

21世纪高等学校计算机基础教育系列教材

21 SHIJI GAODENG XUEXIAO JISUANJI JICHU JIAOYU XILIE JIAOCAI

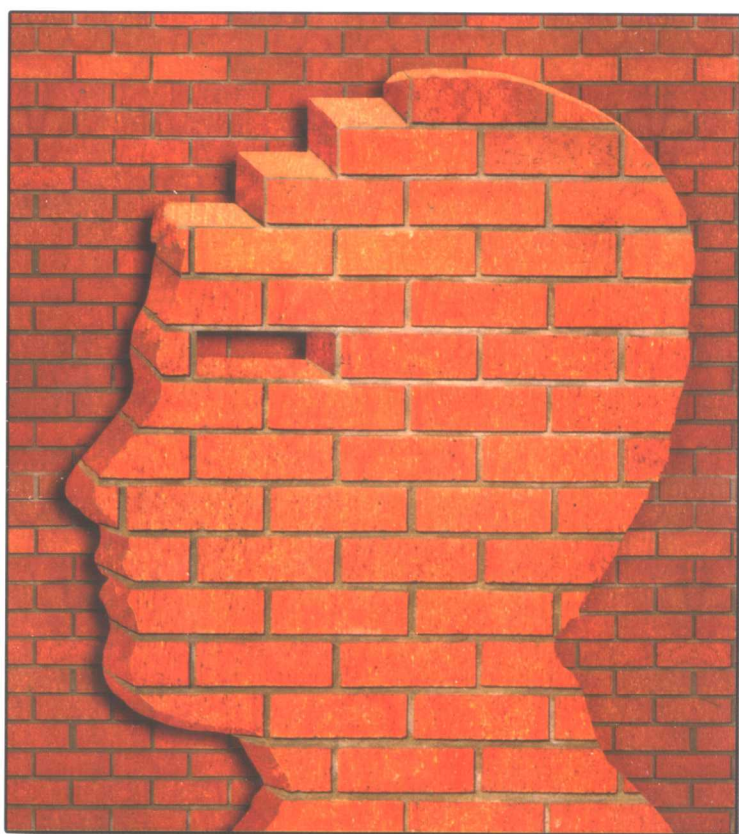
Java 语言 程序设计

案例 教程

■ 陈 炜 主编

■ 沈 昕 宋燕红

刘桂玲 魏雪英 等 编著



人民邮电出版社
POSTS & TELECOM PRESS

21 世纪高等学校计算机基础教育系列教材

Java 语言程序设计案例教程

陈炜 主编

沈昕 宋燕红 刘桂玲 魏雪英 等编著

人民邮电出版社

图书在版编目 (CIP) 数据

Java 语言程序设计案例教程 / 陈炜主编; 沈昕等编著. 北京: 人民邮电出版社, 2005.1
(21 世纪高等学校计算机基础教育系列教材)

ISBN 7-115-12995-9

I. J... II. ①陈...②沈... 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 (2004) 第 143482 号

内容简介

全书共分为 11 章, 主要介绍了 Java 编程的基础知识、基本算法和流程控制、数组和方法、面向对象编程、常见错误和异常处理、线程及其操作、图形用户界面的设计与实现、Applet 基础、输入与输出、Java 网络编程等内容。同时, 还配合知识的应用详细讲解了 87 个实例, 并且提供了大量习题。

全书以计算机实例操作为主线, 以实例带动知识点的学习。通过学习大量实用和经典的编程实例, 读者可以快速掌握、应用所学的 Java 语言编程知识。

本书适于作为高等院校非计算机专业和高职高专计算机专业的教材, 还适于作为初学者的自学用书。

21 世纪高等学校计算机基础教育系列教材

Java 语言程序设计案例教程

◆ 主 编 陈 炜

编 著 沈 昕 宋燕红 刘桂玲 魏雪英 等

责任编辑 邹文波

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

读者热线: 010-67129259

北京隆昌伟业印刷有限公司印刷

新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张: 21

字数: 510 千字

2005 年 1 月第 1 版

印数: 1-5 000 册

2005 年 1 月北京第 1 次印刷

ISBN 7-115-12995-9/TP · 4392

定价: 27.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

前 言

Java 语言是由美国 SUN 公司开发的一种具有面向对象、分布式和可移植等性能的功能强大的多线程动态计算机编程语言。同时,Java 语言还是一种跨平台的程序设计语言,可以在各种类型的计算机和操作系统上运行。Java 语言非常适合于企业网络和 Internet 环境,现在已成为 Internet 中最受欢迎、最有影响的编程语言之一。另外,它还因其可移植、安全性能高和网络移动性等逐渐成为一种行业标准。

Java 语言以其独有的开放性、跨平台性和面向网络的交互性风靡全球,是目前最常用的计算机编程语言,也是主要的网络开发语言之一。

本书使用 Java 2 SDK 1.4.2_05 作为开发工具,来介绍 Java 语言。本书的特点是内容广泛且通俗易懂,以实例为主,全面介绍了用 Java 语言编程所需的各方面知识。

全书以计算机实例操作为主线,以实例带动知识点的学习,通过大量实用、经典的编程实例,来介绍 Java 语言,使读者可以快速掌握、应用所学的 Java 语言编程知识。本书内容由浅及深、由易到难、循序渐进、图文并茂,理论与实际制作相结合,读者在阅读学习时不但能够快速入门,而且可以达到较高的水平。教师在使用本教材进行教学时,可以一边指导学生练习各章的实例,一边讲解各实例知识进阶中的知识和概念,将它们有机地结合在一起,以达到事半功倍的效果。

本书的作者由大学的计算机教师和计算机公司的程序设计员组成,他们有丰富的教学 and 实践经验,并总结了一套任务驱动式的教学方法。采用这种方法学习的学生将更容易掌握 Java 语言的编程方法和编程技巧。

本书主编为陈炜,策划和主审校为沈大林。参加本书编写工作的主要人员有陈炜、沈昕、宋燕红、刘桂玲、魏雪英、胡野红和肖柠朴等。本书的电子教案等相关资料请到人民邮电出版社网站(www.ptpress.com.cn)上下载。

由于作者水平有限,加上时间仓促,书中难免有不妥之处,恳请广大读者批评指正。

编 者

目 录

第 1 章 Java 简介	1
1.1 Java 的历史与语言结构	1
1.1.1 Java 的发展简史	1
1.1.2 计算机语言的发展	2
1.1.3 Java 语言的结构	2
1.1.4 Java 语言的特点	3
1.2 Java 的安装	3
1.2.1 Java 语言的开发工具	3
1.2.2 安装 Java 2 SDK	4
1.2.3 更新系统环境变量	7
1.3 Java 的应用	10
1.3.1 Java 应用程序与 Java 小程序	10
1.3.2 DOS	11
1.3.3 HTML	14
习题	18
第 2 章 Java 编程入门	19
2.1 Java 程序的结构	19
2.1.1 Java 程序布局	19
2.1.2 Java 源程序的组成	19
2.1.3 注解语句	20
2.1.4 打印语句	21
2.1.5 转义符号和统一码	21
2.2 变量和常量	23
2.2.1 数据类型	23
2.2.2 变量和常量	24
2.3 运算符和表达式	29
2.3.1 算术运算符和算术表达式	29
2.3.2 关系运算符和关系表达式	30
2.3.3 逻辑运算符和逻辑表达式	31
2.3.4 赋值运算符和赋值表达式	32
2.3.5 条件运算符和条件表达式	32
2.3.6 位运算符和位表达式	33

2.3.7 数据类型转换	34
2.3.8 优先级	37
2.4 对象变量	37
2.4.1 String 类	38
2.4.2 Math 类	39
2.4.3 字符串和数字的相互转换	40
2.4.4 键盘输入数据	41
习题	47
第3章 基本算法和流程控制	50
3.1 算法和控制结构	50
3.1.1 算法	50
3.1.2 算法的描述方法	51
3.2 选择结构语句	53
3.2.1 if 语句	54
3.2.2 switch 语句	57
3.3 循环结构语句	63
3.3.1 while 语句	63
3.3.2 for 语句	67
3.3.3 跳转语句	72
3.3.4 多重循环语句	73
3.4 应用实例	78
习题	90
第4章 数组和方法	96
4.1 数组	96
4.1.1 什么是数组	96
4.1.2 数组的使用	97
4.1.3 二维数组	100
4.1.4 什么是排序	104
4.2 方法	108
4.2.1 方法	108
4.2.2 递归	113
习题	117
第5章 面向对象程序设计	121
5.1 面向对象程序设计概述	121
5.1.1 什么是面向对象	121
5.1.2 常用术语的含义	121

5.1.3 程序设计过程	122
5.1.4 面向对象程序设计的特点	123
5.2 Java 语言的面向对象程序设计	125
5.2.1 域	125
5.2.2 构造方法	127
5.2.3 方法	128
5.3 方法的使用和对象数组	134
5.3.1 调用方法	134
5.3.2 访问方法	136
5.3.3 方法重载	138
5.3.4 this	139
5.3.5 对象比较和对象数组	142
5.4 类的继承和多态	149
5.4.1 类继承的概念	149
5.4.2 类继承的实现	150
5.4.3 super	152
5.4.4 类的多态	154
5.5 包和访问控制符	156
5.5.1 包	156
5.5.2 常见系统包	157
5.5.3 访问控制符	158
5.6 接口	162
5.6.1 接口的概念	162
5.6.2 接口的实现	163
5.6.3 抽象类和抽象方法	164
习题	165
第 6 章 常见错误和异常处理	170
6.1 常见错误	170
6.1.1 编译错误	170
6.1.2 运行错误	171
6.1.3 逻辑错误	172
6.2 异常处理	173
6.2.1 什么是异常处理	173
6.2.2 Exception 类	173
6.2.3 异常处理的过程	175
6.2.4 多异常的处理	179
6.2.5 自定义异常处理	179
习题	182

第 7 章 线程及其操作	184
7.1 线程的实现	184
7.1.1 线程的定义	184
7.1.2 创建线程	185
7.2 线程的状态及调度	192
7.2.1 线程的状态	192
7.2.2 线程的调度与优先级	194
7.2.3 控制线程	196
7.3 线程的同步	197
习题	198
第 8 章 图形用户界面的设计与实现	200
8.1 Java 图形用户界面概述	200
8.2 标准组件	202
8.2.1 组件分类	202
8.2.2 容器类组件	202
8.2.3 控制组件	203
8.2.4 Component 类	203
8.3 简单图形用户界面	204
8.3.1 Frame 与 Panel	204
8.3.2 字体设置	207
8.3.3 绘制简单图形	209
8.3.4 设置颜色	215
8.3.5 显示图像	217
8.4 Java 事件处理	218
8.4.1 Java 事件处理机制	218
8.4.2 按钮点击事件	221
8.4.3 捕获窗口事件	222
8.4.4 键盘事件	223
8.4.5 鼠标事件	225
8.4.6 焦点事件	227
8.4.7 事件适配器	229
8.5 布局管理	232
8.5.1 FlowLayout 类	232
8.5.2 BorderLayout 类	235
8.5.3 GridLayout 类	237
8.5.4 CardLayout 类	239
8.5.5 GridBagLayout 类	242

8.6 常用标准组件.....	242
8.6.1 按钮.....	242
8.6.2 标签.....	245
8.6.3 文本组件.....	246
8.6.4 复选框.....	250
8.6.5 单选按钮.....	252
8.6.6 下拉列表.....	254
8.6.7 列表.....	255
8.6.8 滚动条.....	257
8.6.9 画布.....	258
8.6.10 对话框.....	259
8.6.11 文件对话框.....	260
8.7 Swing 组件	260
8.7.1 JApplet 类	262
8.7.2 JFrame 类	262
8.7.3 JLabel 类	263
8.7.4 JTextField 类	264
8.7.5 JPasswordField 类	265
8.7.6 JButton 类	266
8.7.7 JCheckBox 类	268
8.7.8 JRadioButton 类	269
8.7.9 JComboBox 类	270
习题	272
第9章 Applet 基础	274
9.1 Java Applet 概念.....	274
9.1.1 Java Applet 的概念.....	274
9.1.2 Applet 类	275
9.1.3 Applet 的主要方法及生命周期.....	276
9.1.4 Applet 标记.....	278
9.1.5 HTML 与 Applet 参数传递	279
9.2 Applet 绘制图形与图像处理.....	281
9.2.1 绘制图形.....	281
9.2.2 图像处理.....	285
9.3 Applet 的其他应用.....	288
9.3.1 声音处理.....	288
9.3.2 动画特效.....	291
9.3.3 其他特效.....	296
习题	296

第 10 章 输入与输出	297
10.1 输入与输出基础	297
10.1.1 流的概念	297
10.1.2 字节流与字符流	297
10.1.3 标准输入/输出	300
10.2 文件输入/输出	302
10.2.1 Java 文件和目录管理	302
10.2.2 文件输入/输出流	305
10.2.3 随机存取文件	307
习题	309
第 11 章 Java 网络编程	311
11.1 网络基础	311
11.1.1 TCP/IP	311
11.1.2 通信端口	312
11.1.3 URL	313
11.1.4 客户机/服务器模式	313
11.1.5 Java 网络编程	313
11.2 InetAddress 编程	314
11.3 URL 编程	316
11.3.1 创建 URL 对象	316
11.3.2 获取 URL 对象的属性	317
11.4 Socket 编程	318
11.4.1 Socket 与 ServerSocket 类	318
11.4.2 Socket 通信	320
11.5 数据报通信	325
习题	326

第 1 章

Java 简介

Java 语言是目前最常用的计算机编程语言，也是主要的网络开发语言之一。Java 语言具有面向对象、分布式和多线程等先进高级计算机语言的特点，同时它还因可移植性、安全性能高和网络移动性等逐渐成为一种行业标准。对于初次接触计算机编程语言的人来说，Java 语言简单易学，不需要长时间的培训就可以编写出适合企业或个人需要的程序。

1.1 Java 的历史与语言结构

在揭开 Java 编程世界的神秘面纱之前，让我们先来了解一下它的简史、语言结构和特点。

1.1.1 Java 的发展简史

1991 年，Java 语言诞生于美国 SUN 计算机公司，它是由“绿色工程”（Green Project）小组开发的，最初的名字不是 Java，而是 Oak（橡树），当时是为一些小型家用消费电子产品而开发的一个分布式代码管理系统。使用者可以通过 E-mail 对电视机、洗衣机等家用电器进行控制操作。虽然 Oak 的网络安全性能很好，但是由于当时市场和语言本身的问题，整个计划并没有达到预期的目标。眼看 Oak 就要夭折时，Internet 的迅速发展拯救了它。

1994 年，“绿色工程”小组将他们的开发转向了 Internet，用 Oak 编写了一系列网络应用程序，像网络浏览器 WebRunner 等。Oak 的新生让 SUN 公司看到了其发展的意义，于是给 Oak 起一个更响亮的名字——Java。

1995 年，随着 Java 在 Internet 和多媒体上的发展，它得到了全球计算机业的高度重视。Netscape 和 Microsoft 公司都先后公布了支持 Java 语言的浏览器，像 Netscape Navigator 2.0 以上和 Microsoft Explore 2.0 以上的版本。SUN 公司也推出了一种用 Java 编写的、可嵌入浏览器内部的小程序 Applet，使得 Web 页面上出现了更丰富多彩的图像与动画。

Java 最初的 1.0 版和 1.1 版在设计上有不少缺点，SUN 公司在 1.2 版上倾注了大量的心血，使其更加完善，并加入了许多新的设计。鉴于 1.2

版与之前版本的巨大差别, SUN 公司将 1.2 版以及其以后版本命名为“Java 2”。目前, Java 2 的正式版本已经发展到 1.4 版, 测试版本为 1.5 版。

1.1.2 计算机语言的发展

在介绍 Java 语言原理之前, 先让我们了解一下计算机语言的发展。

在计算机被发明后, 如何编写程序操作计算机这个难题就一直困扰着科学家。最初是使用机器语言 (Machine Code), 也就是计算机中央处理器 (CPU) 本身所使用的语言。因为它可以直接被 CPU 使用, 所以表达准确、运行速度非常快。但显而易见, 由一组组十六进制数字组成的机器语言对大多数人来说很难编写、阅读和理解。在这种情况下, 计算机很难被广泛使用和推广。

汇编语言 (Assembly Language) 的出现使编写计算机程序变得更容易。具有英文意义的命令名称代替了一组组简单的数字, 一条条语句的出现, 也使程序内容变得清楚, 易于理解, 但是计算机只能读懂机器语言, 所以用汇编语言编写的程序要先用汇编程序 (Assembler) 翻译成机器语言, CPU 才能运行。如今汇编语言被计算机高级专业人员广泛使用, 但是, 汇编语言需要用许多条命令去完成一个极其简单的任务, 编写过程繁琐且耗时长, 所以它很难普及。

高级语言 (High-level Language) 使每一条语句的功能大大加强, 同时还保持了易写、易读和易于理解的特性, Java 语言就是一种高级语言。通过对高级语言的学习, 编写计算机语言对普通人来说终于再不是一座不可征服的高山, 但是对于计算机的 CPU 来说, 高级语言太笼统太简单了, 它需要一个功能强大的翻译器来帮助理解。翻译器有两种, 一种是编译器 (Compiler), 另一种是解释器 (Interpreter)。

编译器和解释器的作用都是将用高级语言写好的程序翻译成计算机能够理解的机器语言, 所不同的是翻译的方式。

编译器的翻译方式是把程序全部翻译成机器语言, 然后 CPU 再运行翻译好的机器语言。

解释器的翻译方式是在程序运行时每翻译一句高级语言, 就传递给 CPU 运行这部分翻译好的机器语言, 也就是边翻译边运行。

用编译器程序运行速度快, 但翻译时间过长; 用解释器翻译时间较短, 但运行速度很慢。那么 Java 语言使用的是哪一种翻译方法呢? 下面将对此逐步讨论。

1.1.3 Java 语言的结构

事实上, Java 两种翻译方法都使用了, 但是改进很大, 使翻译和运行的时间都很理想。Java 并没有用编译器直接将程序翻译成机器语言, 而是先将程序翻译成字节码 (Byte Code), 这样可适用于各种操作系统平台, 而且翻译的时间也大大减少。当然, 这种字节码格式的程序, 计算机的 CPU 是不可能理解的, 所以 Java 就通过一种名为 Java 虚拟机 (Java Virtual Machine, JVM) 的程序, 再将字节码翻译成机器语言。因为第二步翻译是在程序运行过程中进行的, 所以有时候也可以把 Java 虚拟机看成是一种编译器, 但是翻译速度远比一般的编译器要快。而

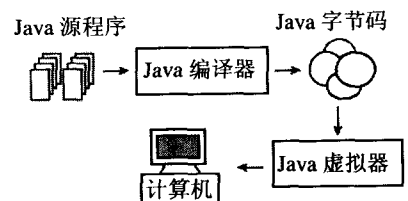


图 1-1 Java 语言的翻译方法示意图

且 Java 虚拟机是针对每一台计算机的,也就是说当同一个程序在不同的计算机上运行时,Java 编译器将其翻译成一样的字节码文件,Java 虚拟机再根据所在计算机 CPU 的不同,将字节码文件翻译成相应的机器语言。Java 语言的翻译方法如图 1-1 所示。

1.1.4 Java 语言的特点

Java 语言具有以下几个特点。

(1) 结构简单。Java 语言的程序构成与 C 语言和 C++ 语言类似,但是 Java 语言摒弃了 C 语言和 C++ 语言的复杂、不安全等特性。例如,指针的操作和内存的管理等都简单多了。此外,Java 语言提供了种类丰富、功能强大的类库,提高了编程效率。

(2) 面向对象。在现实世界中,任何实体都可以看作是一个对象,对象具有状态和行为两大特征。在 Java 语言中,没有采用传统的、以过程为中心的编程方法,而是采用以对象为中心,通过对象之间的调用来解决问题的编程方法。

(3) 与平台无关。使用 Java 语言编写的应用程序不需要进行任何修改,就可以在不同的软、硬件平台上运行。这主要是通过 Java 虚拟机(JVM)来实现的。

(4) 可靠性。因为 Java 语言最初的设计目的是应用于电子类家庭消费产品,所以要求较高的可靠性。例如,Java 语言提供了异常处理机制,有效地避免了因程序编写错误而导致的死机现象。

(5) 安全性。现今的 Java 语言主要应用于网络应用程序的开发,因此对安全性有很高的要求。如果没有安全保证,用户运行从网络下载的 Java 语言应用程序是十分危险的。Java 语言通过使用编译器和解释器,在很大程度上避免了病毒程序的产生和网络程序对本地系统的破坏。

(6) 多线程。多线程是指在一个程序中可以同时执行多个简单任务。线程也被称为轻量进程,是一个传统大进程里分出来的独立的、可并发执行的单位。C 语言和 C++ 语言采用单线程体系结构,而 Java 语言支持多线程技术。

1.2 Java 的安装

随着 Java 语言的广泛使用,许多公司都推出了自己的 Java 语言的开发工具,虽然它们的集成开发环境(IDE)不尽相同,各有侧重方向,但是在 Java 基本应用上是一样的。下面将介绍几种时下比较流行的 Java 语言的开发工具。

1.2.1 Java 语言的开发工具

(1) Visual J#.NET 是世界上最大的软件公司 Microsoft 出品的 Visual Studio.NET 家族中的一名负责 Java 开发的成员,其取代了 Visual Studio 中的 Visual J++。它的集成开发环境非常友好,注重 Java 语言与其他编程语言之间的互操作性并且能结合出新的 .NET 功能,例如 Microsoft ASP.NET 和 Microsoft ADO.NET 等。但是 Visual Studio.NET 对电脑硬件的要求比较高,如果是一般的家庭电脑,运行速度很慢,而且用 Visual J#.NET 所编写的程序必须在装有 Microsoft.NET Framework 的电脑上运行。所以 Visual J#.NET 更适合于编写基于 .NET 的应

用程序的计算机高级程序员。如果想了解更多的 Visual J#.NET 信息, 可以访问 <http://www.microsoft.com/china/msdn/vjsharp/>。

(2) JBuilder X 是 Borland 公司的 Java 语言开发工具 JBuilder 的最新版本。与之前的版本相比, JBuilder X 主要集中开发 Java 在网页应用、网络服务和数据库上的功能, 并且支持各种版本的计算机系统。这也从一个方面反映出当今 Java 发展的趋势。JBuilder X 与超过 80 个由第三方提供的可重复使用的组件和插件一起发布, 提供给 JBuilder 用户更多附加价值。从 Borland 公司的网站 <http://www.borland.com/jbuilder/> 上可以获得 JBuilder 的最新资料。

(3) JCreator 是由 Xinox Software 公司开发的。这家总部在荷兰的软件公司以编写高级语言开发工具而闻名。JCreator 对计算机系统要求不高, 比其他大多数具有集成开发环境的软件运行速度要快, 而且还具有允许使用者自定义操作窗口界面及无限 Undo/Redo 等功能。JCreator 是免费下载的, 可以到它的主页 <http://www.jcreator.com/> 上了解更多的信息。

(4) FreeJava 是一个免费的 Java 语言开发工具, 其界面如图 1-2 所示。其主要特点是可以快捷方便地查阅 Java 类库和函数、帮助编辑源程序、快速编译和运行 Java 程序、用不同颜色显示关键字, 以及双击编译错误提示可直接跳到相应的源程序处等。注意使用 FreeJava 之前必须要先安装 Java 2 JDK。

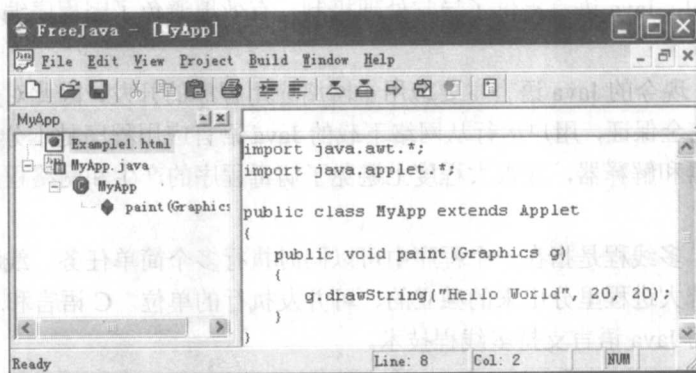


图 1-2 FreeJava 的界面

(5) Java 2 SDK (Software Development Kit) 是 SUN 公司编写的 Java 语言编译器。虽然它没有集成开发环境, 只能在 DOS 命令窗口下运行, 但是操作简单, 初学者非常容易掌握。SUN 公司根据不同的计算机系统和用户的不同需求, 开发了多种 Java 工具软件, 例如, 适用于个人的 J2SE (Java 2 Standard Edition) 和适用于企业的 J2EE (Java 2 Enterprise Edition), 相关信息可在公司网站 www.sun.com 上查看。本书将以 J2SE 的 Java 2 SDK 为开发工具来介绍 Java 语言。

1.2.2 安装 Java 2 SDK

1. 安装 Java 2 SDK

目前, SUN 公司出品的 Java 2 SDK 已经发展到 1.5.0 测试版; 与 1.4 版相比, 1.5 版新增加了一些实用的类, 并且重新编排了 Java 语言的 API。因为 1.5 版还只是测试版, 所以本书将使用 Java 2 SDK 1.4.2_05 来介绍 Java 语言。这两个版本的基础功能完全相同, 所以用户可

以自行决定使用哪个版本。两个版本都可以从 SUN 公司的网站上免费下载。

1.4.2 版的网址是：

<http://java.sun.com/j2se/1.4.2/download.html>

1.5.0 测试版的网址是：

<http://java.sun.com/j2se/1.5.0/download.jsp>

安装 Java 2 SDK 的操作步骤如下。

(1) 打开浏览器，在“地址”栏中键入上述的 1.4.2 版的网址，按 Enter 键打开网页，如图 1-3 所示。

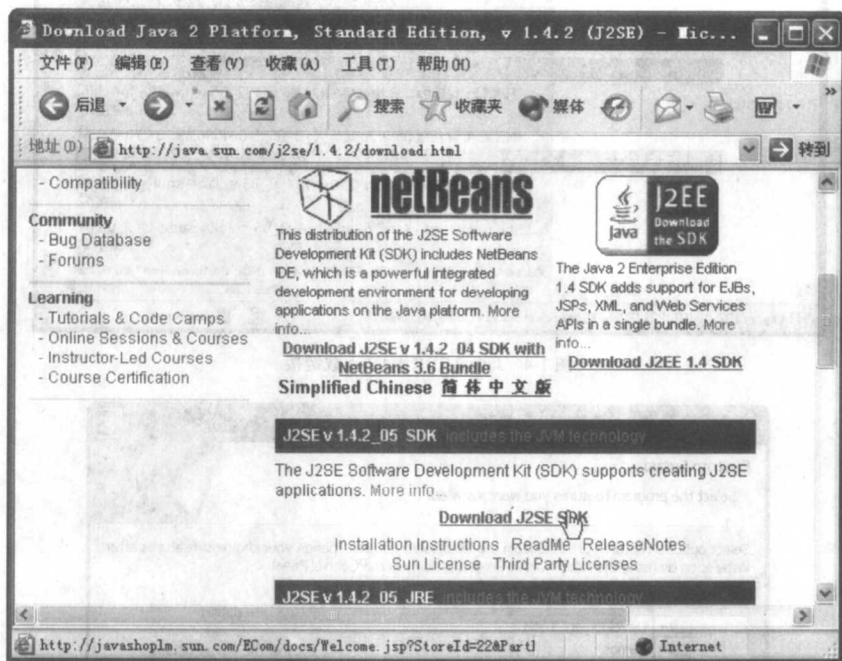


图 1-3 Java 2 SDK 1.4.2 官方下载网站

(2) 单击 J2SE v 1.4.2_05 SDK 栏中的“Download J2SE SDK”链接，打开 LICENSE AGREEMENT（许可协议）页面。选中“Accept”单选项，单击“Continue”按钮，打开 DOWNLOAD（下载）网页，如图 1-4 所示。Windows 系统使用的安装文件名称为 j2sdk-1_4_2_05-windows-i586-p.exe，其大小为 51.09MB。单击对应的链接，开始下载 Java 2 SDK 1.4.2。

(3) 下载完成后，双击 j2sdk-1_4_2_05-windows-i586-p.exe 文件图标，开始安装 Java 2 SDK。

(4) 稍等片刻，弹出“Java 2 SDK,SE v1.4.2.05-License”对话框，单击选中对话框下方的第一个单选项，表示接受协议。

(5) 单击“Next”按钮，弹出“Java 2 SDK,SE v1.4.2.05-Custom Setup”对话框，如图 1-5 所示。

(6) 在对话框中，设定安装的组件和安装软件的路径。如果要重新设定安装的组件，可以单击组件名前的黑色小三角，打开其下拉列表，选择是否安装。建议安装所有 Java 组件。如果要重新设定安装的位置，可以单击“Change”按钮，从弹出的对话框中选择文件的安装路径。我们可以在计算机硬盘的任何地方安装 Java2 SDK，但是对初学者来说最好使用默认

路径: C:\jdk1.4.2_05\。

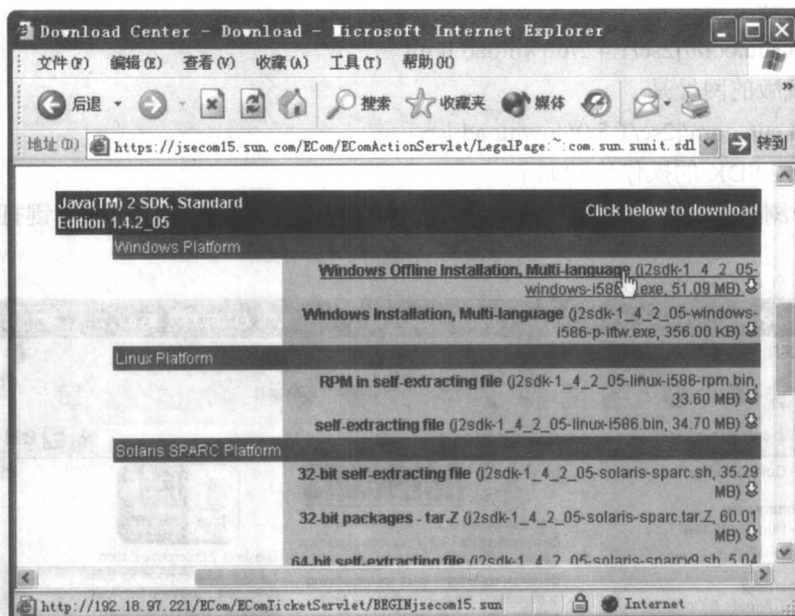


图 1-4 Java 2 SDK 1.4.2 下载链接

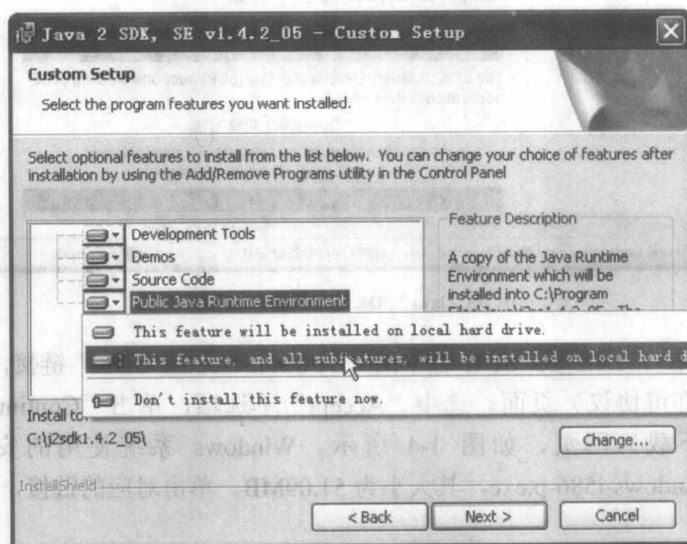


图 1-5 “Java 2 SDK,SE v1.4.2.05-Custom Setup”对话框

(7) 单击“Next”按钮,弹出“Java 2 SDK, SE v1.4.2_05-Browser Registration”对话框,如图 1-6 所示。该对话框用来选择使用 Java Plug-in 的浏览器。因为今后将在选定的浏览器中使用 Java Plug-in 来运行 Java Applet,所以建议选择所有列出的浏览器。

(8) 单击“Install”按钮,开始进行安装。

(9) 安装完成后,系统自动弹出“Java 2 SDK, SE v1.4.2_05-Complete”对话框。单击“Finish”按钮,关闭对话框,安装完成。

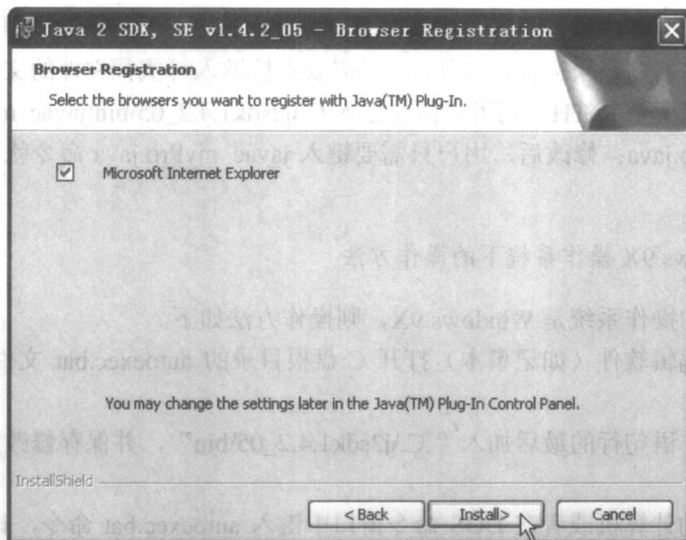


图 1-6 Java Plug-in 浏览器选择对话框

2. Bin 文件夹

安装好 Java 2 SDK 后, 打开 C:\j2sdk1.4.2_05\bin 文件夹, 其中有 20 多个以 .exe 为扩展名的可执行文件, 它们都是 Java 语言工具, 都是可以在 DOS 环境下执行的文件。其中常用的有以下几个文件。

(1) javac 文件。javac 文件是 Java 语言的编译器。在 DOS 环境下的“命令提示符”窗口中, 键入 javac, 然后再键入含有 Java 源程序的文件名, 可以用来编译该源程序, 生成相应的字节码文件。

(2) java 文件。java 文件是 Java 语言的解释器。在 DOS 环境下的“命令提示符”窗口中, 键入 java, 然后再键入编译好的字节码文件名, 可以用来运行程序, 输出结果。

(3) appletviewer 文件。appletviewer 文件是 Java Applet 的解释器。在 DOS 环境下的“命令提示符”窗口中, 键入 appletviewer, 再键入含有字节码文件的 HTML 程序名, 可以用来运行编译好的 Applet 程序。

(4) jar 文件。jar 文件是 Java 语言压缩工具。该工具可以将 Java 源程序打包成一个比原文件小的 jar 文件。

(5) javadoc 文件。javadoc 文件用于生成 API 文档。

(6) javah 文件。javah 文件用于从 Java 类中调用 C++ 语言代码。

(7) javap 文件。javap 文件用于分析字节码文件。

在本书中, 常使用的是前 3 个可执行文件。它们是运行程序的主要工具, 我们将在本章的实例 1 和实例 2 中具体介绍它们的使用方法。

1.2.3 更新系统环境变量

由于 Java 2 SDK 的编译和运行基本上都是在 DOS 环境下进行的, 因此, 为了能在任何提示符下都方便地直接使用 C:\j2sdk1.4.2_05\bin 文件夹中的可执行文件和 Java 类库, 我们需