

中央财政“支持高等职业学校提升专业服务产业发展能力”项目建设成果

Java 程序设计

——理实一体化教学课程

JAVA CHENGXUSHEJI

LISHI YITIHUA JIAOXUE KECHENG

主编◎钱 立 郭 琳



西南交通大学出版社

[Http://press.swjtu.edu.cn](http://press.swjtu.edu.cn)

中央财政“支持高等职业学校提升专业服务产业发展能力”项目建设成果

Java 程序设计

——理实一体化教学课程

主 编 钱 立 郭 琳
副主编 唐 权 徐红梅
编 委 朱 倩 吴 瑕 许 俊
陈 印 骆文亮

图书在版编目 (CIP) 数据

Java 程序设计: 理实一体化教学课程 / 钱立, 郭琳
主编. —成都: 西南交通大学出版社, 2014.2
ISBN 978-7-5643-2856-6

I. ①J… II. ①钱… ②郭… III. ①JAVA 语言—程序
设计 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2014) 第 020026 号

Java 程序设计
——理实一体化教学课程
主编 钱立 郭琳

责任编辑	李芳芳
助理编辑	宋彦博
特邀编辑	张少华
封面设计	墨创文化
出版发行	西南交通大学出版社 (四川省成都市金牛区交大路 146 号)
发行部电话	028-87600564 87600533
邮政编码	610031
网 址	http://press.swjtu.edu.cn
印 刷	成都中铁二局永经堂印务有限责任公司
成品尺寸	185 mm × 260 mm
印 张	15.25
字 数	381 千字
版 次	2014 年 2 月第 1 版
印 次	2014 年 2 月第 1 次
书 号	ISBN 978-7-5643-2856-6
定 价	31.00 元

图书如有印装质量问题 本社负责退换
版权所有 盗版必究 举报电话: 028-87600562

前 言

Java 是一种可以编写跨平台应用程序的面向对象的程序设计语言,是由 Sun 公司于 1995 年 5 月推出的 Java 程序设计语言和 Java 平台(即 JavaSE、JavaEE、JavaME)的总称。虽然目前 Sun 公司被 Oracle 公司收购了,但 Oracle 公司仍然重视 Java,并继续努力发展 Java 语言和 Java 平台技术。

Java 编程语言是个简单、面向对象、分布式、解释性、健壮、安全与系统无关、可移植、高性能、多线程和动态的语言,其独特的特性吸引了广大计算机软件开发人员学习和使用。

本书中编写的内容和形式上体现了高职高专教育的培养目标,即“培养适合生产、建设、管理、服务第一线需要的高等技术应用型专门人才”,并符合高职高专教育培养要求,即培养具备较广知识面、有基本理论和原理知识支撑,具有实际熟练技能的人才。在编写教材时,不但讲解 Java 语言程序设计上的最基本、最常用的内容,也讲解了 Java 语言应用开发上常用到的知识,附有程序案例,力求简洁明了,使读者能较快入门,并能自己动手编写程序。

全书共 10 章,前 5 章是 Java 语言基础部分,后 5 章是 Java 基础应用开发部分。每一章关于程序编写上的知识点讲解,均附带完整的程序实例,并进行了一定的程序分析。通常在知识点讲解后,为巩固知识点和灵活应用,设计了适当的练习题。

本书由钱立、郭琳担任主编并负责统稿,具体编写分工为:第 1 章由唐权编写,第 2 章由唐权、朱倩编写,第 3 章由钱立编写,第 4 章由郭琳编写,第 5 章由吴瑕编写,第 6 章由徐红梅编写,第 7 章由钱立、骆文亮编写,第 8 章由许俊、陈印编写,第 9 章由钱立、朱倩编写,第 10 章由郭琳、陈印编写。在编写本书的过程中,有多位老师提出了宝贵的建议和修改意见,在此表示衷心的感谢。

由于作者水平有限,书中难免存在疏漏和不足之处,恳请各位专家、老师和同学提出宝贵意见(作者 E-mail: qianli_scvtc@126.com)。

编 者

2013 年 6 月

目 录

第 1 章 Java 概述	1
1.1 Java 的发展史及特点	1
1.2 Java 的体系结构及工作原理	5
1.3 搭建 Java 开发环境	7
1.4 第一个 Java 程序	10
1.5 本章小结	12
习 题	12
第 2 章 Java 语言编程基础	14
2.1 Java 语言符号	14
2.2 数据类型与常量变量	16
2.3 运算符与表达式	22
2.4 流程控制语句	28
2.5 数 组	37
2.6 本章小结	43
习 题	44
第 3 章 Java 面向对象编程基础	45
3.1 类和对象	45
3.2 类的构造方法	53
3.3 类的方法重载	56
3.4 类的继承和方法重写	57
3.5 super 与 this, static 与 final	61
3.6 单态模式与工厂模式	70
3.7 综合实例	71
3.8 本章小结	74
习 题	75
第 4 章 Java 抽象类、接口和包	76
4.1 抽象类	76
4.2 接 口	80
4.3 Java 包	85
4.4 Java 类库及 JDK API 文档	88
4.5 本章小结	105
习 题	105
第 5 章 Java 异常处理	107
5.1 Java 异常概述	107
5.2 异常的处理	108

5.3	异常的层次结构	112
5.4	自定义异常	115
5.5	本章小结	118
	习 题	119
第 6 章	Java 文件处理与输入输出流	120
6.1	文件处理	120
6.2	流的分类及层次	125
6.3	I/O 流的使用 I	127
6.4	I/O 流的使用 II	133
6.5	对象序列化	135
6.6	本章小结	137
	习 题	138
第 7 章	Java 图形用户界面设计基础	139
7.1	图形用户界面 GUI 及 AWT	139
7.2	AWT 事件处理模型 I	144
7.3	AWT 事件处理模型 II	149
7.4	AWT 的布局管理器	158
7.5	Swing	162
7.6	菜单、工具栏及对话框设计	170
7.7	Graphics 绘图及文字	178
7.8	本章小结	185
	习 题	186
第 8 章	Applet 基础及 Java 多媒体技术	187
8.1	Applet 基础	187
8.2	Java 多媒体相关技术	195
8.3	本章小结	201
	习 题	201
第 9 章	多线程及网络编程基础	202
9.1	Java 线程	202
9.2	网络编程基础	212
9.3	本章小结	227
	习 题	228
第 10 章	Java 数据库编程基础	229
10.1	JDBC 体系结构	229
10.2	JDBC API 接口及数据库编程实例	232
10.3	本章小结	235
	习 题	236
	参考文献	238

第 1 章 Java 概述

【学习目标】

- 了解 Java 语言的历史及特点；
- 了解面向对象设计的特点；
- 掌握 Java 的 JRE 和 JVM 含义及开发过程。

【知识要点】

- Java 的体系结构，JRE 及 JVM；
- 开发环境搭建；
- 第一个 Java 程序的编写，编译及运行。

1.1 Java 的发展史及特点

1.1.1 知识点

1. Java 的发展史

目前 Java 的中文官网是 http://www.java.com/zh_CN/，如图 1.1 所示。



图 1.1 Java 中文官网首页

Java 来自于 Sun 公司的一个叫 Green 的项目，其原先的目的是为家用电子消费产品开发一个分布式代码系统，这样就可以把 E-mail 发给电冰箱、电视机等家用电器，对它们进行控制，和它们进行信息交流。开始设计者准备采用 C++，但 C++ 太复杂，且安全性差，最后基

于 C++ 开发了一种新语言 Oak (Java 的前身)。Oak 是一种用于网络的精巧而安全的语言, Sun 公司曾以此投标一个交互式电视项目, 但结果被 SGI 打败。此时的 Oak 几乎无家可归, 恰巧此时 Mark Andreessen 开发的 Mosaic 和 Netscape 启发了 Oak 项目组成员, 他们用 Java 编制了 HotJava 浏览器, 得到了 Sun 公司首席执行官 Scott McNealy 的支持, 触发了 Java 进军 Internet。

Java 技术是由美国 Sun 公司倡导和推出的, 它包括 Java 语言和 Java Media APIS、Security APIs、Management APIs、Java Applet、Java RMI、JavaBean、JavaOS、Java Servlet、JavaServer Page 以及 JDBC 等。现把 Java 技术的发展历程简述如下:

1990 年, Sun 公司由 James Gosling 领导的小组设计了一种平台独立的语言 Oak, 主要用于为各种家用电器编写程序。

1995 年 1 月, Oak 被改名为 Java; 5 月 23 日, Sun 公司在 Sun World 95 上正式发布 Java 和 HotJava 浏览器。

1995 年 8 月至 12 月, Netscape 公司、Oracle 公司、Borland 公司、SGI 公司、Adobe 公司、IBM 公司、AT&T 公司、Intel 公司获得 Java 许可证。

1996 年 1 月, Sun 公司宣布成立新的业务部门——JavaSoft 部, 以开发、销售并支持基于 Java 技术的产品, 由 Alan Baratz 任总裁。同时推出 Java 开发工具包 JDK (Java Development Kit) 1.0, 为开发人员提供用来编制 Java 应用软件所需的工具。

1996 年 2 月, Sun 公司发布 Java 芯片系列, 包括 PicoJava、MicroJava 和 UltraJava, 并推出 Java 数据库连接 JDBC (Java Database Connectivity)。

1996 年 3 月, Sun 公司推出 Java WorkShop。

1996 年 4 月, Microsoft 公司、SCO 公司、苹果电脑公司 (Apple)、NEC 公司等获得 Java 许可证。Sun 公司宣布允许苹果电脑、HP、日立、IBM、Microsoft、Novell、SGI、SCO、Tandem 等公司将 Java 平台嵌入其操作系统中。

1996 年 5 月, HP 公司、Sybase 公司获得 Java 许可证。北方电讯公司宣布把 Java 技术和 Java 微处理器应用到其下一代电话机中的计划。5 月 29 日, Sun 公司在旧金山举行第一届 JavaOne 世界 Java 开发者大会, 业界人士踊跃参加。Sun 公司在大会上推出一系列 Java 平台新技术。

1996 年 8 月, JavaWorkShop 成为 Sun 公司通过互联网提供的第一个产品。

1996 年 9 月, Addison-Wesley 和 Sun 公司推出 Java 虚拟机规范和 Java 类库。

1996 年 10 月, 德州仪器等公司获得 Java 许可证。Sun 公司提前完成 JavaBean 规范并发布, 同时发布第一个 Java JIT (Just-In-Time) 编译器, 并打算在 Java WorkShop 和 Solaris 操作系统中加入 JIT。10 月 29 日, Sun 公司发布 Java 企业计算技术, 包括 JavaStation 网络计算机、65 家公司发布的 85 个 Java 产品及应用、7 个新的 Java 培训课程及 Java 咨询服务、基于 Java 的 Solstice 互联网邮件软件、新的 Java 开发者支持服务、HotJava Views 演示、Java Tutor、Java Card API 等。Sun 公司宣布完成 Java Card API 规范, 这是智能卡使用的第一个开放 API。Java Card 规范将把 Java 能力赋予全世界亿万张智能卡。

1996 年 11 月, IBM 公司获得 JavaOS 和 HotJava 许可证。Novell 公司获得 Java WorkShop 许可证。Sun 公司和 IBM 公司宣布双方就提供 Java 化的商业解决方案达成一项广泛协议, IBM 公司同意建立第一个 Java 检验中心。

1996 年 12 月, Xerox 等公司获得 Java 或 JavaOS 许可证。Sun 公司发布 JDK1.1、Java 商贸工具包、JavaBean 开发包及一系列 Java APIs; 推出一个新的 JavaServer 产品系列, 其中包括 Java Web Server、Java NC Server 和 JavaServer Toolkit。Sun 公司发布 100%纯 Java 计划, 得到百家公司的支持。

1997 年 1 月, SAS 等公司获得 Java 许可证。Sun 公司交付完善的 JavaBean 开发包, 这是在确定其规范后不到 8 个月完成的。

1997 年 2 月, Sun 公司和 ARM 公司宣布同意使 JavaOS 运行在 ARM 公司的 RISC 处理器架构上。Informix 公司宣布在其 Universal Server 和其他数据库产品上支持 JDK1.1。Netscape 公司宣布其 Netscape Communicator 支持所有 Java 化的应用程序和核心 API。

1997 年 3 月, HP 公司获得 Java WorkShop 许可证, 用于其 HP-UX 操作系统。西门子、AG 公司等获得 Java 许可证。日立半导体公司、Informix 公司等获得 JavaOS 许可证。Novell 公司获得 Java Studio 许可证。Sun 公司发售的 JavaOS 1.0 操作系统, 这是一种在微处理器上运行 Java 环境的最小、最快的方法, 可提供给 JavaOS 许可证持有者使用。Sun 公司发售 HotJava Browser 1.0, 这是一种 Java 浏览器, 可以方便地按需编制专用的信息应用软件, 如客户自助台和打上公司牌号的网络应用软件。

1996 年 6 月, Sun 公司发布 JSP1.0, 同时推出 JDK1.3 和 Java Web Server 2.0。

1999 年 11 月, Sun 公司发布 JSP1.1, 同时推出 JSWDK1.0.1 和 Java Servlet 2.2。

2000 年 9 月, Sun 公司发布 JSP1.2 和 Java Servlet 2.3 API。

2001 年 6 月 5 日, NOKIA 宣布, 到 2003 年将出售 1 亿部支持 Java 的手机。

2001 年 9 月 24 日, J2EE1.3 发布。

2002 年 2 月 26 日, J2SE1.4 发布, 自此 Java 的计算能力有了大幅提升。

2004 年 9 月 30 日 18:00, J2SE1.5 发布, 成为 Java 语言发展史上的又一里程碑。为了表示该版本的重要性, J2SE1.5 更名为 Java SE 5.0

2005 年 6 月, JavaOne 大会召开, Sun 公司公开 Java SE 6。此时, Java 的各种版本已经更名, 以取消其中的数字“2”: J2EE 更名为 Java EE, J2SE 更名为 Java SE, J2ME 更名为 Java ME。

2006 年 12 月, Sun 公司发布 JRE6.0。

2009 年 04 月 20 日, Oracle 公司以 74 亿美元收购 Sun, 取得 java 的版权。

2010 年 11 月, 由于 Oracle 公司对于 Java 社区的不友善, Apache 扬言将退出 JCP。

2011 年 7 月 28 日, Oracle 公司发布 java 7.0 的正式版。

2. Java 语言的特点

Java 语言有许多有效的特性吸引着程序员们, 最主要的有以下几个:

1) 简洁有效

Java 语言是一种相当简洁的“面向对象”程序设计语言。Java 语言省略了 C++语言中所有的难以理解、容易混淆的特性, 例如头文件、指针、结构、单元、运算符重载、虚拟基础类等。它更加严谨、简洁。

2) 可移植性

对一个程序员而言, 写出来的程序不需修改就能够同时在 Windows、MacOS、UNIX 等

平台上运行，简直是不可想象的好事，而 Java 语言让这个原本遥不可及的事变得可能。使用 Java 语言编写的程序，只要做较少的修改，甚至有时根本不需修改就可以在不同的平台上运行。

3) 面向对象

可以这么说，“面向对象”是软件工程学的一次革命，大大提升了人类的软件开发能力，是一个伟大的进步，是软件发展的一个重大的里程碑。

在过去的 30 年间，“面向对象”有了长足的发展，充分体现了其自身的价值，到现在已经形成了一个包含了“面向对象的系统分析”、“面向对象的系统设计”、“面向对象的程序设计”的完整体系。所以作为一种现代编程语言，是不能够偏离这一方向的，Java 语言也不例外。

4) 解释型

Java 语言是一种解释型语言，相对于 C/C++ 语言来说，用 Java 语言写出来的程序效率低，执行速度慢。但它正是通过在不同平台上运行 Java 解释器，对 Java 代码进行解释，来实现“一次编写，到处运行”的宏伟目标的。为了达到目标，牺牲效率还是值得的，况且，现在的计算机技术日新月异，运算速度也越来越快，用户是不会感到太慢的。

5) 适合分布式计算

Java 语言具有强大的、易于使用的联网能力，非常适合开发分布式计算的程序。Java 应用程序可以像访问本地文件系统那样通过 URL 访问远程对象。

使用 Java 语言编写 Socket 通信程序十分简单，使用它比使用任何其他语言都简单。而且它还十分适用于公共网关接口 (CGI) 脚本的开发，另外还可以使用 Java 小应用程序 (Applet)、Java 服务器页面 (Java Server Page, 简称 JSP)、Servlet 等手段来构建更丰富的网页。

6) 拥有较好的性能

正如前文所述，由于 Java 是一种解释型语言，所以它的执行效率相对就会慢一些，但由于 Java 语言采用了两种手段，使其拥有较好的性能。

(1) Java 语言源程序编写完成后，先使用 Java 伪编译器进行伪编译，将其转换为中间码（也称为字节码），再解释；

(2) 提供了一种“准实时” (Just-In-Time, JIT) 编译器，当需要更快的速度时，可以使用 JIT 编译器将字节码转换成机器码，然后将其缓冲下来，这样速度就会更快。

7) 健壮、防患于未然

Java 语言在伪编译时，做了许多早期潜在问题的检查，并且在运行时又做了一些相应的检查，可以说是一种最严格的“编译器”。

它的这种“防患于未然”的手段将许多程序中的错误扼杀在摇篮之中。经常有许多在其他语言中必须通过运行才会暴露出来的错误，在编译阶段就被发现了。

另外，在 Java 语言中还具备了许多保证程序稳定、健壮的特性，有效地减少了错误，这使得 Java 应用程序更加健壮。

8) 具有多线程处理能力

线程，是一种轻量级进程，是现代程序设计中必不可少的一种特性。多线程处理能力使

得程序能够具有更好的交互性、实时性。

Java 在多线程处理方面性能超群，具有让设计者惊喜的强大功能，而且在 Java 语言中进行多线程处理很简单。

9) 具有较高的安全性

由于 Java 语言在设计时，在安全性方面考虑很仔细，做了许多探究，使得 Java 语言成为目前最安全的一种程序设计语言。

尽管 Sun 公司曾经许诺过：“通过 Java 可以轻松构建出防病毒、防黑客的系统”，但“世界上没有绝对的安全”这一真理是不会因为某人的许诺而失灵的。

就在 JDK (Java Development Kit)1.0 发布不久后，美国 Princeton (普林斯顿) 大学的一组安全专家发现了 Java1.0 安全特性中的第一例错误。从此，Java 安全方面的问题开始被关注。不过至今所发现的安全隐患都很微不足道，而且 Java 开发组还宣称，他们对系统安全方面的 Bugs 非常重视，会对这些被发现的 Bugs 立即进行修复。

而且由于 Sun 公司开放了 Java 解释器的细节，所以有助于通过各界力量，共同发现、防范、制止这些安全隐患。

10) Java 是一种动态语言

Java 是一种动态的语言，这表现在以下两个方面：

- (1) 在 Java 语言中，可以简单、直观地查询运行时的信息；
- (2) 可以将新代码加入一个正在运行的程序中去。

11) Java 是一种中性结构

“Java 编译器生成的是一种中性的对象文件格式”。也就是说，Java 编译器通过伪编译后，将生成一个与任何计算机系统无关的“中性”的字节码。

这种中性结构其实并不是 Java 首创的，在 Java 出现之前 UCSD Pascal 系统就已在一种商业产品中做到了这一点，另外在 UCSD Pascal 之前也有这种方式的先例，在 Niklaus Wirth 实现的 Pascal 语言中就采用了这种降低一些性能，换取更好的可移植性和通用性的方法。

Java 的这种字节码经过了许多精心的设计，使得其能够很好地兼容于当今大多数流行的计算机系统，在任何机器上都易于解释，易于动态翻译成为机器代码。

1.2 Java 的体系结构及工作原理

1. Java 的体系结构及技术平台

Java 不仅仅是一种网络编程语言，还是一个不断扩展的开发平台。Sun 公司针对不同的市场目标和设备进行定位，把 Java 划分为如下三个平台：

(1) J2SE (Java2 Standard Edition) 是 Java2 的标准版，主要用于桌面应用程序的编程。它包含了构成 Java 语言基础和核心的类。我们学习 Java 的过程，主要是在该平台上进行的。

(2) J2EE (Java2 Enterprise Edition) 是 Java2 的企业版，主要是为企业应用提供一个服

务器的运行和开发平台。J2EE 不仅包含 J2SE 中的类，还包含了诸如 EJB、servlet、JSP、XML 等许多用于开发企业级应用的类包。J2EE 本身是一个开放的标准，任何软件厂商都可以推出自己符合 J2EE 标准的产品，J2EE 将逐步发展成为强大的网络计算平台。

(3) J2ME (Java2 Micro Edition) 是 Java2 的微缩版，主要为消费电子产品提供一个 Java 的运行平台，使得能够在手机、机顶盒、PDA 等消费电子产品上运行 Java 程序。

2. Java 运行时环境 (JRE)

JRE 为 Java Runtime Environment (Java 运行时环境) 的简称，Java Runtime Environment (包括 Java Plug-in) 是 Sun 的产品，包括两部分：Java Runtime Environment 和 Java Plug-in。JRE 是可以在其上运行、测试和传输应用程序的 Java 平台。它包括 Java 虚拟机、Java 核心类库和支持文件。它不包含开发工具如编译器、调试器和其他工具。JRE 需要辅助软件 Java Plug-in，以便在浏览器中运行 applet。

3. Java 虚拟机 (JVM)

Java 虚拟机 (JVM) 是可运行 Java 代码的假想计算机。只要根据 JVM 规范描述将解释器移植到特定的计算机上，就能保证经过编译的任何 Java 代码能够在该系统上运行，如图 1.2 所示。

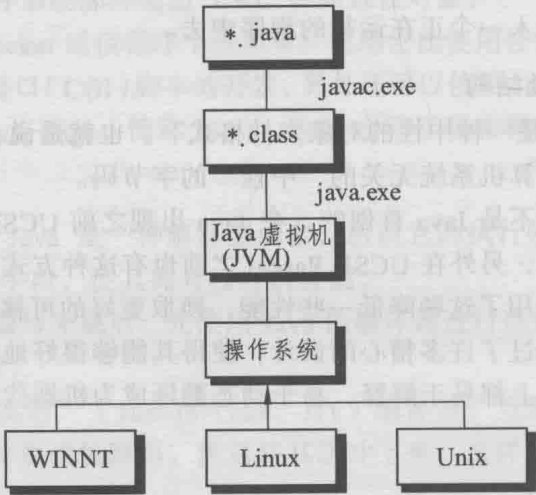


图 1.2 Java 虚拟机

从图 1.2 中不难明白 Java 可以实现可移植性的原因，只要在操作系统上 (WINNT、Linux、Unix) 植入 JVM，Java 程序就具有可移植性，也符合 Sun 公司提出的口号 “Write Once, Run Anywhere” (“一次编写，处处运行”)。

4. 内存自动回收机制

在程序的执行过程中，系统会给创建的对象分配内存，当这些对象不再被引用时，它们所占用的内存就处于废弃状态，如果不及时对这些废弃的内存进行回收，就会带来程序运行效率下降等问题。

在 Java 运行环境中，始终存在着一个系统级的线程，专门跟踪对象的使用情况，定期检测出不再使用的对象，自动回收它们占用的内存空间，并重新分配这些内存空间让它们为程序所用。Java 的这种废弃内存自动回收机制，极大地方便了程序设计人员，使他们在编写程序时不需要考虑对象的内存分配问题。

5. 代码安全性检查机制

Java 是网络编程语言，在网络上运行的程序必须保证其安全性。如何保证从网络上下载的 Java 程序不携带病毒而安全地执行呢？Java 提供了代码安全性检查机制。

Java 在将一个扩展名为 .class 的字节码文件装载到虚拟机执行之前，先要检验该字节码文件是否符合字节码文件规范，代码中是否存在着某些非法操作。检验工作由字节码检验器 (Bytecode Verifier) 或安全管理器 (Security Manager) 进行。检验通过之后，将字节码文件加载到 Java 虚拟机中，由 Java 解释器解释为机器码并执行。Java 虚拟机把程序的代码和数据都限制在一定内存空间里执行，不允许程序访问超出该范围，保证了程序的安全运行。

1.3 搭建 Java 开发环境

1. 安装 JDK

对于学习 Java 者来说，应该使用 Sun 公司的 Java 开发工具箱 JDK(Java Development Kit)，它拥有最新的 Java 程序库，功能逐渐增加且版本在不断更新。Sun 公司 2009 年已被 Oracle 公司收购，但 Oracle 公司仍在支持 Java 的发展。对于软件开发人员可到 <http://www.oracle.com/technetwork/java/javase/downloads/index.html> 页面免费下载 Java 标准版，如图 1.3 所示。下面我们在 Microsoft Windows 操作系统平台上安装 JDK，建立 Java 的开发环境。



图 1.3 JDK 下载页面

以 JDK7 版本为例,从上面给出的站点页面下载安装文件 jdk-7u40-windows-i586.exe。双击该安装文件,按照安装文件的提示一步步执行即可安装,如图 1.4 所示。JDK 的默认安装路径是 C:\Program Files\Java\jdk1.7。

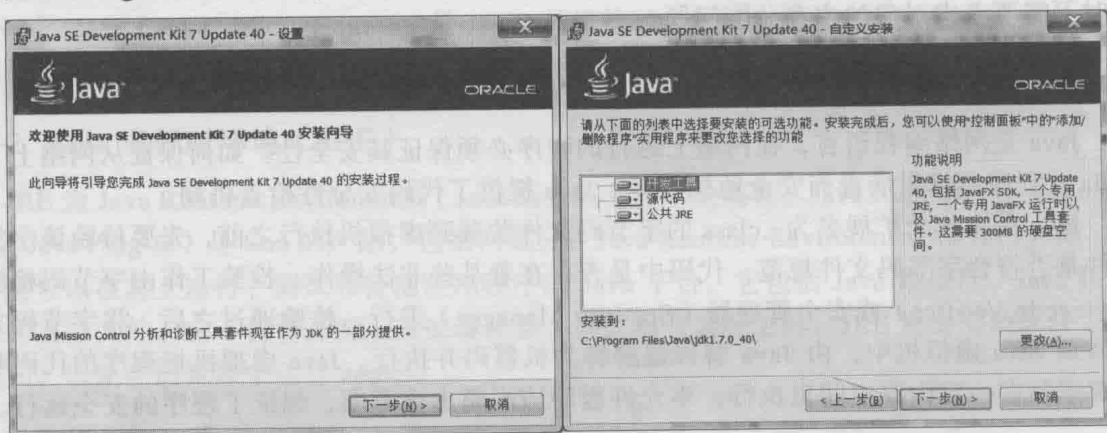


图 1.4 JDK 安装过程

2. 设置环境变量

在运行 Java 程序或进行一些相关处理时,将用到工具箱中的工具和资源,这就需要设置两个环境变量 PATH 和 CLASSPATH,以获取运行工具和资源的路径。

在不同的操作系统下设置环境变量的方式有所不同,下面以 Windows XP 系统为例,说明设置环境变量的操作方法和步骤。

- (1) 右击“我的电脑”图标,在快捷菜单中单击“属性”选项;
- (2) 在出现的“系统属性”对话框窗口上单击“高级”选项,并选择“环境变量”按钮;
- (3) 在出现的“环境变量”对话框上,单击用户变量框内的“新建”按钮;
- (4) 出现如图 1.5 所示的“新建用户变量”对话框窗口。在“变量名”文本框中输入“CLASSPATH”;在“变量值”文本框中输入“.;c:\jdk1.7\lib\dt.jar;c:\jdk1.7\lib\tools.jar”;然后,单击“确定”按钮。这就设置了环境变量 CLASSPATH。

以同样方法设置 PATH,在“变量值”文本框中输入“c:\jdk1.6.0\bin”。

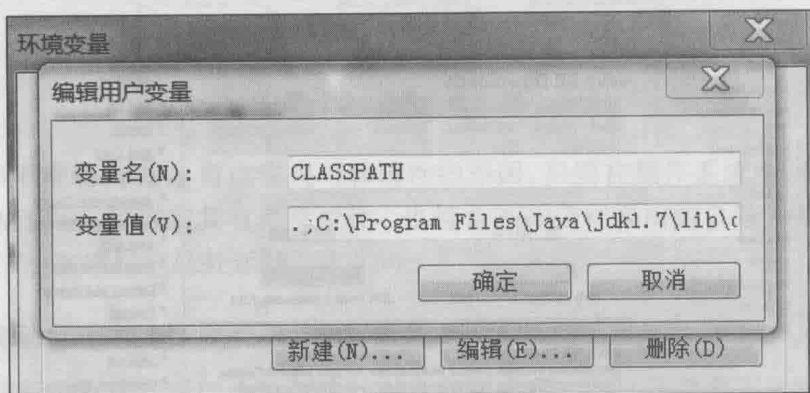


图 1.5 设置环境变量

3. Java 集成开发工具 IDE

IDE (Integrated Development Environment, 集成开发环境) 是可以辅助开发程序的应用软件。该软件是用于程序开发环境的应用程序, 一般包括代码编辑器、编译器、调试器和图形用户界面工具, 就是集成了代码编写功能、分析功能、编译功能、debug 功能等一体化的开发软件套。所有具备这一特性的软件或者软件套(组)都可以叫做 IDE。如微软的 Visual Studio 系列、Borland 的 C++ Builder、Delphi 系列等。

该程序可以独立运行, 也可以和其他程序并用。IDE 为开发人员使用 Visual Basic、Java 和 Power Builder 等现代编程语言提供了方便。不同的技术体系有不同的 IDE。比如可以称 C++、VB、C#、易语言等语言的集成开发环境为 IDE。

Java 的 IDE 常见的有 Eclipse、NetBeans 等, 本书主要采用 Eclipse 来进行 Java 开发学习。

Eclipse 是一个开放源代码的、基于 Java 的可扩展开发平台。就其本身而言, 它只是一个框架和一组服务, 用于通过插件组件构建开发环境。幸运的是, Eclipse 附带了一个标准的插件集, 包括 Java 开发工具(JDK)。Eclipse 还包括插件开发环境, 它的用途并不限于使用 Java 语言开发, 还可以支持诸如 C/C++、COBOL、PHP 等编程语言的插件, 进行其他语言的软件开发应用等, 可以从 <http://www.eclipse.org/downloads> 下载该 IDE, 如图 1.6 所示。



图 1.6 Eclipse 网站下载页面

NetBeans 由 Sun 公司(2009 年被 Oracle 公司收购)在 2000 年创立, 它是开放源运动以及开发人员和客户社区的家園, 旨在构建世界级的 Java IDE。NetBeans 当前可以在 Solaris、Windows、Linux 和 Macintosh OS X 平台上进行开发, 并在 SPL(Sun 公用许可)范围内使用。NetBeans IDE 可以使开发人员利用 Java 平台快速创建 Web、企业、桌面以及移动的应用程序, NetBeans IDE 目前支持 PHP、Ruby、JavaScript、Ajax、Groovy、Grails 和 C/C++ 等开发语言。可以从 <https://netbeans.org/downloads> 下载该 IDE, 如图 1.7 所示。

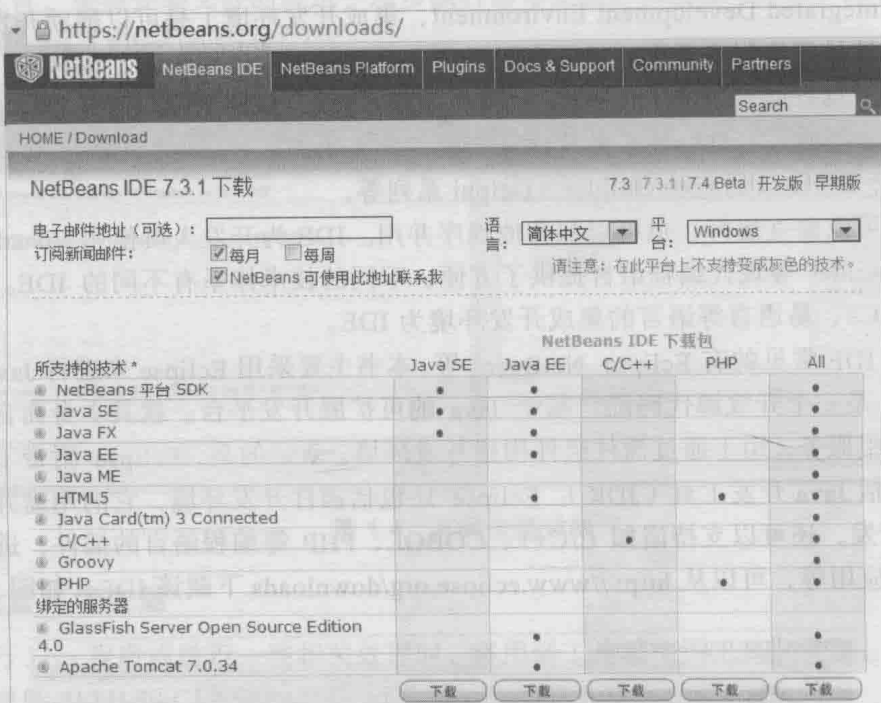


图 1.7 Eclipse 网站下载页面

1.4 第一个 Java 程序

1.4.1 Java 程序的分类

Java 程序分为两种形式：一种是网页上使用的 Applet 程序（Java 小程序），另一种是 Application 程序（即 Java 应用程序）。

1.4.2 教学案例

1. 案例描述

编写第一个 Java 程序，输出“世界，你好！”。

2. 操作步骤

1) 编写源代码

打开记事本，输入如下代码，录入结束后，保存文件在 D 盘，文件名为 Hello.java，如图 1.8 所示。

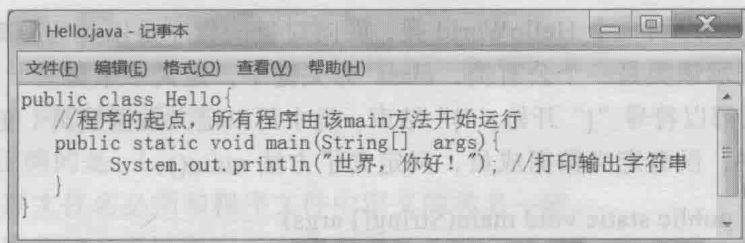


图 1.8 第一个 Java 源代码

2) 使用 Javac 命令编译源代码

Javac.exe 是 Java 源代码编译器。在操作系统左下角, 打开菜单“运行”, 输入“cmd”回车确认后, 弹出命令行窗口, 输入“javac d:\Hello.java”。如果没有编译错误, 将在 D 盘下产生编译好的 Hello.class 字节码文件, 如图 1.9 所示。

3) 使用 Java 命令执行字节码

Java.exe 是 Java 字节码编译器。在如上的命令行窗口中, 输入“java Hello”, 则可看到运行结果, 如图 1.9 所示。

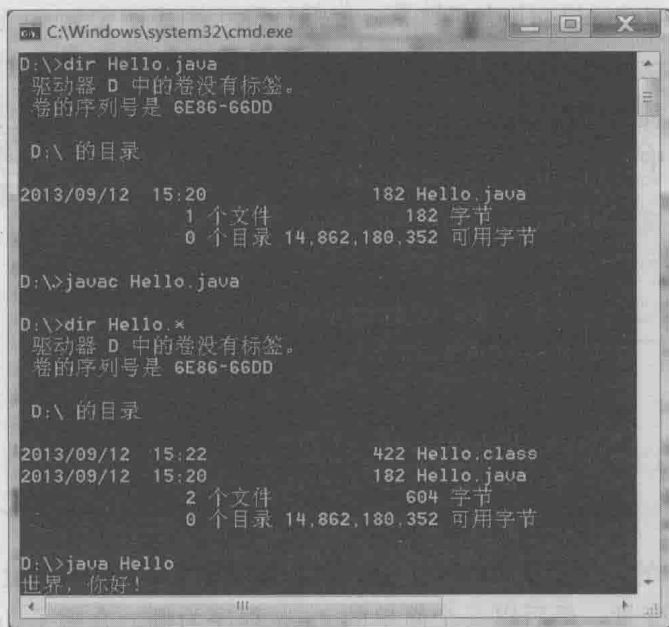


图 1.9 编译源文件和执行字节码

3. 案例分析

所有的 Java Application 都是从 public static void main(String args[])方法开始运行的, 这是一个程序入口点, 正如 C 语言程序的 main()方法, 学习者务必要牢记。

简要分析该程序的结构:

(1) 类似于 C 和 C++, 字符“//”为单行注解。注解行不是程序的代码部分, 只是为了程序易读。

(2) public class HelloWorld{}