

零点起航



—— 计算机编程系列教材

Java 2

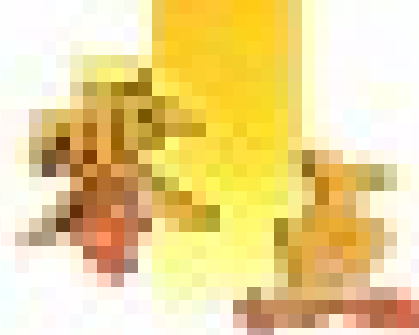
基础教程



©零点工作室 李忠伟 刘健 等编著



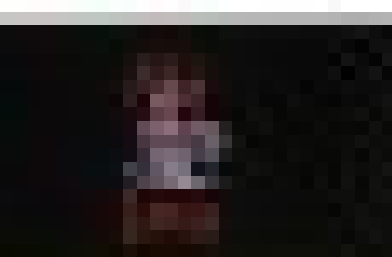
机械工业出版社
CHINA MACHINE PRESS



第 2 版

—— 程序员最佳实践 ——

Java 2



OFFICIAL JAVASE™ 2 SE EDITION



CHAPMAN & HALL

零点起航 —— 计算机编程系列教材

Java 2 基础教程

零点工作室 李忠伟 刘 健等编著



机械工业出版社

Java 是 Sun 公司开发的新一代编程语言，其面向对象、跨平台、分布式、安全性强等特点，使其成为目前最为瞩目和流行的编程语言。本书以最新版本的 JavaSDK1.4.2 为例，从基本的面向对象概念入手，通过大量的实例练习，系统地介绍了 Java 的语法、对象和类的概念、数组和字符串、异常处理、多线程机制、文件与输入输出流和网络应用程序设计方法。在内容的选取和章节的设置上充分考虑了初学者的实际需要，以形象化、易懂的语言讲述 Java 语言与面向对象设计与开发的概念。通过本书的学习，读者可以轻松掌握 Java 语言的基本知识和应用程序设计的基本方法。

本书内容全面，语言流畅，实例丰富，注重理论联系实际。本书可供从事 Java 应用程序设计与开发人员学习和参考，尤其适合作为大、中专院校和培训班的教材使用。

图书在版编目 (CIP) 数据

Java 2 基础教程/李忠伟等编著. —北京: 机械工业出版社, 2004.9

(零点起航——计算机编程系列教材)

ISBN 7-111-15282-4

I. J... II. 李... 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 095366 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 商红云

封面设计: 陈沛 责任印制: 施 红

煤炭工业出版社印刷厂印刷·新华书店北京发行所发行

2005 年 1 月第 1 版第 1 次印刷

787mm×1092mm 1/16·16.75 印张·410 千字

定价: 23.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话 (010) 68993821、88379646

封面无防伪标均为盗版

系列教材序言

随着计算机技术的飞速发展,各种应用软件也在社会和领域得到了广泛应用,软件的设计和开发也成为日常工作中的一部分。对于从事计算机方面工作的人员,掌握一种或几种编程工具是其必须具备的专业技能,也是胜任工作的基本条件。同时,很多高校学生、电脑爱好者也希望自己能够掌握基本的软件设计方法,以满足未来工作和学习的需要。

零点起航——计算机编程系列教材选择了目前常用的编程软件,涵盖了应用程序设计、数据库开发和网络编程等开发工具,着眼于大专院校教师、学生和工程技术人员自学和培训。整套教材由7本书组成:

- (1)《Visual Basic.NET 基础教程》
- (2)《Visual C++.NET 基础教程》
- (3)《PowerBuilder 9.0 基础教程》
- (4)《Java 2 基础教程》
- (5)《C++Builder 6.0 基础教程》
- (6)《Delphi 7 基础教程》
- (7)《Visual FoxPro 7.0 基础教程》

整套教材本着从零开始的思想,从基础培训的角度入手,在内容的选取和章节的设置上充分考虑了初学者的实际需要,力求简明清晰、通俗易懂。在详细讲解软件功能和用法的同时,引导读者练习一些针对性、实用性很强的程序实例,以加深对内容的理解。在每章的最后,都附带了一些习题,通过对这些习题的思考和练习,读者可以对该章所学内容有更加深刻的认识。该套教材的作者都是长期从事计算机教学和软件设计的高校教师和专业人员,具有丰富的理论和实践经验,在写作过程中融入了多年的经验和体会,为初学者提出了许多有益的建议。

零点起航——计算机编程系列教材中的各教材自成体系,读者可以根据自己的实际需要选择。

希望这套教材对您的学习、工作和生活有所帮助。

零点工作室网站: <http://www.zerobook.net>

主编邮箱: guandianzhu@qdcnc.com

零点工作室

2004年6月

前 言

Java 语言是一种优秀的面向对象的程序设计语言，具有平台无关性、安全机制、高可靠性和内嵌的网络支持等特点。由于 Java 语言的这些特点，使得 Java 语言成为当前网络应用程序编写的首选工具之一。

本书以最新版本的 JavaSDK1.4.2 为例，从基本的面向对象概念入手，通过大量的实例练习，系统地介绍了 Java 的语法、对象和类的概念、数组和字符串、异常、多线程机制、流式输入输出和文件以及网络应用程序设计。在内容的选取和章节的设置上充分考虑了初学者的实际需要，以形象化、易懂的语言讲述 Java 语言与面向对象设计与开发的概念。书中还列举了大量的实例，稍加修改后就可以直接应用于实际的开发工作。

全书共分 8 章，按照由易到难、循序渐进的规律安排内容。第 1 章介绍了 Java 的发展过程和特点，引出了两种 Java 程序，并简述了 Java 程序的结构和开发过程。第 2 章详细介绍 Java 语言编程的基础知识，包括一般程序设计语言的大部分内容，如基本数据类型、运算符、表达式和流程控制语句等。第 3 章详细地介绍面向对象的概念与技术，包括类、继承、接口和包等。第 4 章介绍了数组和字符串这一特殊的类的定义和使用。第 5 章介绍了 Java 中的异常类和异常处理方法。第 6 章介绍 Java 的多线程机制，包括线程的概念、多线程的实现方法、线程的生命周期、线程的控制、线程的异步和同步等。第 7 章介绍 Java 中的流式输入输出和文件处理。第 8 章介绍网络应用程序设计的基础知识，特别介绍了 Client/Server 模式的编程方法，结合多线程机制、流式输入输出，给出了该模式下的应用程序设计实例。

计算机语言的学习应该与编程实践有机地结合。读者除学习本书中讲解的语法规则与用法外，还应该注重编程实践，边学边练，这样才能更好地掌握 Java 语言程序设计的内容和方法。

本书内容详实，操作性强，在编写方法上遵循由浅入深、循序渐进、难点分散的原则，适合对 Java 语言和面向对象编程技术感兴趣的初、中级读者阅读，也可作为一般读者的自学用书。通过学习本书，可以使对面向对象编程技术和 Java 语言“一点都不懂”的读者，较快地掌握面向对象的概念和 Java 语言程序设计的基本知识和方法，获得 Java 应用程序开发的初步经验。

本书由李忠伟、刘健负责编写，参与本书编写的还有张福顺、徐俊恭、湛浩旻、汪学清、张书钦、戴朝晖、隋成城、张忠林、林新忠、宋一兵和唐际宇，在此向他们表示衷心的感谢。

编者
2004 年 5 月

目 录

系列教材序言

前言

第 1 章 Java 应用程序设计概述	1	4.4 小结	100
1.1 认识 Java	1	4.5 习题	101
1.2 Java 在 Windows 系统中的 安装与配置	5	第 5 章 异常处理	102
1.3 编译和运行 Java 应用程序	12	5.1 异常的概念	102
1.4 第一个 Java 应用程序	13	5.2 抛出异常	104
1.5 小结	18	5.3 捕获异常	106
1.6 习题	19	5.4 自定义异常	110
第 2 章 Java 基本语法	20	5.5 小结	113
2.1 语句与注释	20	5.6 习题	113
2.2 标识符及关键字	21	第 6 章 排序、查找和递归	114
2.3 基本数据类型	23	6.1 排序	114
2.4 运算符与表达式	26	6.2 查找	117
2.5 流程控制语句	29	6.3 递归	121
2.6 小结	37	6.4 小结	124
2.7 习题	38	6.5 习题	125
第 3 章 面向对象编程	39	第 7 章 多线程机制	126
3.1 基本概念	39	7.1 基本概念	126
3.2 创建类	40	7.2 创建线程	127
3.3 创建对象	43	7.3 线程的控制与调度	132
3.4 类的封装、继承和多态性	45	7.4 线程的同步机制	136
3.5 接口	60	7.5 小结	151
3.6 包	67	7.6 习题	151
3.7 小结	68	第 8 章 输入输出流与文件	152
3.8 习题	69	8.1 标准输入输出类	152
第 4 章 数组和字符串	70	8.2 流与输入输出类	152
4.1 数组	70	8.3 文件对象 File	153
4.2 字符串	77	8.4 字节输出流 OutputStream 类	158
4.3 StringBuffer 对象	99	8.5 字符输出流 Writer 类	165
		8.6 字节输入流 Input Stream 类	168

8.7 字符输入流 Reader 类.....	174	第 10 章 图形用户界面设计	221
8.8 随机访问文件	177	10.1 创建图形用户界面	221
8.9 小结	179	10.2 AWT 的事件模型	225
8.10 习题.....	179	10.3 界面布局.....	227
第 9 章 网络应用程序设计	181	10.4 对话框.....	229
9.1 基本概念	181	10.5 综合应用.....	230
9.2 URL 类	183	10.6 小结.....	238
9.3 InetAddress 类	186	10.7 习题.....	238
9.4 套接字 Socket 类	188	附录	239
9.5 数据报 UDP 类	195	附录 A Java 常用工具的使用	239
9.6 网络应用程序实例	199	附录 B Java 关键字表	243
9.7 小结	220	附录 C Java 部分类库	244
9.8 习题	220	参考文献	259

第 1 章 Java 应用程序设计概述

Java 语言自诞生之日起,就因其面向对象和可跨平台移植的特点成为程序开发的必然选择,特别是在网络应用程序开发方面,目前已成为主流应用程序开发语言之一。那么,Java 只是一种程序设计语言吗?实际上,Java 是一种程序设计平台,既是一种开发环境,又是一种应用环境。

本章将向读者介绍 Java 语言的发展与现状,以及使用 Java 设计应用程序的基本知识。

1.1 认识 Java

Java 是 1990 年 6 月由 Sun 公司开发的革命性编程语言,之所以用“革命性”来形容 Java,是因为用 Java 编写的软件克服了传统的软件往往与具体的实现环境有关的缺陷,它是一种跨平台的、适合与网络计算环境的、面向对象的程序设计语言。

在了解 Java 前,我们先来回顾程序设计语言的发展过程。

1.1.1 程序设计语言的发展过程

程序设计语言是程序设计人员用来设计程序并为计算机所理解的语言,程序设计人员通过一组语句的集合向计算机表达其意图,而计算机则接收、理解并执行这组语句的操作。

按照程序设计语言的发展历史,大致可划分为如下几种类型:

1. 面向机器的语言

最早的计算机程序与具体的硬件设备结合非常紧密,通常是针对某种特定类型的计算机或者硬件设备而编写的,因此被称为面向机器的程序,程序设计语言因此也被称为面向机器的语言。由面向机器的语言设计和开发出的程序,一般都能充分发挥硬件运算速度快的优势,使程序具有运行效率非常高的特点。

面向机器的程序设计语言主要是各种机器语言和汇编语言。它们在计算机应用技术的发展历史上起着重要的作用,但本身也存在着固有的缺陷:一是与人类的自然语言相差很大,程序的可读性很差,对程序的修改和扩展等操作带来很大的困难;二是它们通常是针对某种特定类型的计算机或者硬件设备,这就使得程序基本上是不可以被移植到其他的计算机或者硬件设备上,严重地限制、妨碍了面向机器语言在应用开发中的使用。

2. 面向过程的语言

随着计算机技术的飞速发展,面向机器的语言越来越成为软件开发的障碍,因此在 20 世纪 60 年代左右,新的面向过程的程序设计语言被开发并且广泛使用。面向过程的程序设计以数据结构和算法为核心。数据结构用于量化描述需要解决具体问题,算法则研究如何

用更有效的方法来组织解决问题的具体过程。

面向过程的程序设计语言主要有 C、BASIC、FORTRAN、PASCAL 等，它们与人类的自然语言相近，便于编程人员理解和书写，极大地改善了程序的可读性和可维护性。更重要的是，面向过程的语言着重于问题的求解过程而不依赖具体类型的计算机和硬件设备，因此，设计和开发出的应用程序具有较高的可移植性。

由于这些优点，面向过程的程序设计方法被提出后就得到了广泛的应用和认可，在软件技术的发展历史上占有重要的地位，目前仍在发挥着作用。但是，面向过程的程序设计语言也逐渐在应用过程中暴露出如管理和维护困难、可重用性低等缺点。

3. 面向对象的语言

面向对象的语言代表了一种全新的思维模式，这种模式能够方便、有效地实现以往方法所不能实现的软件扩展、软件管理和软件复用，使大规模软件的高效率、高质量的开发、维护和升级成为可能，大大地扩展了软件开发技术。

在 20 世纪 60 年代就提出了面向对象的方法，最早的面向对象软件是 1966 年开发的 Simula I，它首次提出模拟人类的思维方法，把数据和相关操作集成在一起。由于当时的硬件条件和方法本身并不成熟，导致了这种技术没有得到使用和推广。但随着软件危机的出现和面向过程的开发方法越来越多地暴露出其局限性，人们把目光重新转移到面向对象的方法上来。在 1980 年提出的 Smalltalk 语言已经是一种比较成熟、有效的面向对象的程序设计语言了。紧随其后又产生了多种面向对象的语言，其中，最有影响也是被广泛认可并应用的面向对象语言就是 C++。

C++ 语言在兼容当时最为流行的 C 语言的基础上，加入了面向对象的有关内容和规则。因其很多语法规则和 C 语言相近，所以很容易被广大的 C 语言程序设计人员所接受。同时，C++ 具有的面向对象的功能也简化了应用软件的开发、设计和维护过程，降低了其难度，为开发大型软件提供了很大的方便。C++ 的成功应用和广泛推广证明了面向对象技术的实力和发展前景。

面向对象的程序设计方法已经成为 20 世纪 90 年代中期以后最重要的程序设计方法，而 Java 正是以面向对象为基础的语言。本书以 Java 语言为载体来介绍面向对象程序设计的基本概念、思想和方法，并将这些思想、方法贯穿在整个对 Java 语言的学习过程中。

1.1.2 Java 的产生与发展

Java 语言是 Sun 公司于 1990 年开发出来的，当时 Green 项目小组的研究人员正在致力于为未来的智能设备开发出一种新的编程语言，该小组的成员 James Gosling 对 C++ 在执行过程中的表现非常不满，于是把自己封闭在办公室里编写了一种新的语言，并将其命名为 Oak，这个名称源于 Gosling 能从他的办公室窗户中向外看到的那棵橡树。后来，一位律师发现已经有另一个产品注册了这个名称，为了避免不必要的麻烦，就将其改为 Java。和 C++ 语言相比，Java 去除了其中为了兼容 C 语言而保留的非面向对象的内容，使得程序更加严谨、可靠、易懂。尤其是 Java 特有的“一次编写，处处运行”的跨平台优点，使其特别适合在网络应用程序开发中使用，成为一种极具潜力的面向对象的开发工具。

在 Java 语言开发完成之时，正赶上互联网的起步。1994 年，Sun 公司用 Java 编写了 HotJava，这是一个能够运行 Java 程序的 Web 浏览器，人们开始认识到 Java 的两项功能：

能向互联网提供什么和能够创建什么类型的程序。从此,越来越多的人将眼光投向 Java,并开始学习 Java。

Netscape 是第一个认可 Java 语言的公司,在 1995 年就将 Java 解释器集成到 Navigator 浏览器中。Microsoft 随后也在 Internet Explorer 中认可了 Java。这使得人们可以在他们的浏览器中首次运行交互的程序。

Java 语言的出现,引起了一场软件革命,这是因为传统的软件往往与具体的实现环境有关,换了一个环境就要做一番改动,既耗时又费力;而用 Java 语言编写的软件能运行在不同的机器上,只要所使用的机器安装了 Java 语言解释器即可。

Java 语言对未来软件的开发也产生了很大的影响,这可以从下面几个方面考虑:

- 软件开发方法: Java 语言具有面向对象的特性,符合最新的软件开发规范。
- 软件的需求分析: 动态、可视化地描述用户的需求分析,为设计者提供更直观的要求。
- Web 页面的动态设计: Java 语言的动画效果比 GUI 技术更加逼真,利用互联网提供的巨大动画资源,可以共享全世界的动态画面资源。
- 软件产品: 用 Java 语言开发的软件产品具有可视化、可听化、可操作化和可交互性的特点。

1.1.3 Java 语言的主要特点

Java 语言具有简单性、面向对象、平台无关性、分布性与安全性、高效性、多线程、动态执行等主要特点。

(1) 简单性: Java 语言的语法非常类似 C++, 熟悉 C++ 的程序设计人员会很快地掌握 Java。Java 去掉了 C++ 中模糊、复杂、不常用且容易出错的特性以及影响程序健壮性的地方,如指针、结构、运算符的重载、多重继承等,而且通过实现对无用单元的自动收集,大大简化了程序设计者的内存管理工作。

(2) 面向对象: 面向对象编程 (Object-Orient Programming, OOP) 是将计算机程序概念化成一组分离的对象,这些对象彼此之间可以进行交互。一个对象包含了信息的访问以及改变这些信息的方法,这使得程序更易于改进、理解并更具有重用性。Java 语言以面向对象为基础。在 Java 语言中,不能在类外面定义单独的数据和函数,所有对象都要派生于同一个基类 Object,并共享它所有的功能,也就是说,Java 语言最外部的数据类型是对象,所有的元素都要通过类和对象来访问。

(3) 平台无关: 所有使用 Java 的程序设计人员都知道这样一个口号:“一次编写,处处运行”。这一口号反映了 Java 的平台无关性,而 Java 也因为这一特点实现了程序设计与开发人员的梦想。Java 语言规定了统一的数据类型,这为 Java 程序跨平台的无缝移植提供了很大的便利。Java 编译器将 Java 程序编译成二进制代码,即字节码 (bytecode),字节码有统一的格式,不依赖于具体的硬件环境,在任何安装 Java 解释器的系统中,都可以执行这些代码。解释器针对不同的处理器指令系统,将字节码转换为不同的具体指令,保证了程序的“处处运行”。

(4) 分布性与安全性: Java 语言从诞生就和网络紧密地联系在一起。在 Java 中内置了 TCP/IP、HTTP 和 FTP 等协议类库,使得开发网络应用程序更为容易、便利。Java 程序的

三级代码安全检查机制可以有效地防止非法代码的侵入,阻止对内存的越权访问,避免计算机病毒的侵害。可以说,Java 是目前 Internet 上最安全的技术。

(5) 高效性:Java 程序经过编译器编译,生成字节码。字节码是与平台无关的二进制码,有其统一的格式,并且进行了优化设计,能够很容易地直接转换成对应于特定 CPU 的机器码或汇编码,因此运行速度很快。

(6) 多线程:Java 内置了语言级的多线程功能,使用户程序可以并行执行。Java 提供的同步机制保证了各线程对共享数据的正确操作。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,使程序具有更好的交互能力和实时运行能力。

(7) 动态执行:Java 语言中的类是根据需要而装入的,在类库中可以自由地加入新的方法和实例变量而不会影响用户程序的执行。并且 Java 语言通过接口来支持多重继承,比严格的类继承更灵活,也更具有扩展性。

此外,Java 为用户提供了详尽的 API 文档说明。Java 开发工具包中包罗万象的类库,使程序员可以在一个较高层次上进行开发,这也正是 Java 受欢迎的一个重要原因。

1.1.4 Java 的实现机制

1. Java 虚拟机

Java 虚拟机 (Java Virtual Machine, JVM) 是用软件模拟实现的虚拟计算机。Java 虚拟机定义了指令集、寄存器集、类文件结构栈、垃圾回收集等,提供了 Java 跨平台能力的基础框架。

Java 虚拟机执行过程有下面 3 个典型的特点:

- 多线程:Java 虚拟机支持多个线程同时运行。这些线程可独立执行 Java 代码并处理公共数据区和私有栈中的数据。
- 动态连接:Java 虚拟机具有动态连接的特性,这一特性使得 Java 程序很适合在网络上运行。
- 异常处理:Java 虚拟机提供了可靠的异常处理机制,这使得 Java 程序更安全、可靠。

2. 垃圾回收机制

Java 系统不仅要分配对象要使用的内存资源,还要跟踪资源的使用情况,定期检测到不再使用的内存资源,由系统自动回收后做再一次的分配,这就是 Java 中的垃圾回收机制。和 C++ 语言相比,程序员在使用 Java 语言开发程序时不用再考虑对象的释放问题,同时,还提高了程序的安全性,避免因资源消耗而造成的系统隐患。

3. 代码安全性检测

Applet 程序可以将远程代码下载到 Web 浏览器上运行,改变了传统模式下程序的运行方式,扩展了 Web 浏览器的功能,但同时也带来了安全隐患。为此,必须建立行之有效的安全模型,进行代码安全性检测,通过对一些操作的限制来增强网络的安全性,如不能对本地文件进行访问,不能建立新的网络连接等。

Java 2 采用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策略来设定可访问的资源域。这一策略可有效地解决区分本地代码和远程代码所带来的困难。

1.1.5 Java 语言的几个版本

到目前为止, Sun 公司已经推出了 Java 语言的很多版本:

- Java1.0: 最早的 Java 版本, 主要应用于网页。
- Java1.1: 在原有的版本基础上, 改进了用户界面, 完全重写了事件处理, 并加入了 JavaBeans 组件技术。
- 带有 SDK1.2 的 Java 2: 是 Java1.1 版本的扩展, 重组了图形化的用户界面和数据库访问, 对很多方面做了改进。
- 带有 SDK1.3 的 Java 2: 添加了新的特性, 例如可以更好地对多媒体进行访问, 编译速度更快。
- 带有 SDK1.4 的 Java 2: 在功能上更好地完善了以前的版本, 具有运行速度更快、界面更友好、对非专业程序人员来说使用更容易的特点。该版本还具有按钮、菜单栏和其他一些配合 Windows XP 或 Linux 的 Gnome 界面的图形元素。此外, 它提供了新的控制面板、自动升级功能, 并修复了 1.4.0 版本以来的超过 2400 程序错误。

本书要介绍的是 SDK1.4.2 版本的 Java 2, 这是目前最新的 Java 2 的 SDK 版本, 该版本中还加入了 NetBeans 组件的集成开发环境, 使得在该环境下开发 Java 应用程序更方便、快捷。

1.2 Java 在 Windows 系统中的安装与配置

Sun 公司为每个新的 Java 版本都提供了一个免费的开发工具集 (SDK, Soft Development Kit), 本书使用 Java 2 的一个新版本 1.4.2 来介绍 Java 应用程序的开发, 在以后的章节中, 我们简称为 J2SDK1.4.2。该版在原来跨平台性和高安全性的基础上, 增加了很多新的特性和功能, 提供了更高的性能、可规划性、稳定性和服务能力, 特别是发展了丰富的客户端应用程序开发能力, 为开发和发布当今标准化通用的 Web 服务提供了基础。新的版本还有众多的改进, 如: 和 Window XP 或者 Gnome 类似的界面; 新的图形化控制面板; 自动更新; 一些处理任务速度的提高, 能够更高效地运行在 Intel 的 Itanium 处理器上, 还支持了 Intel 的 SSE 和 SSE2 指令集。

在 Sun 的 Java 网站 <http://java.sun.com> 上可以免费获得 Java 的开发工具集, 使用任何一个字处理软件配合它就可以开发 Java 应用程序, 如 Windows 系统提供的记事本。还有很多商业开发工具, 如 Borland JBuilder、IBM Visual Age for Java 和 Sun Forte for Java, 读者可以在掌握了本书介绍的 Java 应用程序开发基础知识后, 根据需要选择合适的开发工具。

注意: 本书所有的例子都是在 Windows 系统下使用记事本编写, 并在 J2SDK1.4.2 测试通过的。

1.2.1 J2SDK1.4.2 的安装

J2SDK1.4.2 可以安装在 Solaris Operating System、Microsoft Windows 和 Linux 平台下, Sun 公司为不同的平台提供了不同的安装程序。要安装 J2SDK1.4.2, 我们必须从 Sun 的 Java

网站上下载到安装程序，或者从 J2SDK1.4.2 的光盘中运行安装程序。在安装前，还应该确认系统中是否已经安装过其他的 Java 开发工具，如果已经安装了其他的 Java 开发工具，有可能在对 J2SDK1.4.2 进行配置时出现一些问题。

J2SDK1.4.2 的安装向导将指导我们完成整个软件的安装，整个安装过程大约要花费 30 分钟左右，主要经历以下 7 个步骤。

(1) 将光盘放入光驱，安装程序就会自动运行（或者在“我的电脑”或“资源管理器”找到安装程序所在的位置，用鼠标双击安装程序），将出现如图 1-1 所示的安装准备界面。安装程序在收集了必须的系统信息后，会提示用户开始安装，如图 1-2 所示。

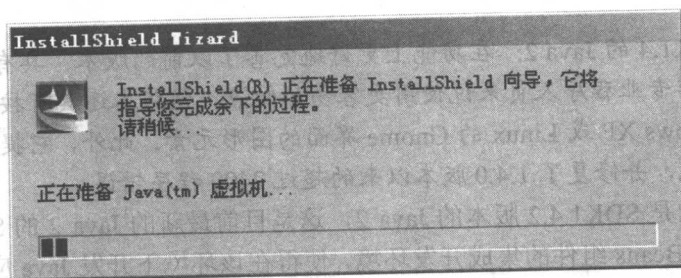


图 1-1 安装准备界面

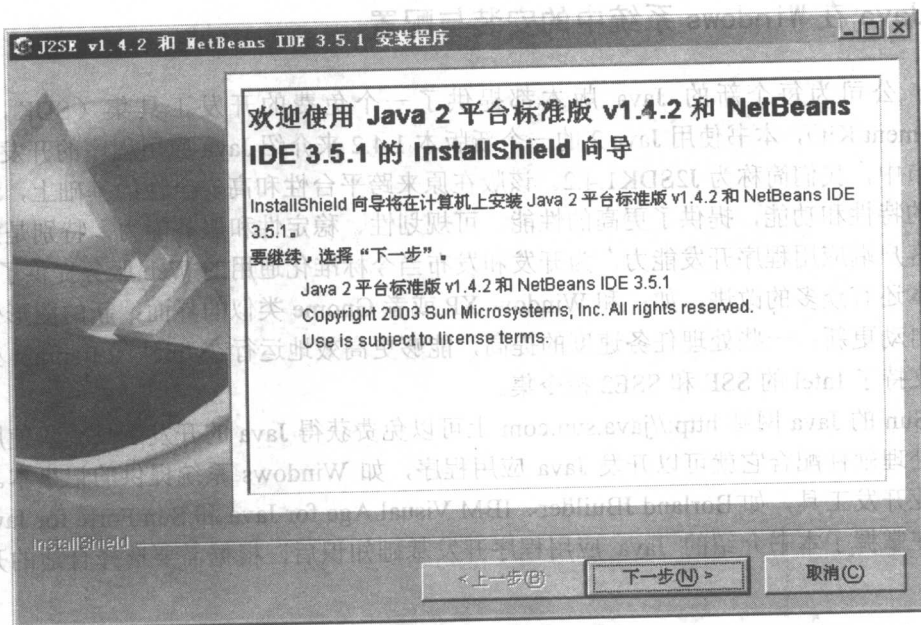


图 1-2 开始安装 J2SDK1.4.2

(2) 单击 **下一步(N) >** 按钮，安装向导会提示我们接受安装协议，如图 1-3 所示。只有我们接受该协议，继续单击 **下一步(N) >** 按钮，安装过程才能继续。

(3) 接着会出现如图 1-4 所示的画面，要求我们选择安装目录。我们可以单击 **浏览...** 按钮选择其他的目录进行安装。一般情况下使用向导提供的默认安装目录。

(4) 单击 **下一步(N) >** 按钮, 向导提示我们是否要将此集成开发环境与 Java 源文件和 NetBeans 模块关联, 如图 1-5 所示。是否关联对我们以后的应用程序设计没有太大的影响, 因此, 在这里我们选择不关联。

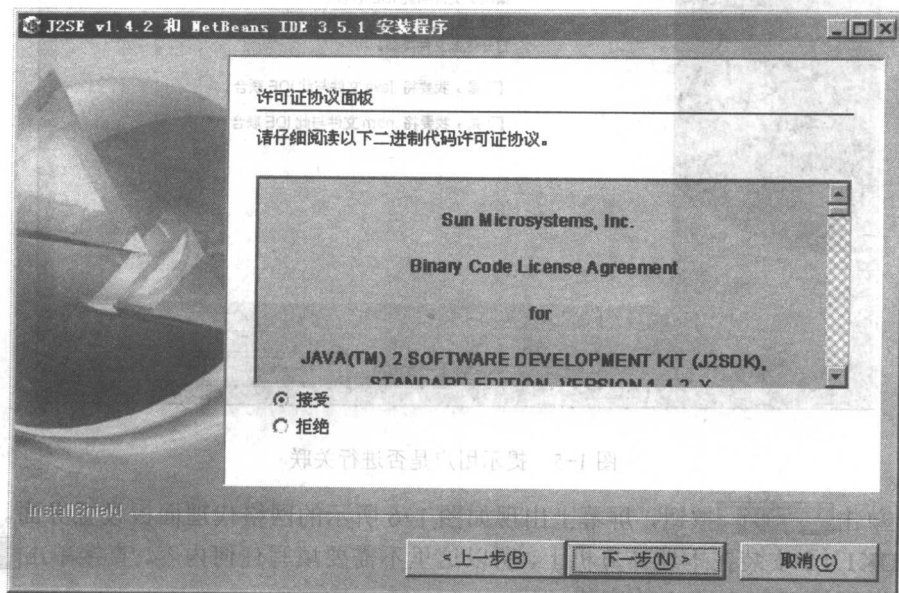


图 1-3 提示接受安装协议

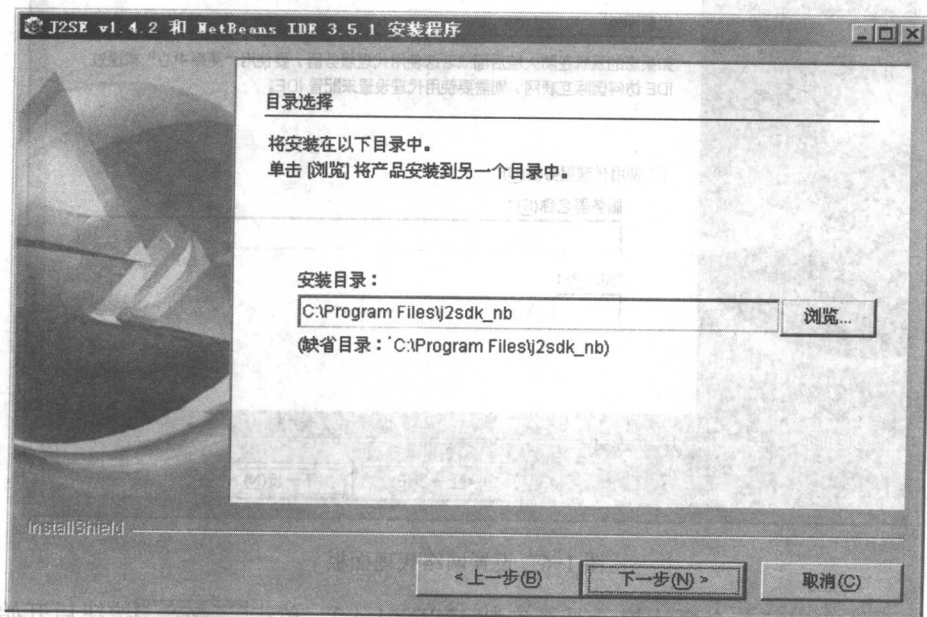


图 1-4 选择 J2SDK1.4.2 的安装目录

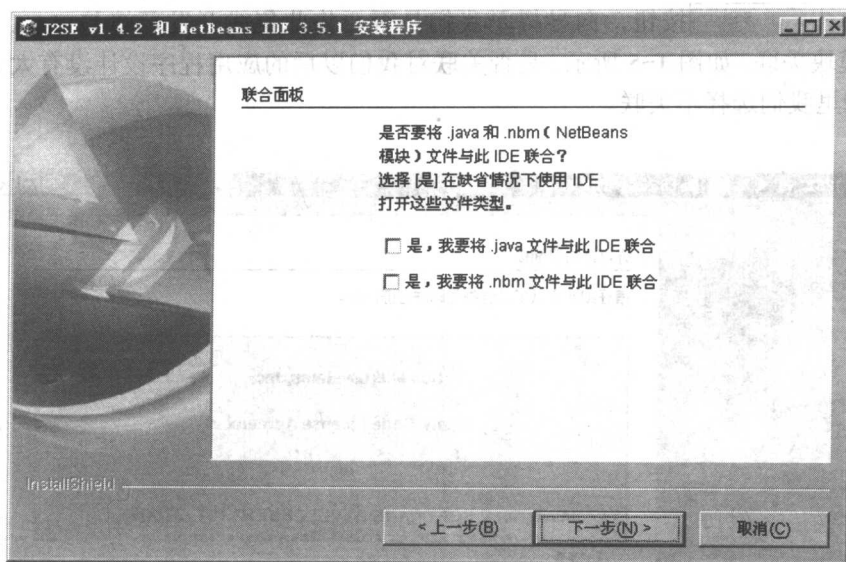


图 1-5 提示用户是否进行关联

(5) 单击 **下一步(N) >** 按钮, 屏幕上出现如图 1-6 所示的网络代理面板设置界面。因为我们将 J2SDK1.4.2 安装在本地计算机上, 所以这里不需要填写任何内容, 直接单击 **下一步(N) >** 按钮就可以开始 J2SDK1.4.2 的安装。

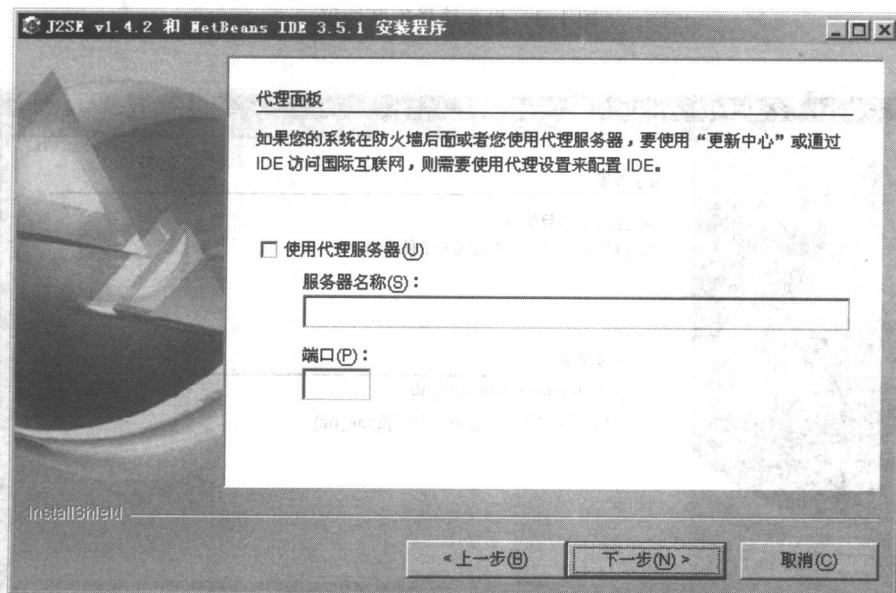


图 1-6 设置网络代理面板

(6) 系统在安装前会再次提示是否安装 J2SDK1.4.2, 单击 **确定** 按钮后开始安装, 安装界面如图 1-7 所示。整个安装过程要花费几分钟到十几分钟的时间, 时间的长短取决于我们使用的计算机的配置。

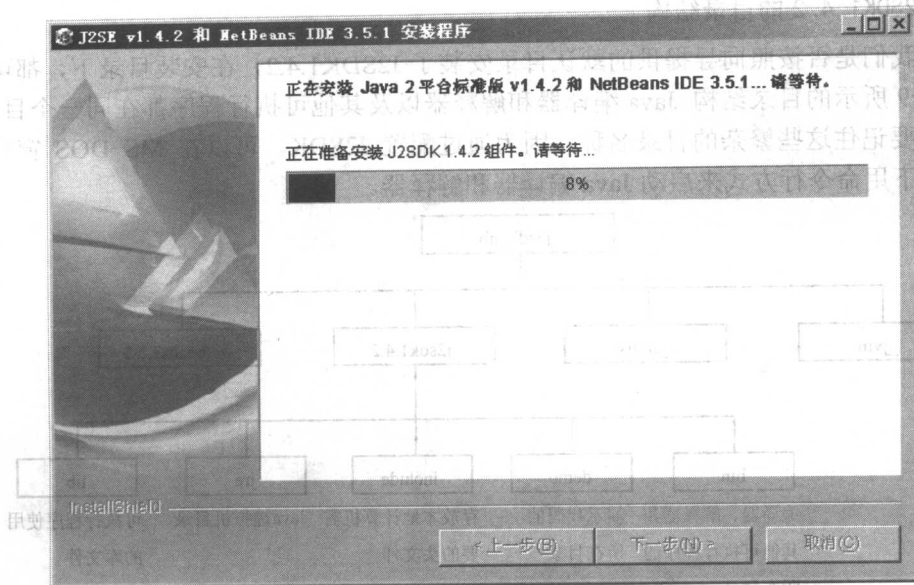


图 1-7 J2SDK1.4.2 开始安装

(7) 安装完成后, 安装向导将提示我们单击 **完成(F)** 按钮结束整个安装过程, 如图 1-8 所示。

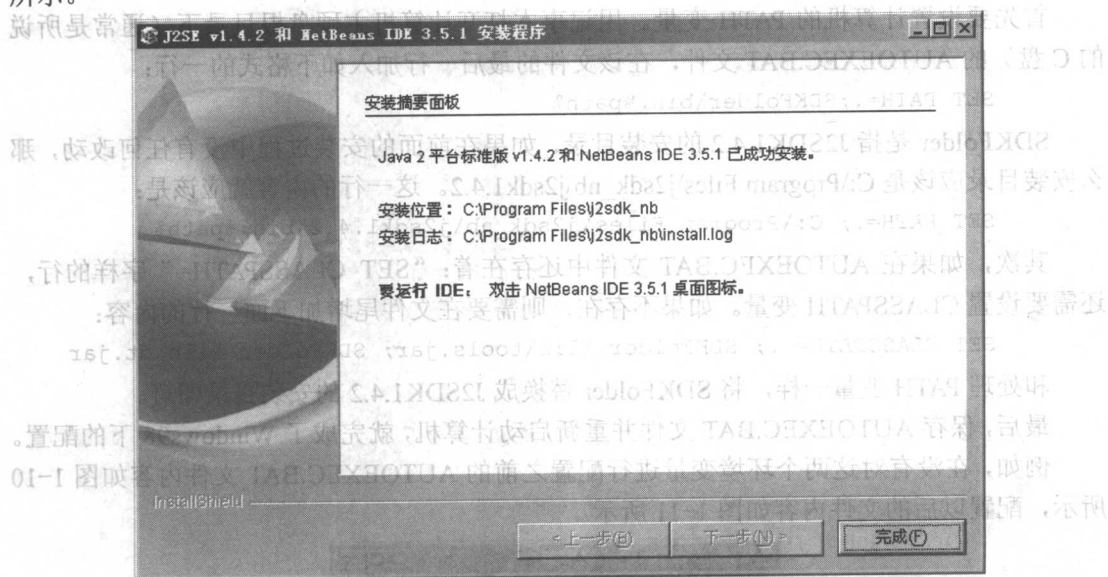


图 1-8 J2SDK1.4.2 安装完成

在安装完 J2SDK1.4.2 后, 还需要调整计算机的环境变量才能开始 Java 应用程序的开发。下面的章节中, 我们就来认识 J2SDK1.4.2 的目录结构, 并调整两个重要的环境变量。

注意: 只要把 J2SDK1.4.2 安装在向导默认的文件夹里就更容易修正可能在以后使用中遇到的任何配置问题。