

高等院校
信息技术应用型
规划教材

丛书顾问 冯博琴
丛书主编 陶 进 侯冬梅

Java程序设计教程

石瑞峰 边 琦 主 编
冯晓龙 刘 岩 副主编



清华大学出版社

高等院校
信息技术应用型
规划教材

丛书顾问 冯博琴
丛书主编 陶 进 侯冬梅

Java程序设计教程

石瑞峰 边 琦 主 编
冯晓龙 刘 岩 副主编

清华大学出版社
北京

内 容 简 介

本书在教学内容选取上以“必需、够用”为度,重点突出关键能力和程序设计能力的训练,以问题引导的方式,建立学生的学习兴趣,提高其求知欲望;以案例驱动的方式,培养学生的学习动机;以实训项目的方式,培养学生应用语言的能力,体现了“工学结合”的教学理念。全书共10章,内容包括Java语言概述、Java语言基础、方法和数组、面向对象程序设计、数据流操作、Java的图形用户界面、Applet及其应用、异常处理、线程技术、综合项目设计。书中给出了面向对象程序设计的大量案例和习题,力求通过程序设计实例讲清相关概念、原理、方法和应用,为教师的备课、学生的学习提供最大方便。

本书可作为高等院校计算机、信息管理相关专业的教材,也可作为Java程序设计的培训教材,还可作为程序设计人员的参考书。

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 程序设计教程/石瑞峰,边琦主编. —北京:清华大学出版社,2012.8

(高等院校信息技术应用型规划教材)

ISBN 978-7-302-28999-9

I. ①J… II. ①石… ②边… III. ①JAVA 语言—程序设计—高等职业教育—教材
IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 122209 号

责任编辑:孟毅新

封面设计:常雪影

责任校对:袁芳

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>, 010-62795764

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市兴旺装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:16.25

字 数:373千字

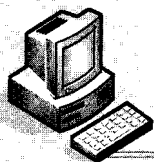
版 次:2012年8月第1版

印 次:2012年8月第1次印刷

印 数:1~3000

定 价:33.00元

产品编号, 041319-01



前言

Java 语言是一种纯面向对象的计算机程序设计语言,其“一次编译,多次运行”的特性使该语言在 Web 编程方面得到了广泛的应用。自 1995 年诞生以来,随着 Internet 的发展,Java 技术日益成熟,Java 语言已为各种应用提供了非常丰富的软件开发工具以及应用程序开发包,生成具有中立体系结构的中间代码,具有很强的可移植性。简单的语法结构、面向对象的设计理念、安全的接口编程、多线程的应用机制使其成为网络应用开发的主流语言,并受到了广大程序设计人员的青睐。

本书在内容组织上主要采用案例驱动的形式,由适当的任务导入,引导学生了解相关的问题以及解决问题的知识要点。为更好地体现“工学结合”的教学思想,书中使用了大量的实际案例,通过对案例的分析及代码实现,给出相关的知识细节,比较全面地讲述了 Java 的编程思想和技术应用。在内容递进层次上,从 Java 语言基础入手,讲解 Java 的基本概念、基本语法结构。在面向对象程序设计方面,重点是对象、类、继承、多态等概念的应用,突出面向对象的程序设计思想。此外,讲解数据流操作、多线程处理以及 Java 的异常处理;最后设计的几个小型项目,综合应用了 Java 的相关知识点。总之,在教学内容选取上,基本理论以“够用”为度,重点突出关键能力和程序设计能力的训练。

本书的特点如下。

- (1) 知识点由浅入深,内容编排循序渐进,易教易学。
- (2) 基本理论知识与设计应用并重,内容新颖,富有启发性。
- (3) 立足基本概念、方法,注重基本技能的培养。
- (4) 语言通俗,简明实用,适于自学。
- (5) 理论与实践并重,理论融入实践项目中,注重学生的创新精神与实践能力的培养,分析问题与解决问题能力的培养。
- (6) 课程实践应用性强,以实践应用为主线,选取实用性、针对性较强的案例,使学生不仅可以掌握 Java 的基本知识,还可以学习程序设计的基本方法与思路。
- (7) 案例教学贯穿始终,任务驱动,教、学、练合一。

本书由石瑞峰、边琦任主编,冯晓龙、刘岩任副主编。具体分工为:石瑞峰编写第 1 章、第 10 章;边琦编写第 3 章;冯晓龙编写第 4 章、第 8 章;刘岩编写第 6 章、第 7 章;赵金考、李钰编写第 2 章;王瑾瑜编写第 9 章;吕润涛编写第 5 章。全书由石瑞峰负责规划与统稿。



书中所有实例的程序代码均在 JDK 1.6 环境下调试通过,读者可从清华大学出版社网站(<http://www.tup.com.cn>)上下载使用,如有疑问可与作者联系:srx0628@126.com。

由于编者水平有限,书中难免有不足之处,真诚希望得到广大专家和读者的批评和指正。

编 者

2012 年 7 月



目 录

第 1 章 Java 语言概述	1
1.1 初识 Java 语言	1
1.1.1 目标	1
1.1.2 情境导入	1
1.1.3 案例分析	2
1.1.4 案例实施	2
1.1.5 Java 语言的发展历程	7
1.1.6 Java 语言的工作原理及其特点	8
1.2 Java 应用程序开发	11
1.2.1 目标	11
1.2.2 情境导入	11
1.2.3 案例分析	11
1.2.4 案例实施	12
1.2.5 JVM 机制	16
1.2.6 Java Application 程序与 Java Applet 程序	17
1.3 小结	18
习题	18
第 2 章 Java 语言基础	20
2.1 基本数据	20
2.1.1 目标	20
2.1.2 情境导入	20
2.1.3 案例分析	21
2.1.4 案例实施	21
2.1.5 数据类型	22
2.1.6 常量和变量	29
2.1.7 运算符和表达式	31
2.2 语句结构	37
2.2.1 目标	37
2.2.2 情境导入	37

2.2.3	案例分析	37
2.2.4	案例实施	38
2.2.5	基本语句	39
2.2.6	选择语句	40
2.2.7	循环语句	46
2.2.8	跳转语句	53
2.2.9	自主演练	54
2.3	小结	56
习题		56
第3章	方法和数组	60
3.1	方法	60
3.1.1	目标	60
3.1.2	情境导入	60
3.1.3	案例分析	61
3.1.4	案例实施	61
3.1.5	方法定义及其应用	62
3.1.6	变量的作用域	66
3.1.7	自主演练	67
3.2	数组	69
3.2.1	目标	69
3.2.2	情境导入	69
3.2.3	案例分析	69
3.2.4	案例实施	70
3.2.5	一维数组	71
3.2.6	多维数组	74
3.2.7	字符数组	75
3.2.8	自主演练	78
3.3	小结	80
习题		81
第4章	面向对象程序设计	83
4.1	对象	83
4.1.1	目标	83
4.1.2	情境导入	83
4.1.3	案例分析	83
4.1.4	案例实施	84
4.1.5	对象的创建与使用	85

4.1.6	面向对象的特征	89
4.1.7	面向过程与面向对象	92
4.1.8	自主演练	93
4.2	类	93
4.2.1	目标	93
4.2.2	情境导入	94
4.2.3	案例分析	94
4.2.4	案例实施	94
4.2.5	类的创建与应用	95
4.2.6	类的继承与多态	103
4.2.7	自主演练	111
4.3	接口和包	112
4.3.1	目标	112
4.3.2	情境导入	112
4.3.3	案例分析	112
4.3.4	案例实施	113
4.3.5	接口的定义与实现	113
4.3.6	包的创建与使用	117
4.3.7	自主演练	120
4.4	小结	122
	习题	122
第 5 章	数据流操作	124
5.1	数据流概述	124
5.1.1	目标	124
5.1.2	情境导入	124
5.1.3	案例分析	124
5.1.4	案例实施	125
5.1.5	流的概念及流的包装	125
5.1.6	输入/输出类	127
5.2	数据流应用	129
5.2.1	目标	129
5.2.2	情境导入	129
5.2.3	案例分析	129
5.2.4	案例实施	129
5.2.5	字节流	130
5.2.6	字符流	132
5.2.7	自主演练	135

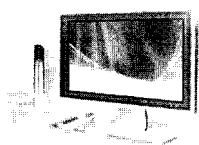
5.3 文件类及其应用	135
5.3.1 目标	135
5.3.2 情境导入	136
5.3.3 案例分析	136
5.3.4 案例实施	136
5.3.5 文件的创建与使用	137
5.3.6 随机文件流	139
5.3.7 自主演练	139
5.4 小结	140
习题	141
第6章 Java 的图形用户界面	142
6.1 图形界面设计	142
6.1.1 目标	142
6.1.2 情境导入	142
6.1.3 案例分析	142
6.1.4 案例实施	143
6.1.5 界面构成	144
6.1.6 JFC 的组成	144
6.1.7 自主演练	145
6.2 事件和事件处理	145
6.2.1 目标	145
6.2.2 情境导入	146
6.2.3 案例分析	146
6.2.4 案例实施	146
6.2.5 事件类	147
6.2.6 事件处理	148
6.2.7 自主演练	155
6.3 基本控件组件与常用容器组件	156
6.3.1 目标	156
6.3.2 情境导入	156
6.3.3 案例分析	156
6.3.4 案例实施	157
6.3.5 基本控件组件	158
6.3.6 常用容器组件	159
6.3.7 自主演练	161
6.4 布局设计	161
6.4.1 目标	162

6.4.2	情境导入	162
6.4.3	案例分析	162
6.4.4	案例实施	162
6.4.5	布局管理器类与布局模型	163
6.4.6	自主演练	166
6.5	小结	166
	习题	166
第7章	Applet 及其应用	169
7.1	初识 Applet	169
7.1.1	目标	169
7.1.2	情境导入	169
7.1.3	案例分析	169
7.1.4	案例实施	169
7.1.5	Applet 基础	171
7.1.6	Applet 与 Applet 类	172
7.1.7	自主演练	173
7.2	Applet 应用程序	173
7.2.1	目标	173
7.2.2	情境导入	173
7.2.3	案例分析	174
7.2.4	案例实施	174
7.2.5	Applet 的开发步骤	175
7.2.6	Applet 的参数传递	176
7.2.7	Applet 中的 GUI	177
7.2.8	自主演练	178
7.3	Applet 多媒体编程	178
7.3.1	目标	178
7.3.2	情境导入	178
7.3.3	案例分析	178
7.3.4	案例实施	179
7.3.5	文字与图形	180
7.3.6	声音与动画	180
7.3.7	自主演练	182
7.4	小结	183
	习题	183

第 8 章	异常处理	185
8.1	异常概述	185
8.1.1	目标	185
8.1.2	情境导入	185
8.1.3	案例分析	185
8.1.4	案例实施	186
8.1.5	异常与异常类	187
8.1.6	异常处理机制	188
8.2	异常处理	188
8.2.1	目标	188
8.2.2	情境导入	188
8.2.3	案例分析	189
8.2.4	案例实施	189
8.2.5	异常的捕获与抛出	191
8.2.6	finally 语句	194
8.2.7	自主演练	196
8.3	小结	196
	习题	197
第 9 章	线程技术	198
9.1	线程的 Java 实现	198
9.1.1	目标	198
9.1.2	情境导入	198
9.1.3	案例分析	199
9.1.4	案例实施	199
9.1.5	基本概念	201
9.1.6	线程的创建	202
9.1.7	线程的状态	203
9.1.8	线程的调度与控制	204
9.2	多线程处理	209
9.2.1	目标	209
9.2.2	情境导入	209
9.2.3	案例分析	209
9.2.4	案例实施	209
9.2.5	同步与锁机制	212
9.2.6	线程的等待与唤醒	213
9.2.7	自主演练	215



9.3 线程的其他特性	216
9.3.1 目标	216
9.3.2 情境导入	216
9.3.3 案例分析	216
9.3.4 案例实施	217
9.3.5 主线程和守护线程	218
9.3.6 线程组与线程池	220
9.3.7 死锁	221
9.4 小结	222
习题	222
第 10 章 综合项目设计	224
10.1 目标	224
10.2 情境导入	224
10.3 案例分析	225
10.4 案例实施	226
10.5 小结	245
参考文献	247



第 1 章 Java 语言概述

Java 语言是一种面向对象的程序设计语言,可以用来编制跨平台的应用软件,其前身是 Oak 语言,语法格式类似于 C++,但要比 C++ 简单,更容易理解,因为它丢弃了 C++ 中的一些难于理解的内容,如指针操作、运算符重载等,避免了对计算机内存的直接访问与操作,使得数据操作的安全性得到进一步的加强,同时大大降低了编程的难度,并以其实用、与平台无关性、安全性等优点,受到广大程序设计人员的青睐。用 Java 语言开发的 Applet 程序应用于全球信息网络平台,改变了以往那种静态呆板的页面,增强了网页的互动性,给人以无穷的视觉效果与想象空间。

1.1 初识 Java 语言

1.1.1 目标

通过安装、配置 Java 的开发工具包,了解 Java 的运行环境。初步认识 Java 程序的语法格式,了解 Java 语言的发展历程、特点,掌握 Java 程序的工作原理。

1.1.2 情境导入

一种语言的使用是离不开它的应用环境的,对于 Java 来说,它的语言要素非常丰富,是一个纯面向对象的程序设计语言,即利用类和对象的机制将数据和方法封装在一起,通过统一的接口与外界进行交互,程序内部通过对象的继承关系,有效地组织整个程序。程序的设计开发离不开良好的开发环境,Java 的应用开发使用专用的工具集,即 JDK(Java Development Kits)套件,它是由 SUN 公司开发、完全免费的工具集,现有 3 种版本:Java 2 Standard Edition、Java 2 Micro Edition 和 Java 2 Enterprise Edition。

Java 2 Standard Edition 简称 J2SE,即标准版(Standard Edition),它是最常用的一个版本,使用 Java Hotspot 虚拟机来提高其性能,其 Java 类库包含了 Java 语言的所有特性;Java 2 Enterprise Edition 简称 J2EE,即企业版(Enterprise Edition),这种版本用于开发 J2EE 应用程序,针对的设备主要是后端的 Server;Java 2 Micro Edition 简称 J2ME,即微型版(Micro Edition),主要用于移动设备、嵌入式设备上的 Java 应用程序开发。J2SDK(Java 2 Software Development Kits)是新版本 JDK 的特定称呼,它包括 Java 的编

译器、解释器、调试器以及 Java API(Application Programming Interface)类库等,所有这些应用开发工具都为 Java 语言的广泛应用奠定了良好的基础。例如,要在计算机屏幕上显示这样一条简单的信息:“学习 Java 语言,需要一步步深入!”,Java 语言将如何解决这个问题呢?

1.1.3 案例分析

信息的输出显示是一个应用程序不可或缺的部分,数据的最终处理结果都将通过不同的显示输出给用户。上面提到的问题表面上看似简单,只是显示一条信息,但是针对信息的显示,不同的语言使用不同的指令,而且指令的使用环境也有所区别。首先需要了解 Java 程序的开发环境,根据 Sun 公司发布的各种版本的 Java 开发包,安装标准版本的工具包即可完成这样的任务,因为这些工具包具有很强的通用性。其次需要了解开发工具的环境变量的配置。

Java 提供了大量 API 类库,当程序调用到系统提供的公用类库时,需要知道这些内容放置的准确位置,类库文件的搜索路径需要在系统环境中设置,同时也可设置一些快捷操作方式以便灵活使用 Java 开发工具。在开发包安装完成后,需要测试开发工具是否正常工作,这样可避免程序调试时出现不必要的错误。

开发一个程序显示上面的信息将涉及 Java 源程序的编制、编译、执行等步骤。由于 Java 语言是以类为基础的面向对象语言,基本的构造单元是类,对象是一种执行单位,所以设计的 Java 程序将以类的方式来组织,在类的内部使用不同的方法来处理程序中用到的各种数据对象。

1.1.4 案例实施

根据 Sun 公司提供的相关技术资料,使用 J2SE 版本的工具包即可搭建好 Java 的开发环境。Java 开发软件的安装步骤如下。

1. 下载 Java 2 SDK 工具包

下载地址是: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>, 下载页面如图 1-1 所示。在页面 Java SE Downloads 中选择 JDK, 接下来在 Java 版本选择页中选择运行的平台类型, 然后单击下载 jdk-6u24-windows-i586.exe 工具包, 如图 1-2 所示。

2. 安装 J2SDK 工具包

运行已下载的 jdk-6u24-windows-i586.exe 工具包, 安装中可使用默认的安装方式, 也可以定制安装目录以及选择需要安装的内容, 安装画面如图 1-3 所示。

Java 2 SDK 属于分离式(命令式)开发环境, 它由一组工具组成, 每一个工具完成一定的工作, 所有的工具组成一个工具包。在该开发环境下主要包含的工具如下所示。

(1) Javac

Javac 是 Java 的编译器, 它将 Java 源程序编译为字节码程序, 要求 Java 的源文件必须使用 .java 作为扩展名。

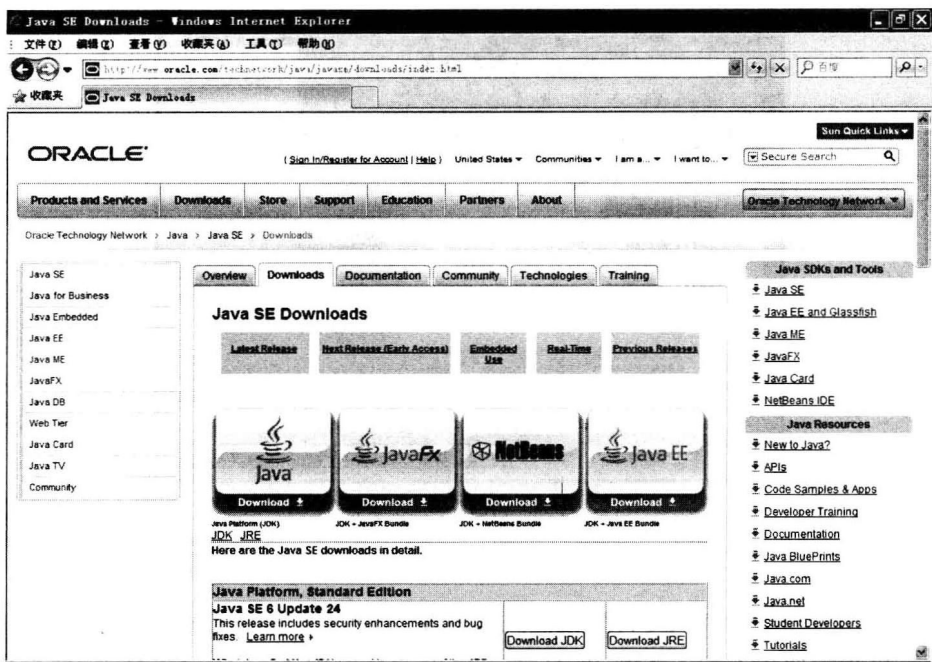


图 1-1 Java 下载主页

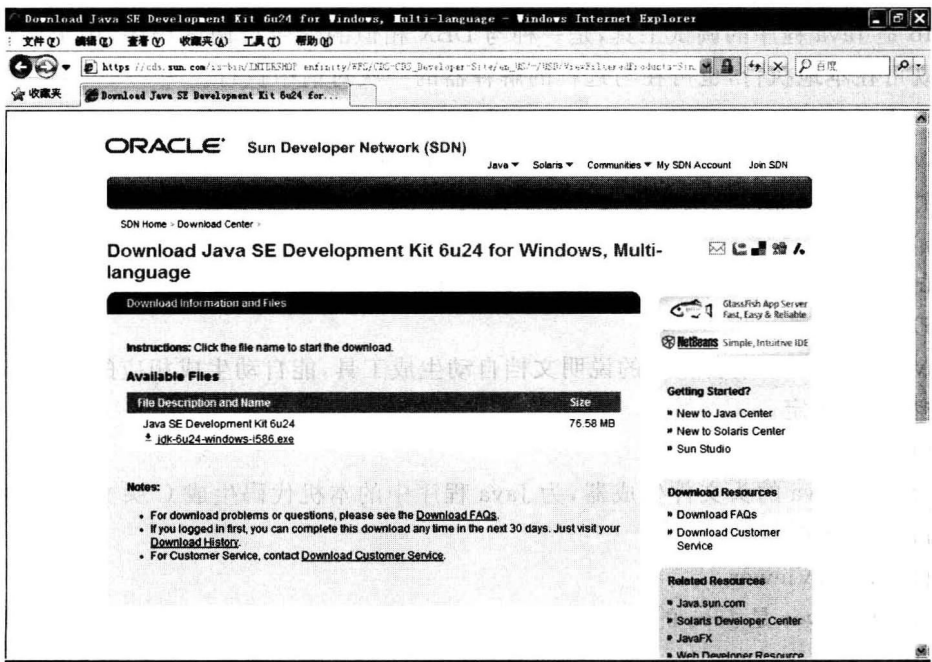


图 1-2 Java 工具包下载页

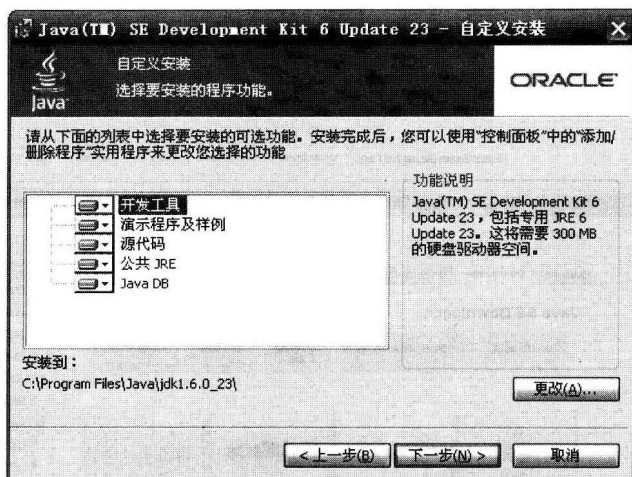


图 1-3 J2SDK 安装界面

(2) Java

Java 是 Java 语言的字节码程序解释器,只是一个基于虚拟机 JVM 平台的程序。不同的计算机系统使用不同的解释器,任何一台机器只要配备了 Java 解释器,就可以运行字节码程序,而不管这种字节码程序是在何种平台上生成的。

(3) Jdb

Jdb 是 Java 程序的调试工具,是一种与 DBX 相似的命令行调试程序,用于调试 Java 类,它既可在本地执行,也可在与远程的解释器的一次对话中执行。

(4) Javap

Javap 是 Java 的反编译器,但它并不把代码反编译为 Java 源代码,而是把字节代码反汇编为由 Java 虚拟机规范定义的字节代码指令。

(5) Jar

Jar 是 Java 的归档工具,将一系列文件合并到单个压缩文件里。

(6) Javadoc

Javadoc 是 Java 语言程序的说明文档自动生成工具,能自动生成相应的文档资料,文档的类型可以指定。

(7) Javah

Javah 是 Java 的头文件生成器,为 Java 程序中的本机代码生成 C 头文件,要注意类所在的路径,即包。

(8) AppletViewer

AppletViewer 是 Java 的 Application 程序的宿主浏览器,用于测试并运行 Java 的小应用程序。

3. 配置运行环境的参数

打开 Windows 系统的“系统属性”对话框,如图 1-4 所示,单击对话框中的“环境变量”按钮,进入“环境变量”对话框,如图 1-5 所示。

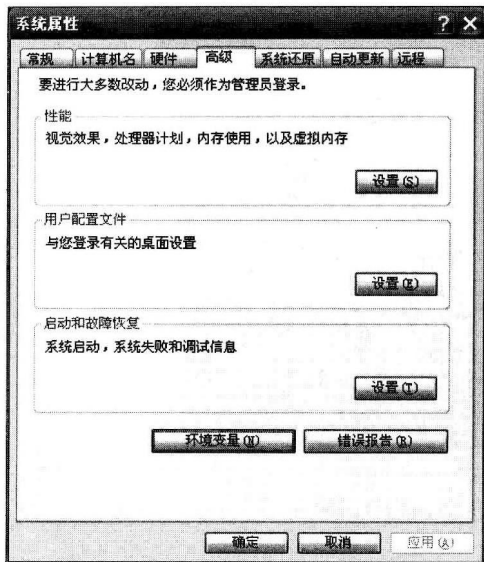


图 1-4 “系统属性”对话框

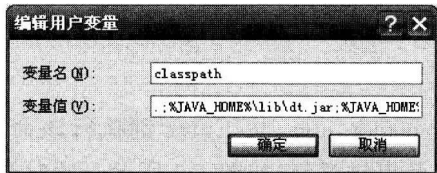
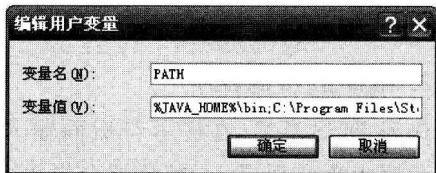


图 1-5 “环境变量”对话框

新建一个用户变量 `JAVA_HOME`, 变量的值设置为 `C:\Program Files\Java\jdk 1.6.0_23`, 使其指向 JDK 的安装路径, 在这个路径下能够找到 `bin`、`lib` 等目录。注意用到的路径要与安装时设定的路径一致, 否则系统将无法在指定的路径下搜索到所需的内容。

新建一个用户变量 `classpath`, 变量的值设置为 `“.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar”`, 其作用是指定 J2SDK 的 Java 类路径, 告诉 Java 类装载器到哪里去寻找第三方提供的类以及用户自定义的类。`classpath` 值的最前面的 `“.”` 表示当前目录, 如图 1-6 所示。

在用户变量 `PATH` 添加 J2SDK 命令的搜索路径, 添加的值是 `“%JAVA_HOME%\bin”`, 用分号将该值与 `PATH` 中原有的值分隔, 设置界面如图 1-7 所示。用户变量的配置只对当前用户有效, 如果将环境参数配置为系统变量, 则对所有用户有效。

图 1-6 `classpath` 配置界面图 1-7 `PATH` 配置界面

4. 运行环境测试

一般在开发包安装完成后, 需要重新启动操作系统, 然后进入 Windows 的 DOS 环境, 在 DOS 提示符后输入 `“Java”` 后按 `Enter` 键, 如果出现图 1-8 所示的 Java 指令使用信息, 说明 Java 运行环境的参数配置已经生效, 此时即可编译 Java 的源程序, 编译无错后即可执行 Java 编译后的类文件。

经过上面的安装、配置操作, Java 的开发环境已经就绪, 接下来就是 Java 应用程序的