼠标左键单击“确定”按钮。当这两个变量都设置完之后,用鼠标左键依次单击图 1.4 和图 1.3 所示的对话框中的“确定”按钮,就完成了 J2SE 环境变量的设置。

2. 在 Microsoft Windows Me 下设置运行路径(path)和类路径(classpath)的步骤

如图 1.7 所示,用鼠标的左键依次单击桌面菜单项“开始”、“程序”、“附件”、“系统工具”和“系统信息”。

这样就会出现如图 1.8 所示的窗口。从中选取“工具”菜单下的“系统配置实用程序”菜单项。

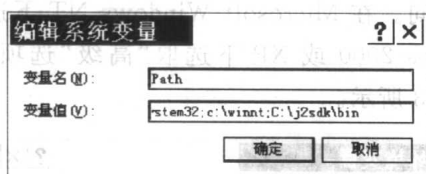


图 1.5 编辑变量对话框

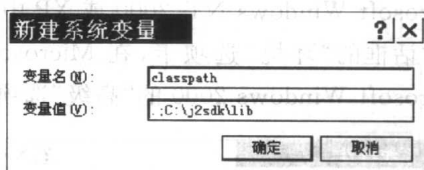


图 1.6 新建变量对话框

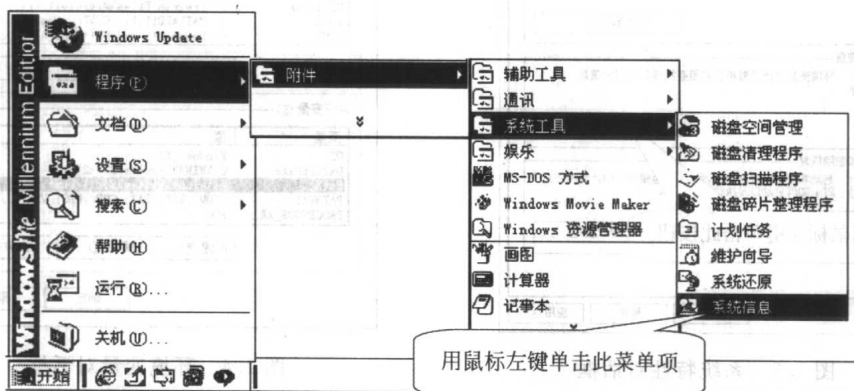


图 1.7 Windows Me 操作系统的桌面菜单

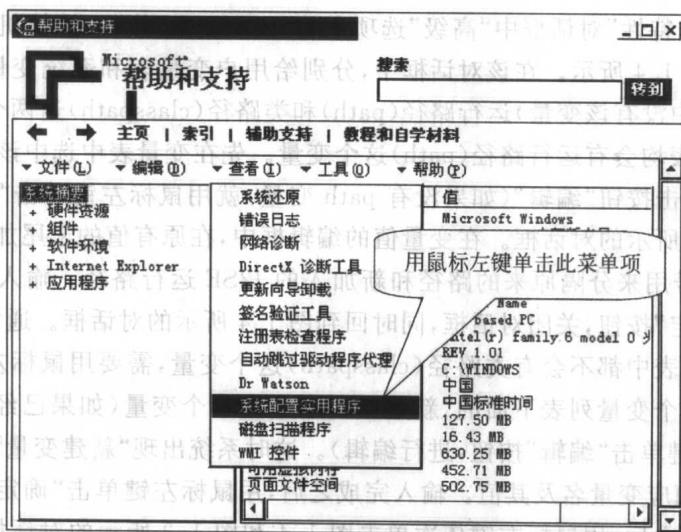


图 1.8 系统帮助和支持对话框

这时系统就会弹出如图 1.9 所示的“系统配置实用程序”对话框，选取其中的“环境”选项卡。

在该选项卡中，如果已经有运行路径(path)或类路径(classpath)变量，就用鼠标左键单击按钮“编辑”，进行编辑；否则鼠标左键单击“新建”按钮创建所缺少的变量。在“系统配置实用程序”对话框中一般会有运行路径(path)变量，这时先在变量表中选中该变量