



课堂实录

孙宇霞 郑千忠 / 编著

Java 开发 课堂实录

- 构思独特，所有案例来自一线实战场景；
- 实用性强，将抽象的理论结合到实战案例上；
- 内容全面，结构清晰，体例丰富；
- 视频教学，专业教学视频帮助读者快速上手。

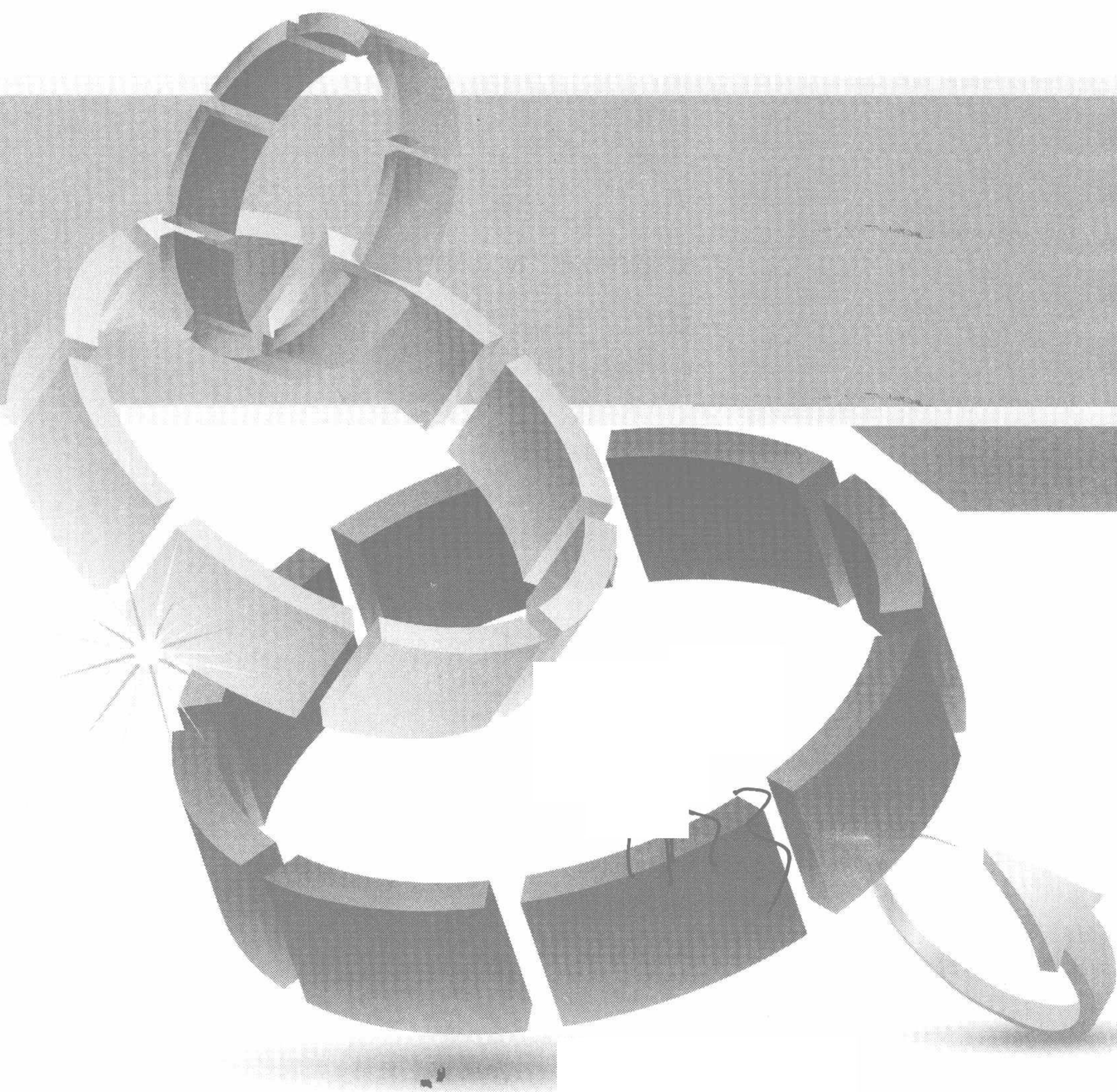


清华大学出版社

课堂实录

孙宇霞 郑干忠 / 编著

Java 开发 课堂实录



清华大学出版社
北京

内 容 简 介

本书结合教学的特点编写，将 Java 软件开发的技術以课程的形式讲解。全书共分 14 课，通过通俗易懂的语言详细介绍了 Java 编程基础知识。本书内容从简单的 Java 元素、数据类型开始，深入讲解 Java 的方法、类和面向对象的高级特征，并且介绍了 Java 中的常用工具类、异常处理、I/O 流、集合框架、图形用户界面和数据库编程等相关知识。最后通过一个成绩管理系统来介绍 Java 编程在实际开发中的应用。

本书可作为在校大学生使用 Java 进行课程设计的参考资料，也可作为非计算机专业学生学习 Java 语言的参考书。

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 开发课堂实录 / 孙宇霞, 郑千忠编著. —北京: 清华大学出版社, 2016
(课堂实录)

ISBN 978-7-302-40315-9

I. ①J… II. ①孙… ②郑… III. ①JAVA 语言-程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2015) 第 113332 号

责任编辑: 夏兆彦
封面设计: 张 阳
责任校对: 徐俊伟
责任印制: 李红英

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

印 刷 者: 北京鑫丰华彩印有限公司

装 订 者: 三河市溧源装订厂

经 销: 全国新华书店

开 本: 190mm×260mm 印 张: 23.5 字 数: 667 千字

版 次: 2016 年 2 月第 1 版 印 次: 2016 年 2 月第 1 次印刷

印 数: 1~3000

定 价: 49.00 元

产品编号: 051594-01

前言

Java 是当今较流行的一种简单的、面向对象的、分布式的、健壮安全的、结构中立的、可移植及性能优异的多线程动态语言。该语言由 Sun 公司于 1995 年推出，发展到今天，已不仅仅表示一门程序语言，更是一种软件开发平台，目前已经演化出了 J2SE、J2EE 和 J2ME 3 个版本。凭借其易学易用和功能强大的特点，可以进行桌面应用、Web 应用、分布式系统及嵌入式系统等应用程序开发，并且在信息技术、科学研究、军事工业、航空航天等领域应用广泛。

本书内容

全书共分为 14 课，主要内容如下。

第 1 课 Java 语言概述。本课首先介绍 Java 语言的发展史和特点，然后介绍 Java 程序的工作原理及环境配置，最后介绍了 Java 程序的开发步骤，包括编写、编译、运行和调试程序等。

第 2 课 简单数据类型及运算。本课首先介绍 Java 语言中的基本数据类型，然后介绍 Java 语言中的变量和常量，最后介绍运算符与表达式之间的整合使用。

第 3 课 流程控制语句。本课主要介绍 Java 提供的流程控制语句，包括空语句、语句块、选择 if 语句、switch 语句、do 语句、for 语句、break 语句以及异常处理语句等。

第 4 课 类与对象。本课首先介绍类的概念，然后介绍面向对象的核心特征，类成员的应用以及修饰符，包括 private、static 和 final。

第 5 课 深入面向对象编程。本课介绍抽象类、接口、内部类和匿名类在 Java 中的使用，方法的重载和重写，super 关键字及包的使用等。

第 6 课 数组与集合。本课主要介绍 Java 中数组和集合的使用，包括声明和初始化数组、数组的排序、集合框架及泛型等。

第 7 课 异常。本课首先介绍异常类型的概念及其分类，然后介绍了 Java 异常处理机制，最后介绍了如何抛出异常、声明异常及如何自定义异常。

第 8 课 线程。本课针对 Java 中的线程应用进行讲解，包括线程的概念、实现一个线程、线程的生命周期、调度和同步线程等。

第 9 课 Java 常用类。本课主要介绍了 Java 语言中常用的工具类，包括 Object 类、包装类 (Integer 类和 Character 类)、字符串类、日期类和 Random 类。

第 10 课 Java 的输入输出流。本课首先介绍了 Java 中的输入流和输出流，然后讲解使用字节流、字符流写入/读取文件，最后介绍了如何使用 File 类中的方法来获取文件的相关信息，比如文件名称、文件大小、文件内容长度等。

第 11 课 图形用户界面应用。本课首先介绍 AWT 中容器的使用，然后介绍 5 种布局管理器的应用、基本组件的创建、事件处理机制以及 Swing 和 Applet。

第 12 课 Java 数据库编程。本课主要介绍 JDBC 连接数据库的相关知识，包括 JDBC 的工作原理、Java 程序连接数据库的基本步骤和 JDBC 的应用，包括增加记录、编辑记录和使用事务等。

第 13 课 Java 的网络编程。本课详细介绍 Java 中的 TCP 编程、URL 编程和 UDP 编程。

第 14 课 成绩管理系统。本课主要介绍使用 Java 结合 SQL Server 数据库创建成绩管理系统的过程，主要功能包括学生信息模块、课程信息模块和成绩信息模块。

本书特色

本书针对初、中级用户量身订做，以课堂课程学习的方式，由浅入深地讲解 Java 语言的应用，同时根据语法特性，突出了开发时的重要知识点，并配以案例讲解。

□ 结构独特

全书以课程为学习单元，每课安排基础知识讲解、实例应用、拓展训练和课后练习 4 个部分讲解 Java 的编程知识。

□ 知识点全

本书紧紧围绕 Java 程序开发展开讲解，具有很强的逻辑性和系统性。

□ 实例丰富

书中各实例均经过作者精心设计和挑选，它们都是根据作者在实际开发中的经验总结而来，涵盖了在实际开发中所遇到的各种场景。

□ 应用广泛

对于精选案例，步骤详细具体，结构清晰简明，分析深入浅出，而且有些程序能够直接在项目中使用，避免读者进行二次开发。

□ 基于理论，注重实践

本书在讲述过程中不仅仅只介绍理论知识，而且在合适位置安排综合应用实例，或者小型应用程序，将理论应用到实践当中，来加强读者实际应用的能力，巩固开发基础和知识。

□ 视频教学

本书为实例配备了视频教学文件，读者可以通过视频文件更加直观地学习 Java 的知识。所有视频教学文件均已上传到 www.ztydata.com.cn，读者可自行下载。

□ 网站技术支持

读者在学习或者工作的过程中，如果遇到实际问题，可以直接登录 www.itzcn.com 与我们取得联系，作者会在第一时间给予帮助。

读者对象

本书适合作为软件开发入门者的自学用书，也适合作为高等院校相关专业的教学参考书，也可供开发人员查阅、参考。

□ Java 软件开发入门者。

□ Java 初学者以及在校学生。

□ 各大中专院校的在校学生和相关授课老师。

□ 准备从事软件开发的人员。

除了封面署名人员之外，参与本书编写的人员还有李海庆、王咏梅、康显丽、王黎、汤莉、倪宝童、赵俊昌、方宁、郭晓俊、杨宁宁、王健、连彩霞、丁国庆、牛红惠、石磊、王慧、李卫平、张丽莉、王丹花、王超英、王新伟等。本书在编写中难免会有疏漏与不妥之处，欢迎读者通过清华大学出版社网站（www.tup.tsinghua.edu.cn）与我们联系，帮助我们加以改正与提高。

目录

第 1 课 Java 语言概述

- | | | | |
|--------------------------------------|----|----------------|----|
| 1.1 基础知识讲解 | 2 | 1.2 实例应用：实现累加器 | 20 |
| 1.1.1 Java 语言简介 | 2 | 1.2.1 实例目标 | 20 |
| 1.1.2 Java 程序运行环境 | 5 | 1.2.2 技术分析 | 20 |
| 1.1.3 简单的 Java 程序 | 9 | 1.2.3 实现步骤 | 20 |
| 1.1.4 Java 程序的基本点 | 15 | 1.3 拓展训练 | 22 |
| 1.1.5 Java Application 和 Java Applet | 18 | 1.4 课后练习 | 22 |

第 2 课 简单数据类型及运算

- | | | | |
|----------------|----|------------|----|
| 2.1 基础知识讲解 | 26 | 2.2.1 实例目标 | 41 |
| 2.1.1 变量与常量 | 26 | 2.2.2 技术分析 | 41 |
| 2.1.2 基本数据类型 | 29 | 2.2.3 实现步骤 | 41 |
| 2.1.3 运算符与表达式 | 33 | 2.3 拓展训练 | 42 |
| 2.2 实例应用：实现计算器 | 41 | 2.4 课后练习 | 43 |

第 3 课 流程控制语句

- | | | | |
|----------------|----|------------|----|
| 3.1 基础知识讲解 | 46 | 3.2.1 实例目标 | 61 |
| 3.1.1 选择语句 | 46 | 3.2.2 技术分析 | 62 |
| 3.1.2 嵌套 if 语句 | 51 | 3.2.3 实现步骤 | 62 |
| 3.1.3 循环语句 | 53 | 3.3 拓展训练 | 63 |
| 3.1.5 跳转语句 | 59 | 3.4 课后练习 | 64 |
| 3.2 实例应用：判断闰年 | 61 | | |

第 4 课 类与对象

- | | | | |
|-----------------|----|------------|----|
| 4.1 基础知识讲解 | 68 | 4.2.1 实例目标 | 93 |
| 4.1.1 类概述 | 68 | 4.2.2 技术分析 | 93 |
| 4.1.2 面向对象和对象 | 70 | 4.2.3 实现步骤 | 93 |
| 4.1.3 类成员 | 78 | 4.3 拓展训练 | 95 |
| 4.1.4 常用修饰符 | 86 | 4.4 课后练习 | 96 |
| 4.2 实例应用：模拟网上购物 | 93 | | |

第 5 课 深入面向对象编程

- | | | | |
|-----------------|-----|-------------------|-----|
| 5.1 基础知识讲解 | 100 | 5.1.5 包的概念 | 113 |
| 5.1.1 抽象类与接口 | 100 | 5.2 实例应用：模拟公司奖励制度 | 115 |
| 5.1.2 内部类和匿名类 | 104 | 5.2.1 实例目标 | 115 |
| 5.1.3 方法的重载和重写 | 108 | 5.2.2 技术分析 | 115 |
| 5.1.4 super 关键字 | 111 | 5.2.3 实现步骤 | 116 |

5.3 拓展训练	118	5.4 课后练习	118
----------------	-----	----------------	-----

第 6 课 数组与集合

6.1 基础知识讲解	122	6.2.2 技术分析	145
6.1.1 数组	122	6.2.3 实现步骤	145
6.1.2 集合	132	6.3 拓展训练	148
6.2 实例应用：使用集合模拟百度贴吧 ..	145	6.4 课后练习	149
6.2.1 实例目标	145		

第 7 课 异常

7.1 基础知识讲解	152	7.2.1 实例目标	164
7.1.1 异常概述	152	7.2.2 技术分析	164
7.1.2 异常分类	153	7.2.3 实现步骤	164
7.1.3 Java 异常处理	154	7.3 拓展训练	166
7.2 实例应用：优化错误提示信息	164	7.4 课后练习	168

第 8 课 线程

8.1 基础知识讲解	170	8.2 实例应用：实现进度条	183
8.1.1 线程基本概念	170	8.2.1 实例目标	183
8.1.2 线程实现	170	8.2.2 技术分析	183
8.1.3 线程的生命周期	173	8.2.3 实现步骤	183
8.1.4 线程的调度	174	8.3 拓展训练	185
8.1.5 线程同步	179	8.4 课后练习	185

第 9 课 Java 常用类

9.1 基础知识讲解	188	签到查询	212
9.1.1 Object 类	188	9.2.1 实例目标	212
9.1.2 包装类	188	9.2.2 技术分析	212
9.1.3 字符串	194	9.2.3 实现步骤	213
9.1.4 日期类	203	9.3 拓展训练	216
9.1.5 Random 类	211	9.4 课后练习	216
9.2 实例应用：模拟实现 QQ 空间			

第 10 课 Java 的输入输出流

10.1 基础知识讲解	222	加新内容	246
10.1.1 输入输出流概述	222	10.2.1 实例目标	246
10.1.2 字节输入输出流	223	10.2.2 技术分析	246
10.1.3 字符输入输出流	234	10.2.3 实现步骤	246
10.1.4 文件	241	10.3 拓展训练	247
10.2 实例应用：读取指定文本，并在末尾添		10.4 课后练习	248

第 11 课 图形用户界面应用

11.1 基础知识讲解	252	11.1.2 容器	253
11.1.1 AWT 概述	252	11.1.3 常用布局管理器	257

11.1.4 AWT 基本组件	264	11.2.1 实例目标	281
11.1.5 事件处理机制	273	11.2.2 技术分析	282
11.1.6 Swing 简介	277	11.2.3 实现步骤	282
11.1.7 Applet	278	11.3 拓展训练	289
11.2 实例应用：制作简易记事本	281	11.4 课后练习	290

第 12 课 Java 数据库编程

12.1 基础知识讲解	294	12.1.7 批处理	316
12.1.1 JDBC 简介	294	12.2 实例应用：学生信息管理	318
12.1.2 安装 JDBC 驱动	296	12.2.1 实例目标	318
12.1.3 连接数据库	298	12.2.2 技术分析	318
12.1.4 访问数据库	301	12.2.3 实现步骤	318
12.1.5 ResultSet 相关操作	310	12.3 拓展训练	322
12.1.6 事务处理	314	12.4 课后练习	323

第 13 课 Java 的网络编程

13.1 基础知识讲解	326	13.2.1 实例目标	338
13.1.1 网络程序设计基础	326	13.2.2 技术分析	338
13.1.2 TCP 编程	328	13.2.3 实现步骤	338
13.1.3 URL 编程	333	13.3 拓展训练	340
13.1.4 UDP 编程	334	13.4 课后练习	341
13.2 实例应用：年龄判断	338		

第 14 课 成绩管理系统

14.1 系统分析	344	14.3.1 数据访问层	347
14.1.1 需求分析	344	14.3.2 业务处理层	348
14.1.2 系统设计	344	14.4 成绩管理模块设计	350
14.2 数据库分析与设计	345	14.4.1 学生信息模块	350
14.2.1 创建数据表	345	14.4.2 课程信息模块	357
14.2.2 创建数据库	346	14.4.3 成绩信息模块	361
14.3 公共模块设计	346		

习题答案

第 1 课

Java 语言概述

在互联网高速发展的背景下，Java 语言正在以它独特的优势迅猛发展着。Java 语言是一种可以撰写跨平台应用程序的面向对象程序设计语言。它是由 Sun 公司于 1995 年推出的 Java 程序设计语言和 Java 平台（包括 JavaSE、JavaME 和 JavaEE）的总称。它有简单、面向对象、跨平台等特点，广泛应用于信息技术、科研、军事、航空航天等领域，是当今 IT 行业重要的一门编程语言。

本课将对 Java 语言的发展历程、特点、运行环境等几个方面进行介绍，使读者对 Java 语言有一个基础的了解。

本课学习目标：

- 了解 Java 语言的发展
- 掌握 Java 语言的特点
- 理解 Java 语言的工作原理
- 掌握如何搭建 Java 的运行环境
- 学会简单例子的编写
- 掌握简单的调试技巧

1.1

基础知识讲解

1.1.1 Java 语言简介

作为当今高级语言中十分引人瞩目的语言，Java 语言所具备的优点吸引着广大的编程人员。Java 是一种跨平台的编程语言，所谓跨平台也就是常说的“一次编写，到处执行”，另外，它还有很多其他的特点。为了使读者对 Java 语言有初步的了解，下面主要从 Java 语言的发展、特点和工作原理等方面进行一些简单介绍。

1.1.1.1 Java 语言的发展

在 Java 的发展历程中，互联网的高速发展无疑对其起到了至关重要的推动作用。Java 语言诞生于 1995 年的美国，当时的 Sun 公司正在开发家电消费类产品相关的应用程序，但是发现现有的编程语言很难解决系统的跨平台问题，为了解决这个问题，一种新的语言就诞生了，它就是 Oak，这也是 Java 语言的雏形。在 Oak 语言中，鉴于安全的考虑包含了 C 语言的一些语法，并且是面向对象的。虽然 Oak 语言的开发使 Sun MircoSystems 公司的项目的开发得以进行，但是在后来的市场上并没有得到很好的推广。

1995 年，Sun Microsystems 公司正式推出 Java 语言，互联网应用的迅猛发展带来了 Java 语言的春天。互联设备之间的差异性以及用户对于较好的人机交互的需求都是 Java 可以解决的问题，它具有跨平台、面向对象、简单以及适用于网络的特点，因而使用 Java 语言开发的 HotJava 浏览器得到广泛的应用，后来不少的编程人员开始尝试使用 Java 来编写应用程序，自此，Java 语言拥有了强劲的发展势头。

如今的 Java 一词不仅仅代表一种语言，它也代表基于 Java 的开发平台。目前的 Java 包含 3 个不同的版本，它们分别是 J2ME、J2EE 和 J2SE (JavaME、JavaEE 和 JavaSE 的简称)。在本书中主要介绍的是 J2SE，也就 Java 的标准版。

1. Java 2 Platform 和 Micro Edition (J2ME)

J2ME 是 Java 的微缩版，是所有领域版本中最小的一版，其主要用于小型数字设备上的应用程序的开发，如手机和 PDA 等。

2. Java 2 Platform 和 Enterprise Edition (J2EE)

J2EE 是 Java 的企业版，通常用于开发多层结构、分布式、Web 形式的企业级应用程序，其定义了一系列用于企业开发的类，如 EJB 和 JSP 等。

3. Java 2 Platform 和 Standard Edition (J2SE)

J2SE 是 Java 的标准版，也是 Java 其他版本的基础，其主要用于桌面应用程序的开发，包含了构成 Java 语言核心的类，如面向对象和数据库连接等。

1.1.1.2 Java 语言的特点

Java 语言的风格很像 C 语言和 C++ 语言，它是一种纯粹的面向对象的语言，它继承了 C++ 语言面向对象的技术的核心，但是也摒弃了 C++ 的一些缺点，比如说容易引起错误的指针以及多继承等，同时增加了垃圾回收机制，释放掉不被使用的内存空间，解决了程序员管理内存空间的烦恼。

Java 语言是一种分布式的面向对象语言，具有面向对象、简单、健壮、多线程、安全等很多特点，下面针对这些特点进行逐一介绍。



1. 面向对象

Java 是一种面向对象的语言，它对面向对象中的类、对象、继承、封装、多态、接口、包等均有良好的支持。为了简单起见，Java 只支持类之间的单继承，但是它使用接口来实现多继承。使用 Java 语言开发程序，需要采用面向对象的思想设计程序和编写代码。

2. 平台无关性

平台无关性的具体表现就在于，Java 是“一次编写，到处运行 (Write Once, Run Any Where)”的语言，因此采用 Java 语言编写的程序具有很好的可移植性，而保证这一点的正是 Java 的虚拟机机制。在引入虚拟机之后，Java 语言在不同的平台上运行不需要重新编译。Java 语言使用 Java 虚拟机机制屏蔽了具体平台的相关信息，使得 Java 语言编译程序只需生成虚拟机上的目标代码，就可以在多种平台上不加修改地运行。

3. 简单性

Java 语言的语法与 C 语言和 C++ 语言很相近，使得很多程序员学起来很容易。对于 Java 来说，它舍弃了很多 C++ 中难以理解的特性，如操作符的重载和多继承等，而且 Java 语言不使用指针，并且加入了垃圾回收机制，解决了程序员需要管理内存的问题，使编程变得更加简单。

4. 解释执行

Java 程序在 Java 平台运行时会被编译成字节码文件，然后就可以在搭建过 Java 环境的操作系统上运行。在运行时 Java 的解释器对这些字节码进行解释执行，执行过程中需要加入的类在链接阶段被载入到运行环境中。

5. 多线程

Java 语言是多线程的，这也是 Java 语言的一大特性，它必须由 Thread 类和它的子类来创建。Java 支持多个线程同时执行，并提供多线程之间的同步机制。任何一个线程都有自己的 run() 方法，而要执行的方法就写在 run() 方法体内。

6. 分布式

Java 语言支持 Internet 应用的开发，在 Java 的基本应用编程接口中就有一个网络应用编程接口，它提供了网络应用编程的类库，包括 URL、URLConnection、Socket 等。Java 的 RMI 机制也是开发分布式应用的重要手段。

7. 健壮性

Java 的强类型机制、异常处理、垃圾回收机制等都是 Java 健壮性的重要表现。对指针的丢弃是 Java 的一大进步。另外 Java 的异常机制也是健壮性的一大体现。

8. 高性能

Java 的高性能主要是相对其他高级的脚本语言来说的，随着 JIT (Just In Time) 的发展，Java 的运行速度也越来越高。

9. 安全性

Java 通常被用在网络环境中，为此，Java 提供了一个安全机制以防止恶意代码的攻击。除了 Java 语言具有许多的安全特性以外，Java 对通过网络下载类具有一个安全防范机制，以分配不同的名字空间以防替代本地的同名类，并包含安全管理机制。

Java 语言的一系列特性使其在众多的编程语言中占有较大的市场份额。Java 语言对对象的支持和强大的 API 使得编程工作变得更加容易和快捷，大大降低了程序的开发成本。Java 的“编译一次，到处执行”的特点也是它吸引众多的商家和编程人员的原因之一。

1.1.1.3 工作原理

作为 Java 语言实现与平台无关性的一大利器，Java 虚拟机是 Java 程序运行中不可或缺的一部分，而 Java 的垃圾回收机制，也是 Java 吸引广大编程人员的一个原因。而作为 Java 程序使用

时安全性的保证，Java 的代码安全性检查也是非常重要的。下面将通过三个方面来介绍 Java 的工作原理。

1. Java 虚拟机 (JVM)

使用 Java 虚拟机是实现 Java 语言平台无关性的保证。一般的高级语言如果要在不同的平台上运行，至少需要编译成不同的目标代码。而 Java 虚拟机 (Java Virtual Machine, JVM) 是软件模拟出来的计算机，它可以在任何处理器上安全兼容地执行保存在.class 文件中的字节码。Java 虚拟机也包含有自己完善的硬件架构，如处理器、堆栈、寄存器等，还具有相应的指令系统。Java 虚拟机在计算机中的位置如图 1-1 所示。

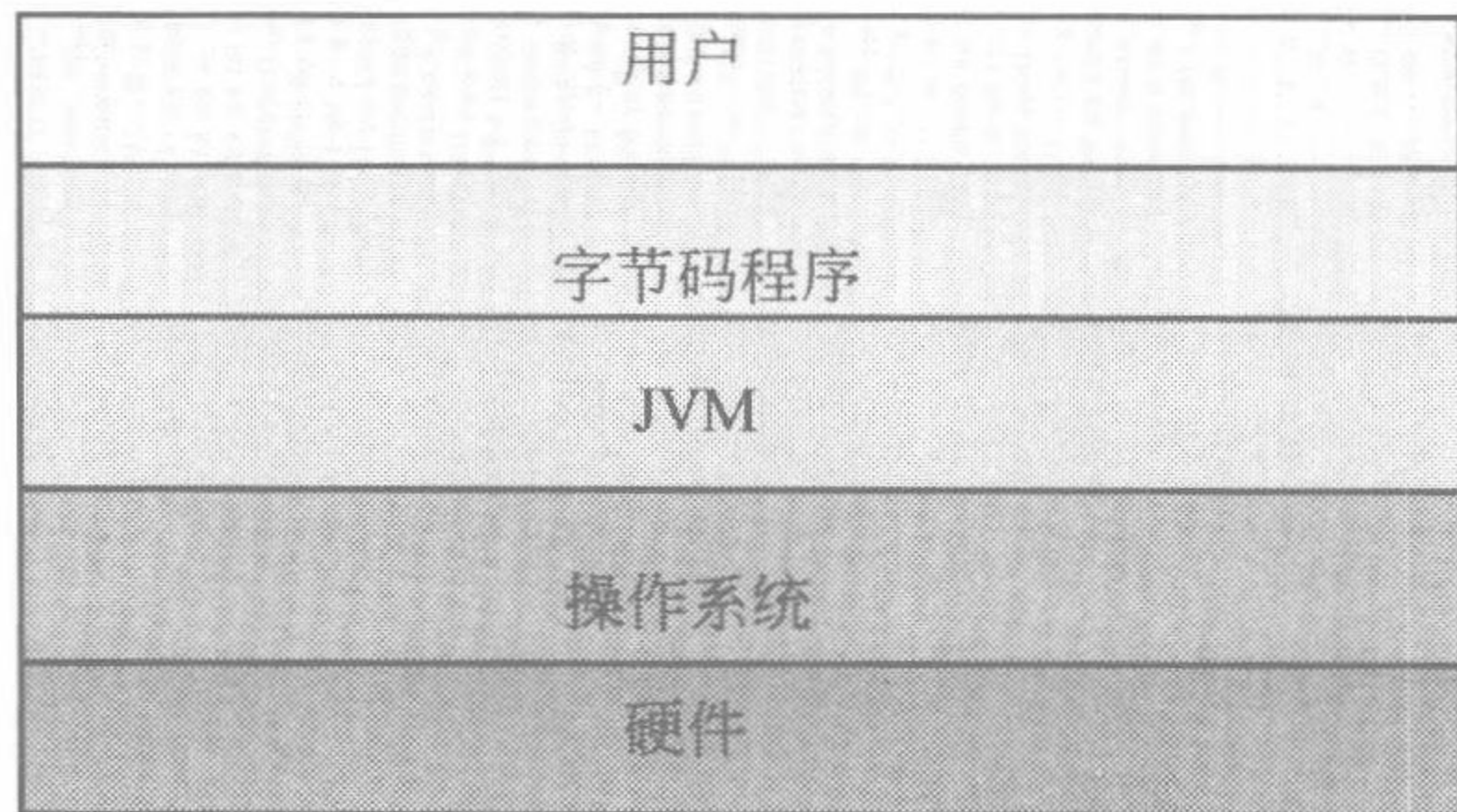


图 1-1 Java 虚拟机在计算机中的位置

Java 语言的跨平台特性主要是指字节码文件可以在任何具有 Java 虚拟机的计算机或者是电子设备上运行，Java 虚拟机上的解释器负责将字节码文件解释成相应的机器语言进行执行，因此 Java 源程序需要编译器编译为.class 文件。Java 程序的执行过程中，首先 Java 编译程序将 Java 源程序编译为 JVM 可执行代码——字节码，这一编译过程同 C/C++ 的编译有些不同，Java 编译器不仅对变量和方法的引用编译为数值引用，也不确定程序执行过程中的内存布局。它将这些符号引用信息保留在字节码中，由解释器在运行过程中创建内存布局，然后再通过查表来确定一个方法的所在地址，这样就有效地保证了 Java 的可移植性和安全性。

Java 程序的执行过程如图 1-2 所示。



图 1-2 Java 程序的执行过程

在将 Java 程序编译为字节码文件之后，字节码的运行要经历 3 个阶段。第 1 个阶段是使用类加载器 (class loader) 将类文件 (.class 文件) 加载到 Java 虚拟机中，在这个阶段需要检验该类文件是否符合类文件规范；第 2 个阶段是字节码检验器 (bytecode verifier) 检查该类文件代码中是否存在某些非法操作；第 3 个阶段，如果字节码检验器检查通过，由 Java 解释器负责把该类文件解释为机器码进行执行。Java 虚拟机采用的是“沙箱”运行模式，即把 Java 程序的代码和数据都限制在一定的内存空间里执行，不允许程序访问该内存空间外的内存，如果是 Applet 程序，还不允许访问客户端机器的文件系统。如图 1-3 是 Java 虚拟机解释执行的过程。

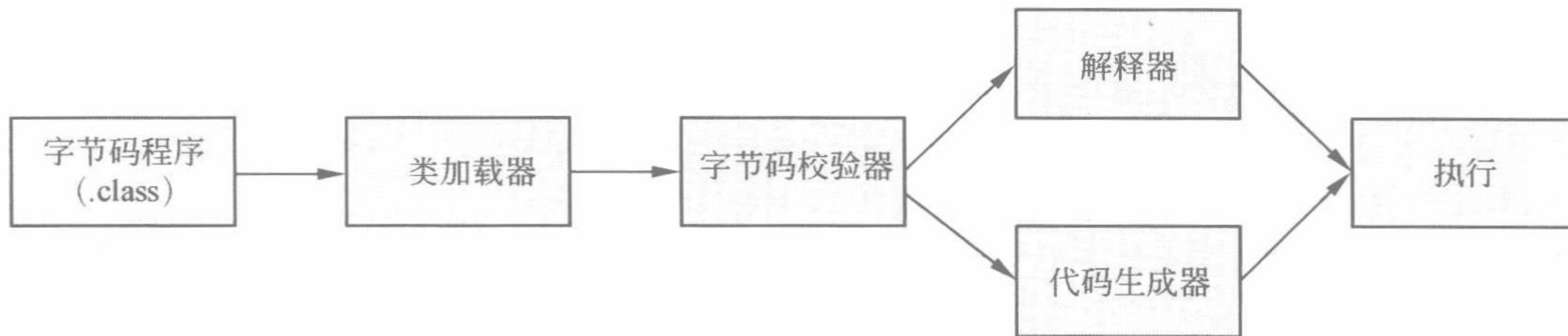


图 1-3 Java 虚拟机解释执行的过程

2. 垃圾回收机制

在程序的执行过程中，部分的内存存在使用过后就处于废弃状态，如果不及时进行回收，就有可

能会导致内存泄露，进而引发系统崩溃。在 C++ 语言中，编程人员在编写程序期间需要把不再使用的内存空间释放出来，但是这种人为的管理往往会因为编程人员的疏忽或者是经验不足等问题造成内存没有回收的状况出现，同时这也给程序员编程的进度产生一定的影响。而在 Java 运行环境中，始终有一个系统级的线程，专门跟踪内存的使用情况，每过一段时间检查是否有不再使用的内存，如果有就进行自动的回收。所谓自动回收并不是直接进行销毁，而是先将其加入到待回收列表中，这一点确实在很大程度上减轻了编程人员的工作量。

3. 代码安全性检查机制

Java 编程语言的出现，使得客户端的计算机可以方便地从网络上进行上传和下载 Java 程序，但是如何保证该 Java 程序的安全性呢？很简单，Java 语言通过 Applet 程序来控制非法程序的安全性。

1.1.2 Java 程序运行环境

运行环境是使用一门语言的前提条件，在下面的小节中会对 Java 运行环境的搭建进行详细的介绍。

1.1.2.1 JDK 的下载和安装

JDK (Java Development Kit) 是 Sun 公司推出的产品，在 JDK 开发工具包中包含了 Java 运行环境 (Java Runtime Environment)、Java 工具和 Java 基础类库。JDK 是整个 Java 的核心，随着 JDK 版本的不断升级，也增加了很多新的功能，运行效率也得到了很大的提高。

在 JDK 中包含了一些基础组件，用来完成不同的功能，如下所示。

- **javac.exe** Java 编译器，将源程序转换成字节码。
- **java.exe** Java 解释器，运行编译后的 Java 程序 (.class 后缀的)。
- **jar.exe** 打包工具，将相关的类文件打包成一个文件。
- **jdb.exe** 调试工具。
- **javadoc.exe** 文档生成器，从源代码注释中提取文档。
- **appletviewer.exe** 小程序浏览器，一种运行 HTML 文件上 Java 小程序的 Java 浏览器。
- **javah.exe** 产生可以调试 Java 过程的 C 过程，或建立能被 Java 程序调用的 C 程序的头文件。
- **javap.exe** Java 反汇编器，显示编译类文件中的可访问功能和数据，同时显示字节码的含义。
- **jconsole.exe** Java 进行系统调试和监控的工具。

要搭建 Java 的运行环境，首先要进行 JDK (Java Development Kits) 的安装。安装之前肯定要先进行下载。对于 JDK 来说，随着时间的推移，JDK 的版本也在不断升级，目前 JDK 的最新版本是 JDK 1.7。由于 Sun 公司在 2010 年被 Oracle (甲骨文) 公司收购，所以要到 Oracle 官方网站 (<http://www.oracle.com/technetwork/java/index.html>) 去下载最新版本的 JDK。其下载和安装过程的主要步骤如下。

(1) 打开 Oracle 官方网站，单击右上角的 Software Downloads 栏目下的 J2SE，进入到新的界面，单击“Java Platform (JDK) 7u10”上的图标进入新的界面，单击单选按钮【Accept License Agreement】，然后单击超链接“jdk-7u10-windows-i586.exe”进入下载界面，将文件下载到硬盘的某个位置。

(2) 安装 JDK 1.7，双击“jdk-7u10-windows-i586”文件，弹出安装对话框，如图 1-4 所示。

(3) 单击【下一步】按钮，进入“自定义安装”对话框，如图 1-5 所示。图中显示有 3 个可选功能，分别是【开发工具】、【源代码】和【公共 JRE】，默认全选。

(4) JDK 的默认安装路径是“C:\Program File\Java\jdk1.7.0_10”，如果不想更改安装路径可以直接单击【下一步】按钮；如果想更改安装路径，单击【更改】按钮进行路径更改，更改完成后

单击【下一步】按钮开始安装，直到出现图 1-6 说明软件安装成功，单击【关闭】按钮完成安装。

(5) 安装完成后，会在“C:\Program File\Java”的目录下会产生一个名为“jdk1.7.0_10”的文件夹，文件夹中的内容如图 1-7 所示。

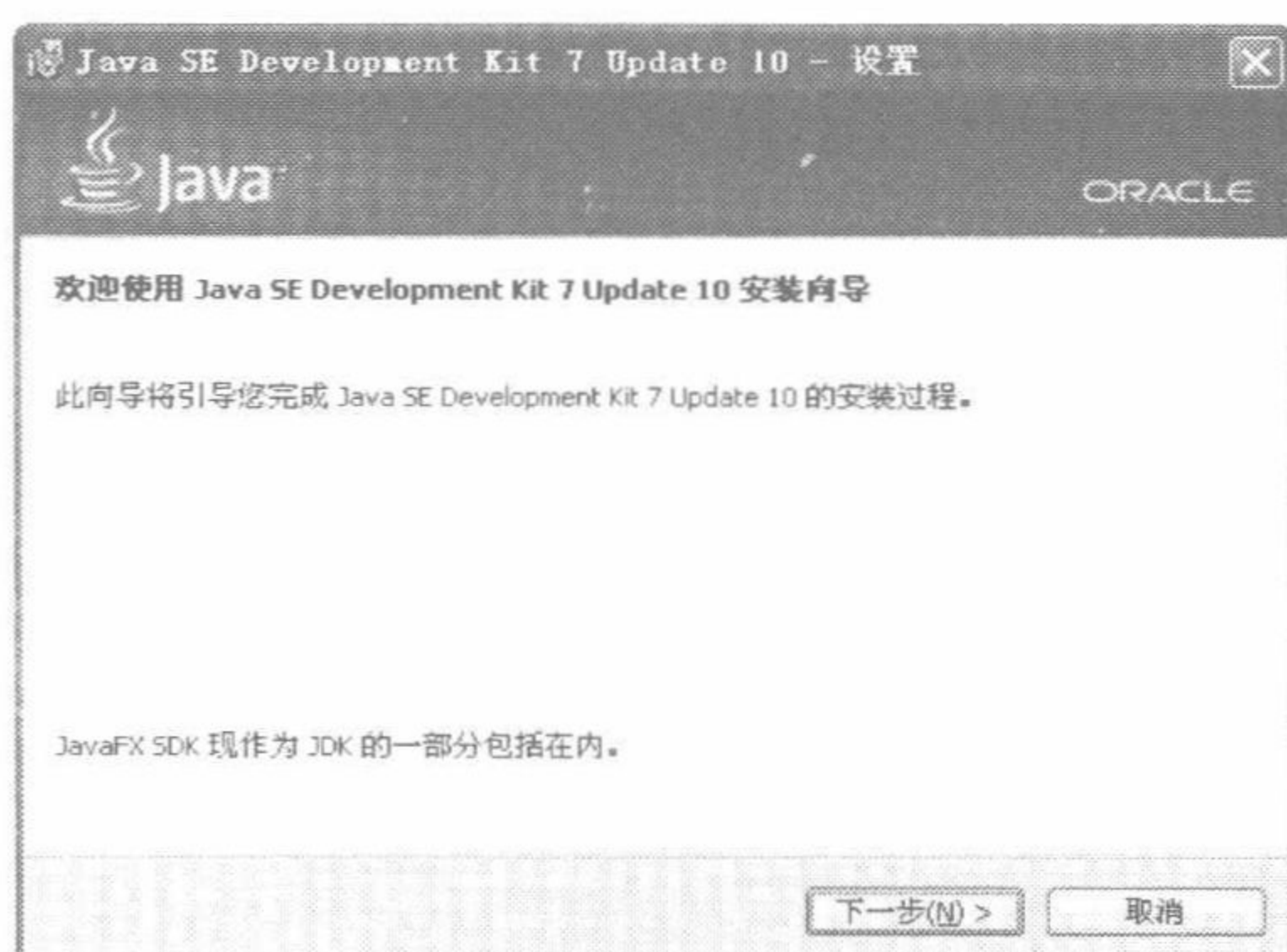


图 1-4 JDK 安装图 1

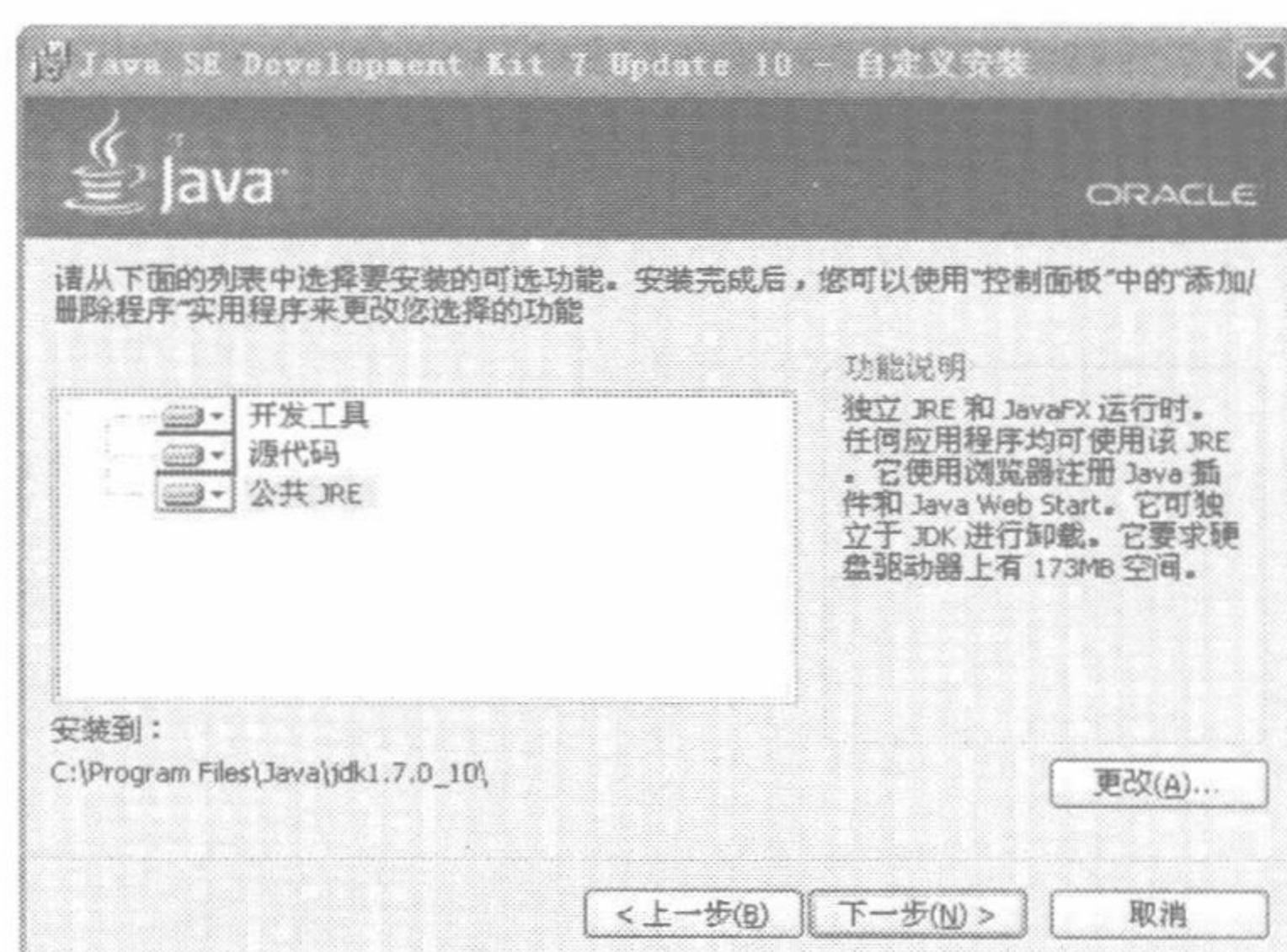


图 1-5 JDK 安装图 2

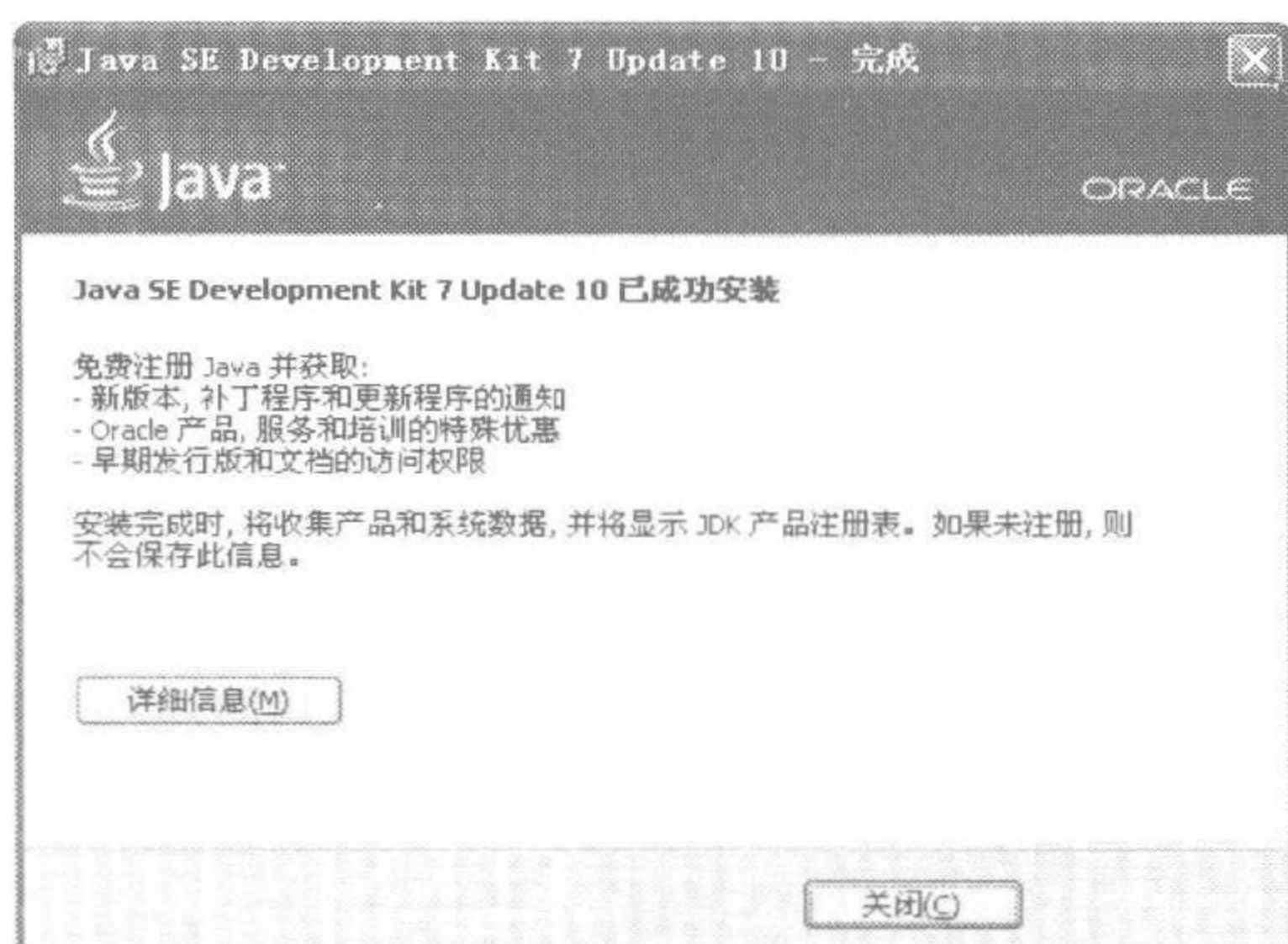


图 1-6 JDK 安装图 3

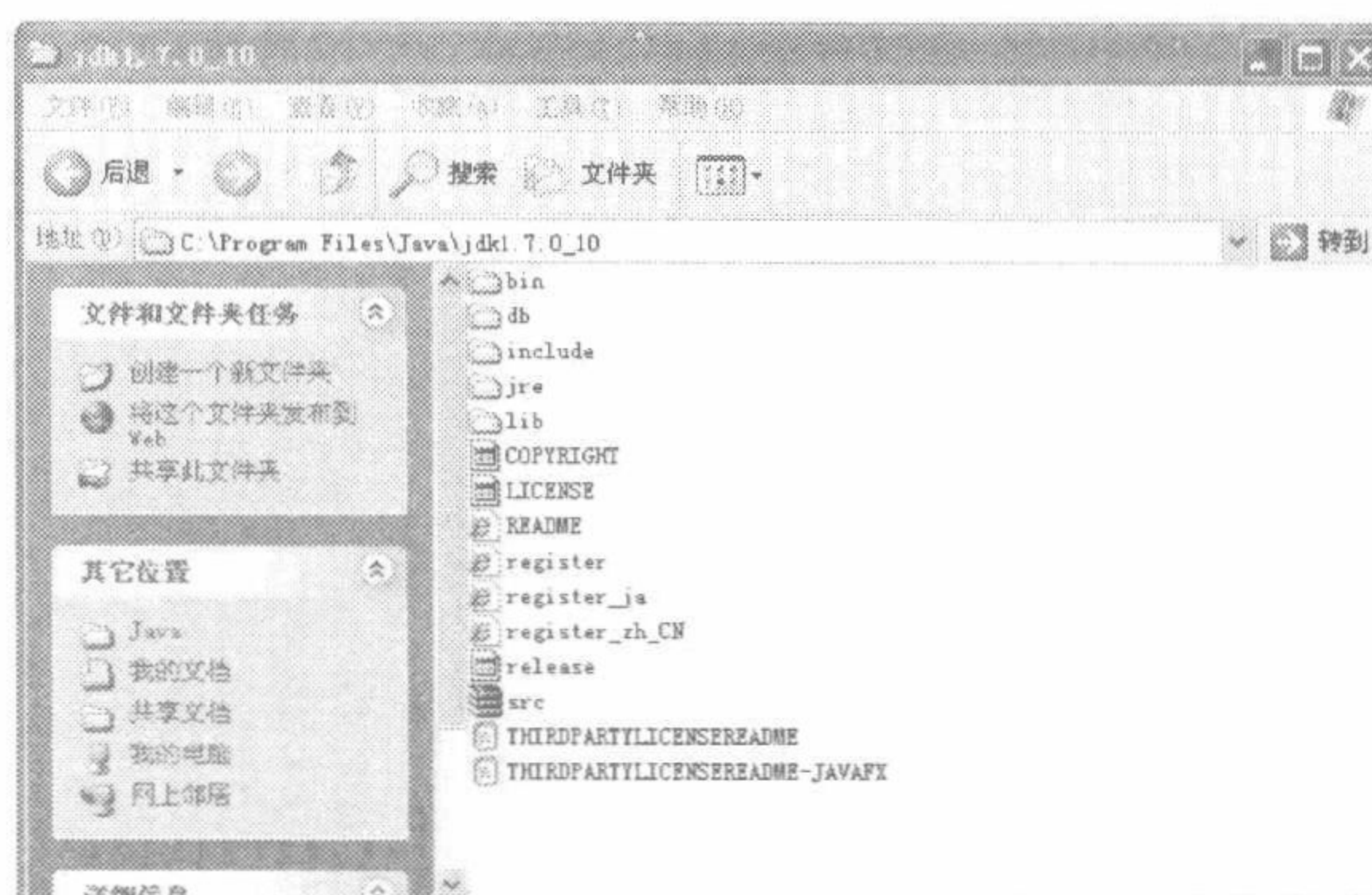


图 1-7 JDK 安装目录

从图 1-7 中发现，JDK 的目录下包含很多的文件夹和其他文件，下面对几个重要的文件夹和文件进行介绍。

- **bin** 目录 提供 JDK 工具程序，包括 javac、javadoc、appletviewer 等可执行程序。
- **demo** 目录 为 Java 使用者提供的一些已经编写好的范例程序。
- **include** 目录 存放用于本地方法的文件。
- **jre** 目录 存放 Java 运行环境文件。
- **lib** 目录 存放 Java 的类库文件。
- **src.zip** Java 提供的 API 类的源代码压缩文件，这个文档中包含 API 中某些功能的具体实现。

1.1.2.2 设置环境变量

在介绍 JDK 环境变量配置的书中，往往要求配置 Classpath 和 Path，Classpath 用于指定 JDK 指定的工具程序所在的位置。Classpath 是 Java 程序运行所特需的环境变量，用于指定运行的 Java 程序所需的类的加载路径。但是随着 JDK 版本的不断升级，针对本书介绍的版本只需要配置一个 Path 变量就可以了。

设置 Path 变量的两种形式，如下所示。

1. 使用命令行设置 Path 变量

打开命令行窗口，输入如下命令。

```
set path=%path%;C:\Program Files\Java\jdk1.7.0_10\bin
```

在上述的代码中，后面 C:\Program Files\Java\jdk1.7.0_10\bin 是 JDK 的安装目录，读者可以根据自己的安装情况来另行设置。设置好 Path 之后，可以在任何目录下执行 Java 命令。如图 1-8 所示。



图 1-8 使用命令行设置 path

2. 使用图形界面设置 Path 变量

使用界面的方式设置 Path 的步骤是：首先右击【我的电脑】图标，选择【属性】选项，在弹出的窗口中选择【高级】标签，如图 1-9 所示。接着单击下方的【环境变量】按钮，弹出如图 1-10 所示的界面。

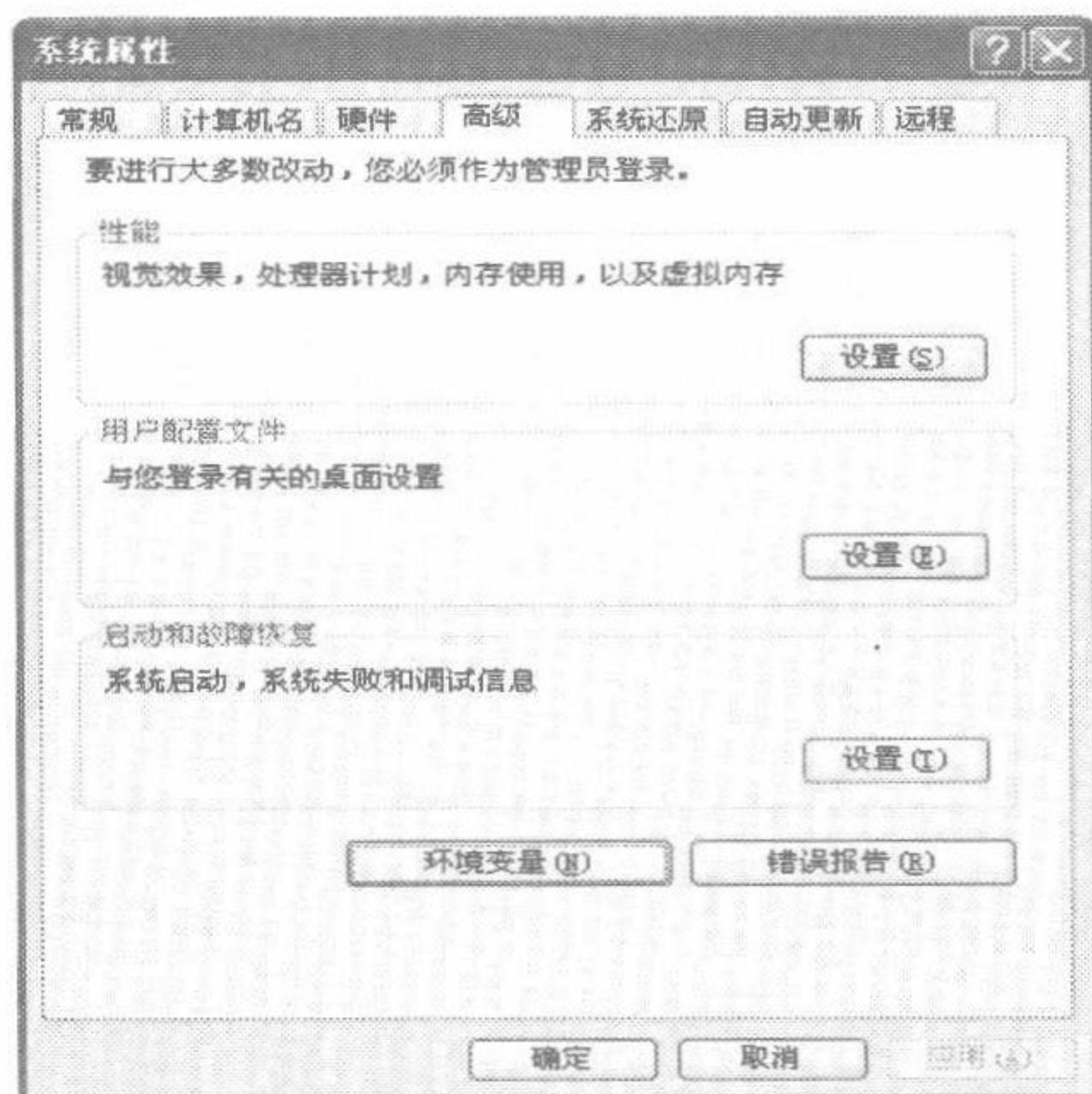


图 1-9 “系统属性”对话框

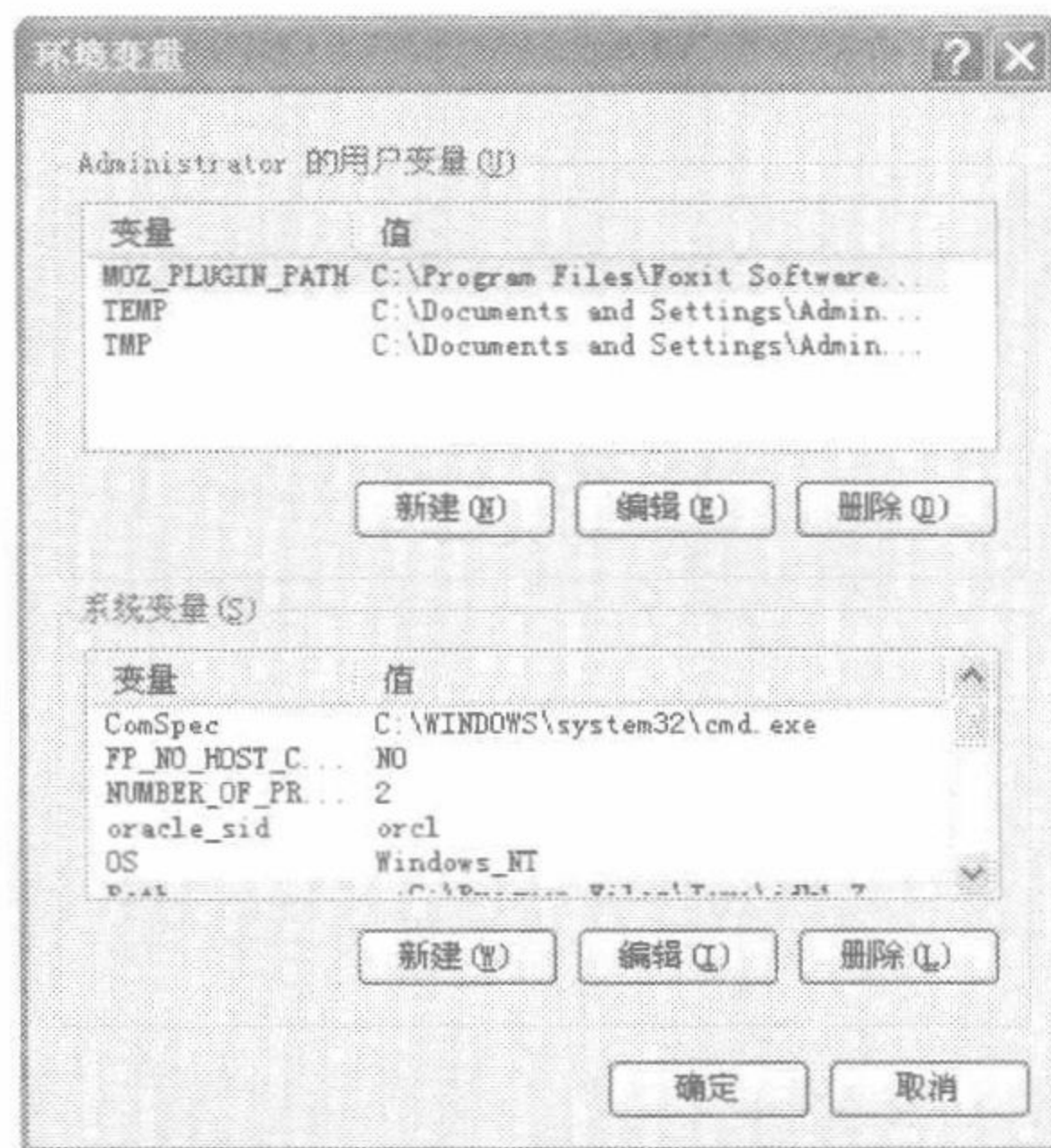


图 1-10 “环境变量”对话框

接着单击“环境变量”对话框下方的【新建】按钮，弹出“编辑系统变量”对话框，在【变量值】一栏的输入框中输入“.;C:\Program Files\Java\jdk1.7.0_10\bin”，如图 1-11 所示。

1.1.2.3 开发工具介绍

百度搜索 Java 开发工具，读者会发现可以查到的 Java 开发工具非常多，包括 Jcreator、JavaWorkshop、Eclipse、JBuilder 以及 JDeveloper 在内的多款编程软件。选择好的工具是进行方便快捷的开发的前提。在众多的软件

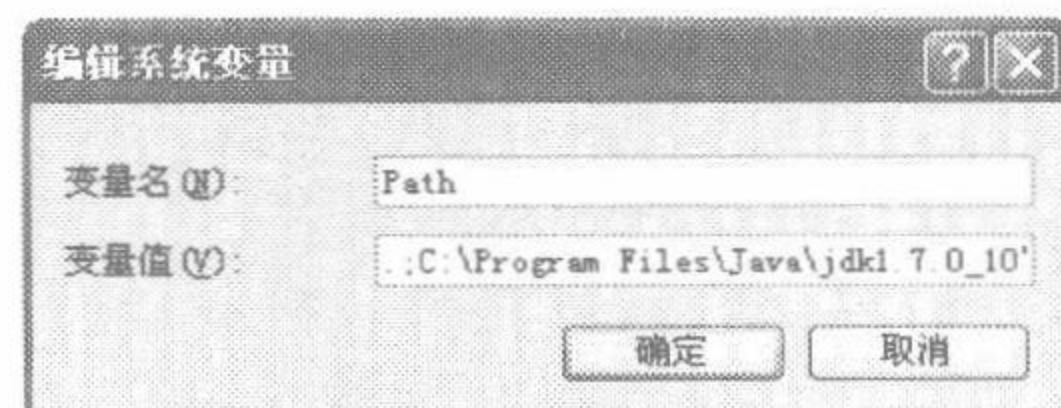


图 1-11 “编辑系统变量”对话框

中，Eclipse 已经被广大爱好者所使用，是一款具有较好发展态势的软件。

在介绍开发工具之前，首先介绍一下 Eclipse 的下载和安装方法。

(1) 打开 Eclipse 的官网 (<http://www.eclipse.org/>), 如图 1-12 所示。



图 1-12 Eclipse 官网

(2) 单击右边的【Download Eclipse】按钮，跳转至“Eclipse Download”页面，如图 1-13 所示。

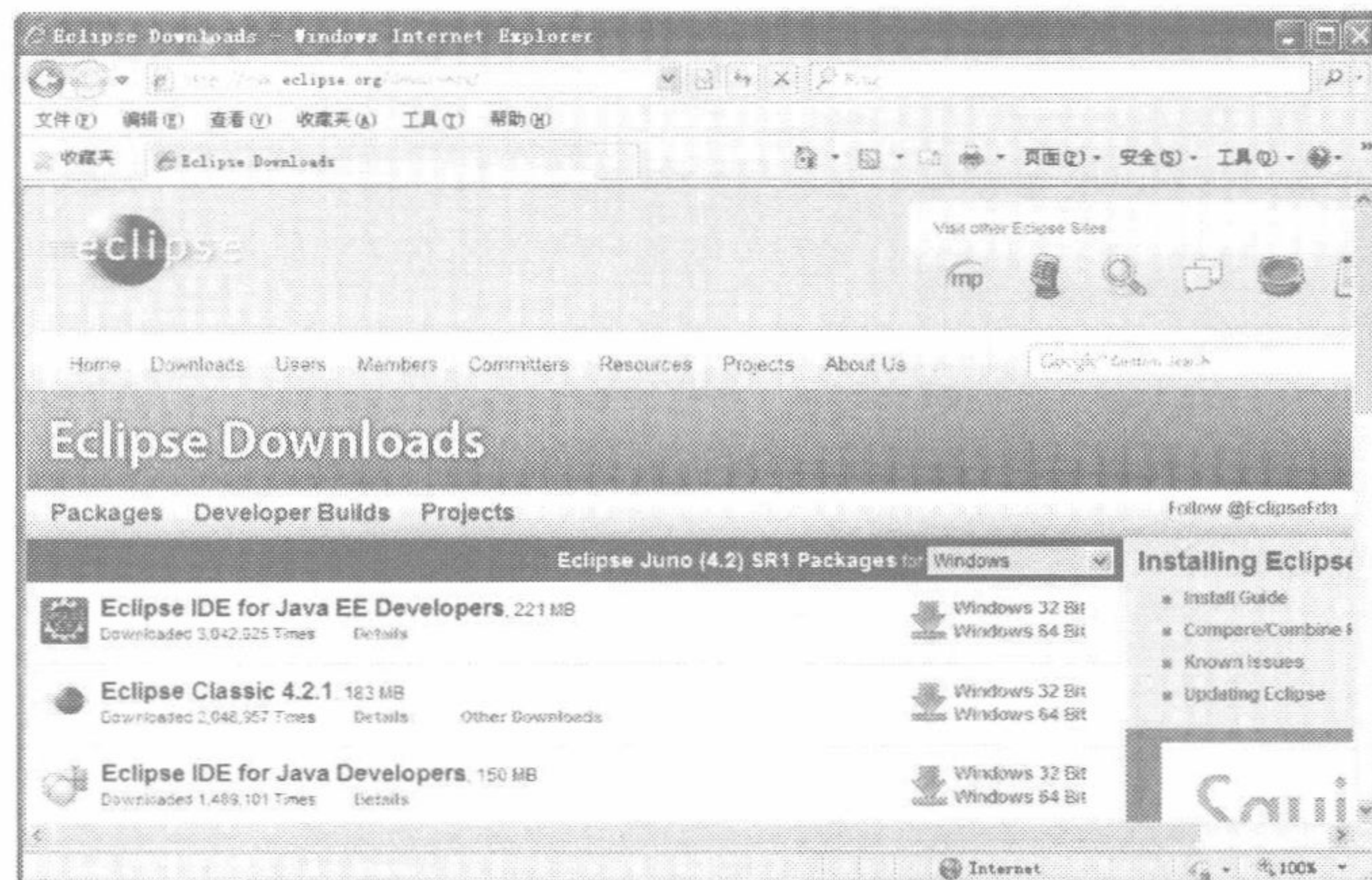


图 1-13 Eclipse Download 页面

(3) 单击第一项右边的“Windows 32 Bit”超链接，下载“eclipse-jee-juno-SR1-win32.zip”文件。

安装 Eclipse 的方法其实很简单，只需要将下载的压缩文件进行解压即可。然后在解压后的文件中打开 eclipse.exe 文件，程序就开始启动，如图 1-14 所示。接着会显示如图 1-15 的对话框，要求选择一个工作区，来保存以后编写的项目文件。单击【OK】按钮，或者单击【Browse】按钮重新选择工作区，之后再单击【OK】按钮即可出现 Eclipse 欢迎界面。

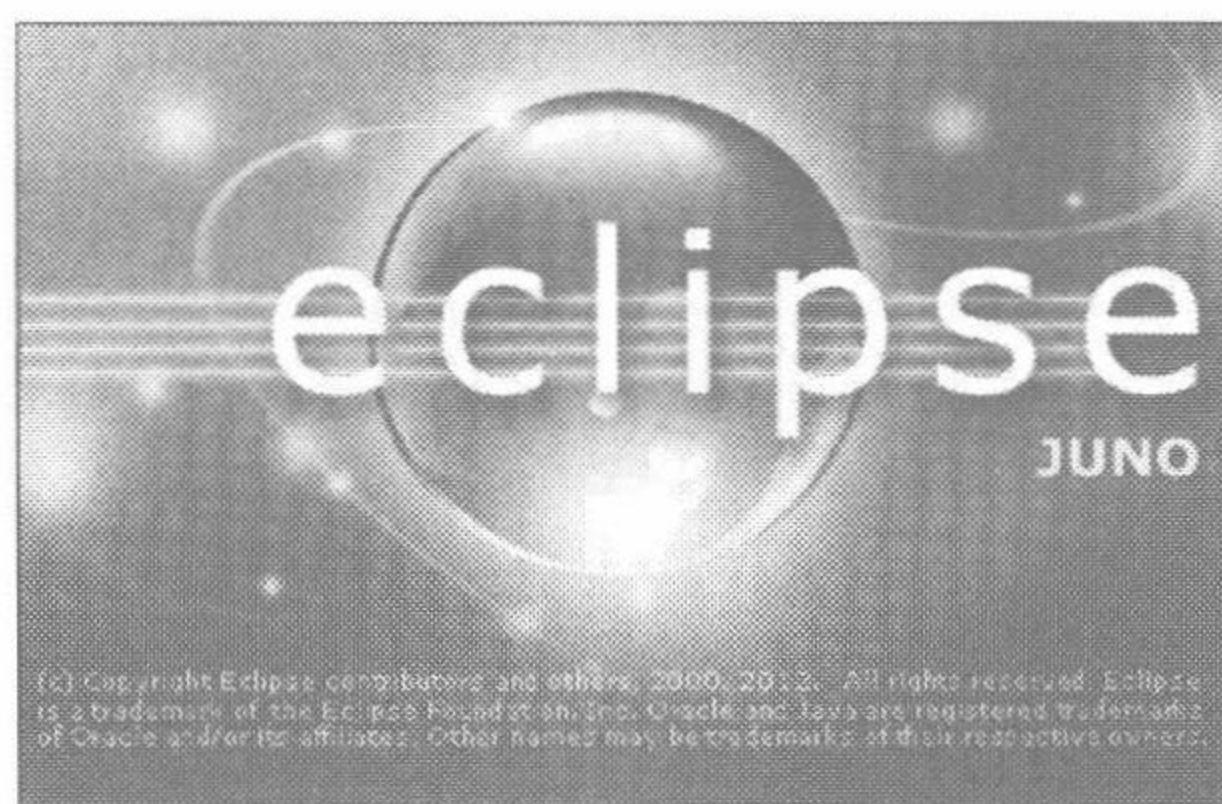


图 1-14 Eclipse 启动

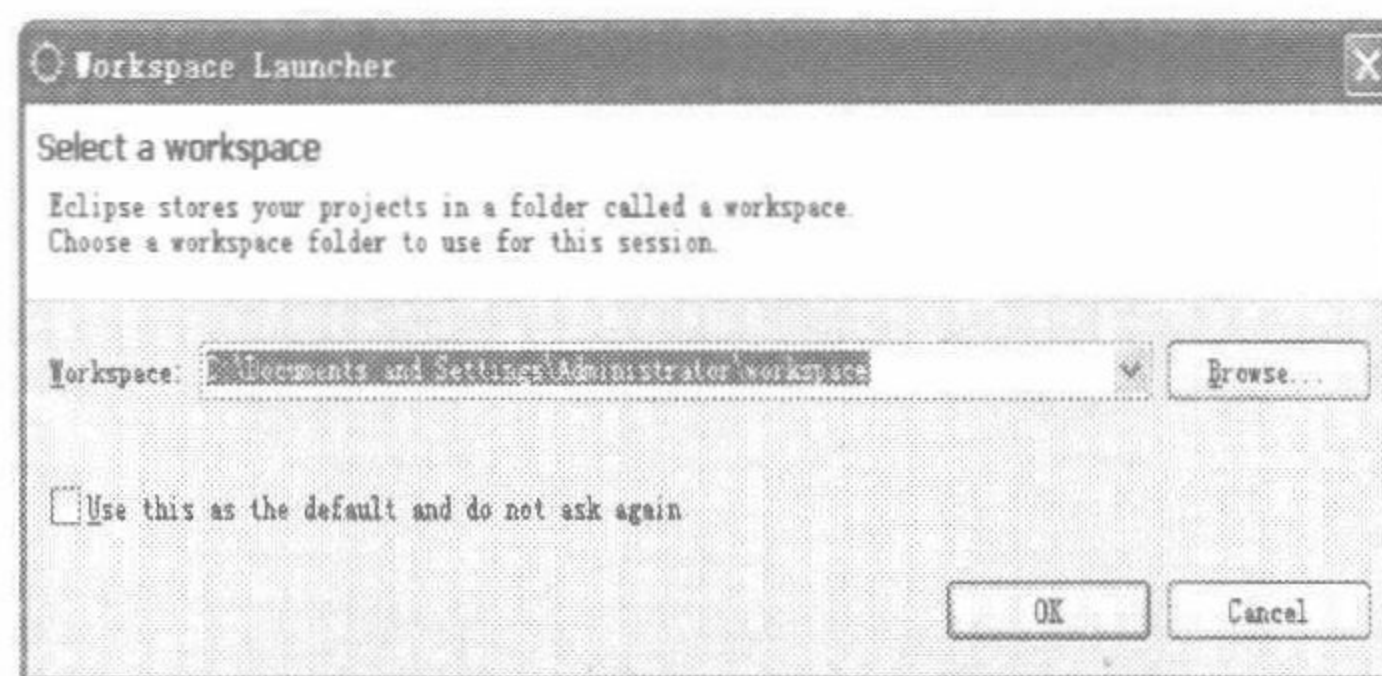


图 1-15 选择工作区

在 Eclipse 的使用过程中，如果计算机上已经安装 Eclipse，要安装新版本的话，必须先删除旧的版本，不能直接将新版本解压到老版本的安装目录下。

1.1.3 简单的 Java 程序

环境搭建完成之后读者可以开始编写第一个 Java 程序了。学过其他编程语言的读者会发现在之前的编程语言的学习中，基本上都是从一个 HelloWorld 的程序开始的，其基本功能就是输出“HelloWorld”语句。下面通过两种方式来完成 HelloWorld 程序的编写。第一种方式是使用文本编辑器编写完之后在命令行执行，第二种方式是使用 Eclipse 工具来编写。

1.1.3.1 编写程序

在这里首先介绍如何使用文本编辑器以及命令行的方式来编写程序。第一步，打开任何一种文本编辑器，在新建的文本文件中写入如下代码。

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

将写完代码后的文件另存为“HelloWorld.java”，保存类型选择为【所有文件】，接着单击【保存】按钮。如图 1-16 所示。

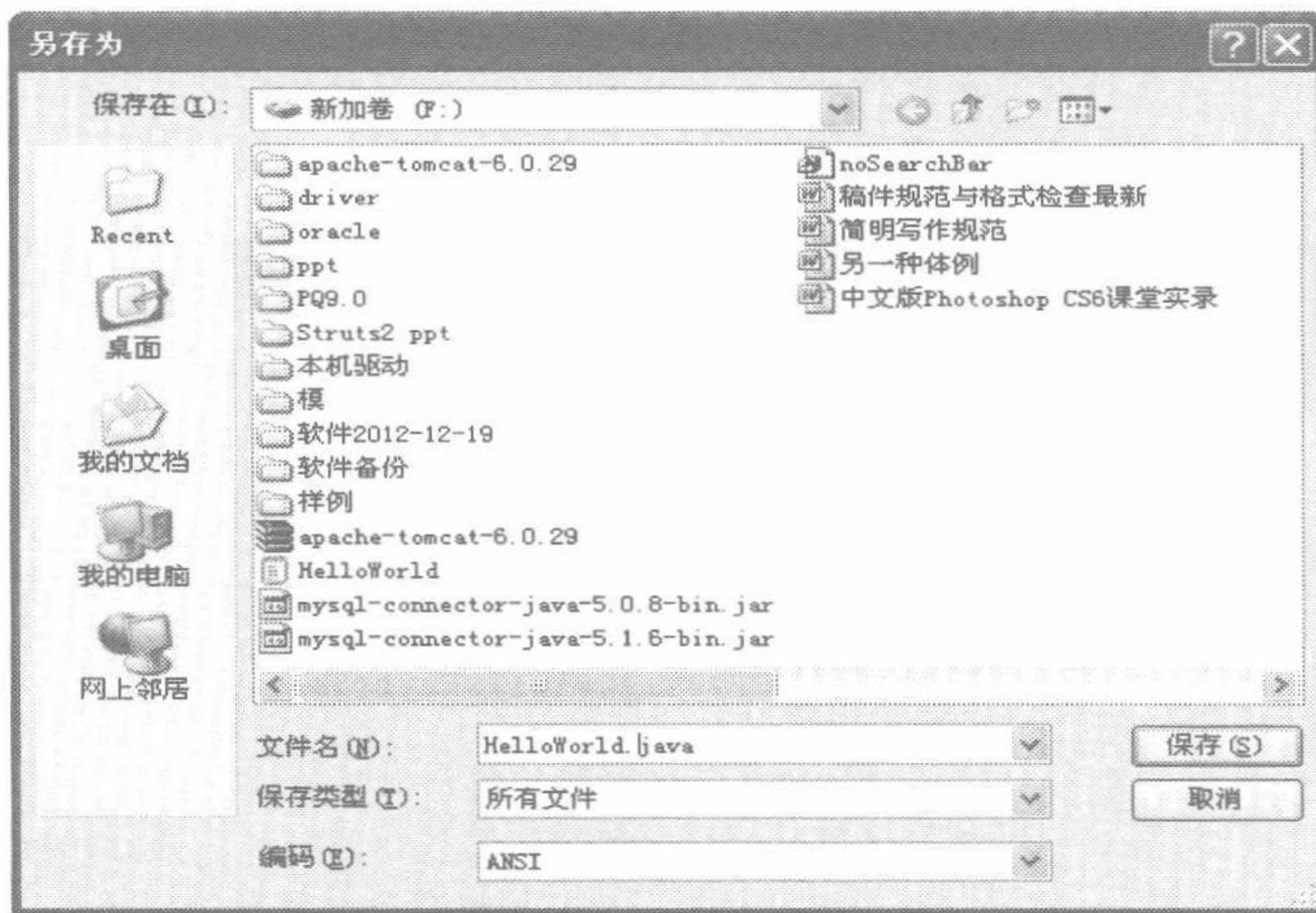


图 1-16 保存“HelloWorld.java”文件

1.1.3.2 编译和运行

在将文件保存完成之后，把文件放在 F 盘下。接着执行第二步操作，编译和执行。打开命令行，输入“F:”，将命令切换到文件的父级目录，在命令行输入如下代码。

```
javac HelloWorld.java
```

在源代码执行成功之后，会发现在 F 盘下会产生一个“HelloWorld.class”文件，这说明源代码已经编译为字节码程序。

接下来是该代码执行的最后一个步骤，在命令行输入如下代码。

```
java HelloWorld
```