

Java程序设计

主编 ◎ 孟祥飞



北京理工大学出版社

BEIJING INSTITUTE OF TECHNOLOGY PRESS

Java 程序设计

孟祥飞 主编

 **北京理工大学出版社**
BEIJING INSTITUTE OF TECHNOLOGY PRESS

图书在版编目 (CIP) 数据

Java 程序设计/孟祥飞主编. —北京: 北京理工大学出版社, 2019. 5

ISBN 978 - 7 - 5682 - 6676 - 5

I. ①J… II. ①孟… III. ①JAVA 语言 - 程序设计 - 教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2019) 第 016849 号

出版发行 / 北京理工大学出版社有限责任公司

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010) 68914775 (总编室)

(010) 82562903 (教材售后服务热线)

(010) 68948351 (其他图书服务热线)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 三河市天利华印刷装订有限公司

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 16

字 数 / 380 千字

版 次 / 2019 年 5 月第 1 版 2019 年 5 月第 1 次印刷

定 价 / 59.80 元

责任编辑 / 钟 博

文案编辑 / 钟 博

责任校对 / 周瑞红

责任印制 / 施胜娟

前 言

Java 语言是目前国内外广泛使用的计算机程序设计语言之一，也是当前进行 WEB 类应用系统开发的首选语言。它具有语言功能丰富、使用灵活方便、性能优异、可移植性强、安全性等特点，应用领域极其宽广。

本书是编者在多年一线的教学实践基础上，为适应当前教学改革和应用型课程建设的要求组织编写而成。本教材注重培养学生的实践与创新能力，融合了同类其他教材的特点，取长补短、融合创新，以项目为载体，采用任务驱动的编写形式，突出“从问题到算法，再到程序”的思维过程，强调计算机求解问题的思路引导和程序设计思维方式的训练，能够使学习者系统学习、快速入门。具体来说，本书具有如下特点：

一、精选实例，引入较多的实用性案例，注重培养学生阅读程序、参考、编写和实践的能力。

二、从实际出发，将知识融入到案例之中，循序渐进、通俗易懂。

三、按单元划分，有利于学生对知识的掌握和技术的运用。

本书内容充实全面，从 Java 技术的发展和学生认知的规律出发，将教学内容分为五个单元，每个单元以任务引导，包含若干实例，最后通过一个企业的真实开发流程贯穿所有知识点。

由于时间仓促和水平有限，书中难免存在缺点和不足，希望广大读者批评指正，以利于教材质量的提高。

编 者

CONTENTS 目录

学 习 篇

单元1 初识 Java	(3)
任务 1.1 搭建 Java 开发环境	(4)
1.1.1 Java 的发展历史	(4)
1.1.2 Java 的特点及优势	(5)
1.1.3 Java 的运行机制	(6)
1.1.4 JDK 简介	(6)
任务 1.2 初入 Java 世界	(10)
1.2.1 Java 程序结构	(11)
1.2.2 使用 Eclipse 创建 Java 程序	(12)
任务 1.3 计算扇形的面积和周长程序设计	(15)
1.3.1 Java 中的标识符	(16)
1.3.2 Java 中的关键字	(16)
1.3.3 Java 中的保留字	(17)
1.3.4 Java 中的注释	(17)
1.3.5 Java 中的分隔符	(17)
1.3.6 Java 中的常量与变量	(18)
1.3.7 Java 中的数据类型	(19)
1.3.8 Java 数据类型转换	(22)
1.3.9 Java 中的运算符与表达式	(23)
单元2 程序设计基础	(27)
任务 2.1 熟悉各流程的关键字	(27)
2.1.1 顺序结构	(27)
2.1.2 选择结构	(28)
2.1.3 分支语句结构	(31)
任务 2.2 while 循环语句、do...while 循环语句与 for 循环语句	(34)
任务 2.3 break 语句和 continue 语句	(38)
单元3 面向对象程序设计	(43)
任务 3.1 描述“人”信息程序设计	(43)
3.1.1 类与对象的概念与关系	(44)



3.1.2 类的定义	(45)
3.1.3 创建对象	(47)
任务 3.2 借书卡程序设计	(50)
3.2.1 类的方法	(51)
3.2.2 构造方法	(53)
3.2.3 方法重载	(56)
3.2.4 变量的作用域	(58)
3.2.5 定义包和导入包的关键词	(61)
任务 3.3 保护人的“隐私”程序设计	(64)
3.3.1 封装	(65)
3.3.2 Java 的修饰符	(67)
任务 3.4 员工信息管理程序设计	(71)
3.4.1 继承	(72)
3.4.2 方法的覆盖	(74)
3.4.3 this 和 super 关键字	(75)
3.4.4 多态	(76)
3.4.5 最终类和抽象类	(79)
任务 3.5 模拟 USB 接口程序设计	(84)
3.5.1 Java 接口	(85)
3.5.2 接口与多态	(87)
3.5.3 面向接口编程的步骤	(88)
3.5.4 接口中常量的使用	(90)
单元 4 数据与异常处理程序设计	(98)
任务 4.1 学生成绩计算程序设计	(98)
4.1.1 一维数组	(100)
4.1.2 二维数组	(107)
任务 4.2 天气预报的数据处理	(112)
4.2.1 创建 String 字符串	(115)
4.2.2 String 类的常用操作	(115)
4.2.3 StringBuffer 类的常用方法	(117)
任务 4.3 除法计算器程序设计	(120)
4.3.1 异常的概念及其处理机制	(122)
4.3.2 异常的分类	(122)
4.3.3 异常的捕获与处理	(123)
任务 4.4 最大公约数计算器设计	(128)
4.4.1 自定义异常	(130)
4.4.2 抛出异常	(131)
4.4.3 上报异常	(131)



单元5 图形用户界面程序设计	(137)
任务5.1 油耗计算程序设计	(137)
5.1.1 抽象窗口工具集 (AWT)	(138)
5.1.2 Swing 组件简介	(139)
5.1.3 JComponent 组件	(139)
5.1.4 JFrame 组件	(140)
5.1.5 Swing 工具集的其他常用组件	(141)
任务5.2 计算器界面设计	(147)
5.2.1 Java 布局管理	(148)
5.2.2 常见的布局管理器	(148)
任务5.3 计算器事件处理	(156)
5.3.1 Java 事件	(156)
5.3.2 Java 事件处理机制	(157)
5.3.3 Java 事件体系结构	(158)
5.3.4 Java 事件监听器和监听方法	(159)
任务5.4 字体设置窗口程序设计	(164)
5.4.1 组合框 JComboBox	(165)
5.4.2 复选框 JCheckBox	(166)
5.4.3 单选按钮 JRadioButton	(166)
5.4.4 列表框 JList	(167)
5.4.5 选择事件	(168)
任务5.5 字体设置菜单设计	(170)
5.5.1 菜单栏 (JMenuBar)	(171)
5.5.2 菜单项 (JMenu)	(172)
5.5.3 菜单子项 (JMenuItem)	(173)
5.5.4 复选菜单项 (JCheckBoxMenuItem)	(174)
5.5.5 单选菜单项 (JRadioButtonMenuItem)	(174)

实 验 篇

单元6 综合设计——“企业即时通”软件	(183)
任务6.1 项目需求分析及总体设计	(183)
6.1.1 项目需求分析	(183)
6.1.2 客户端业务流程分析	(185)
6.1.3 系统总体设计	(192)
任务6.2 详细设计之数据库设计	(193)
6.2.1 数据实体类分析	(193)
6.2.2 数据库与数据表创建	(194)
任务6.3 详细设计之界面设计	(195)
6.3.1 客户端程序界面设计	(196)



6.3.2 服务器端程序界面设计	(197)
任务 6.4 系统实现技术	(199)
6.4.1 网络通信基础知识	(199)
6.4.2 Socket(套接字) 技术实现	(200)
6.4.3 多线程技术实现	(208)
6.4.4 JDBC 实用技术	(215)
任务 6.5 服务器端功能的实现	(219)
6.5.1 数据实体类及业务类的实现	(219)
6.5.2 实现服务器端主窗体	(229)
6.5.3 实现 ServerListenThread 类	(233)
6.5.4 实现 ServerReceiveThread 类	(234)
6.5.5 处理聊天信息的请求	(234)
6.5.6 使用 JTable 显示在线用户信息	(234)
任务 6.6 客户端功能的实现	(235)
6.6.1 登录窗体的实现	(236)
6.6.2 客户端主窗体	(242)
6.6.3 实现客户端接收线程 ClientReceiveThread	(246)
6.6.4 聊天功能的实现	(247)



学 习 篇

单元 1

初识 Java

知识目标



- (1) 了解 Java 语言的发展历史;
- (2) 掌握 Java 语言的特点和运行机制;
- (3) 掌握 Java 语言的标识符、关键字和注释;
- (4) 掌握 Java 语言的数据类型和数据类型转换;
- (5) 掌握 Java 语言的运算符和表达式;
- (6) 掌握 Java 语言的顺序结构、选择结构和循环结构。

能力目标



- (1) 能够正确安装 JDK 和 Eclipse, 并配置 Java 开发环境;
- (2) 能够编写简单的 Java 程序;
- (3) 能够定义和使用不同类型的变量和常量;
- (4) 能够进行数据类型的转换以及熟练运用运算符和表达式;
- (5) 能够使用 Java 的 if 语句和 switch 语句进行流程控制;
- (6) 能够灵活运用循环语句解决实际问题。

Java 是一种可以开发跨平台应用程序的面向对象程序设计语言, 是美国 Sun 公司于 1995 年 5 月推出的 Java 程序设计和 Java 平台的总称。Java 具有良好的通用性、高效性、平台移植性和安全性, 广泛应用于个人计算机、数据中心、游戏控制台、超级计算机、移动电话和 Internet 等领域, 拥有全球最大的开发者专业社群。在全球云计算和移动互联网的产业环境下, Java 具备显著优势和广阔前景。本单元通过 3 个任务介绍 Java 开发环境的搭建、Java 程序结构、Java 数据类型等基础知识, 引导读者进入 Java 世界。



任务 1.1 搭建 Java 开发环境

【任务目标】

- (1) 了解 Java 的发展历史;
- (2) 掌握搭建 Java 集成开发环境的方法;
- (3) 掌握 Ecilpse 的基本使用方法。

【任务陈述】

在使用 Java 语言进行开发工作之前需要安装和配置 Java 开发环境。JDK (Java Development Kin) 是 Sun 公司为 Java 开发者提供的软件开发工具包。Java 的开发工具有多种, 最简单的是“记事本”与控制台的组合, 此外还有 UltraEdit、Jereator、NetBeans IDE、Jav、JBuilder 和 Eclipse 等。Java 集成开发环境的使用方法类似, 在学习过程中只需掌握其中几种即可。本任务以 Eclipse 为例介绍集成开发环境的基本使用方法。

【知识储备】

1.1.1 Java 的发展历史

1991 年, 美国的 Sun 公司成立专门的研究小组对家用消费类电子设备进行前沿研究。由 James Gosling 领导的 Green 小组进行软件相关研究, 该小组在开始阶段选择当时已经成熟的 C++ 语言进行设计和开发, 发现执行 C++ 程序需要大量设备资源, 且不能兼容不同的设备。因此, 该小组在吸收 C++ 语言优势的基础上, 自主开发了一种新的语言, 因公司门前有一棵橡树而起名为 Oak (橡树), 即 Java 语言的前身。

Green 小组的成果最终没有转变成 Sun 公司的产品, 也没有为 Sun 公司带来收益, Oak 面临夭折的危险。1995 年, 随着 Internet 的出现和迅速发展, 亟需一种可以在 Internet 客户端运行, 且能够在不同的计算机设备和操作系统中运行的程序设计语言。此时, Green 小组受到 Mark Ardreesen 开发的 Mosaic 和 Netscape 的启发, 用 Oak 开发了 HotJava 浏览器, 并得到了 Sun 公司首席执行官 Scott MeeNealy 的支持, 为 Oak 进军 Internet 创造了契机。但由于版权问题, Sun 公司不得不对该语言重新命名。在命名会议中有人提出以杯中的爪哇岛 (Java) 咖啡命名, 并得到大家认可, 于是 Oak 语言正式改名为 Java 语言, 图标也设计为冒着热气的咖啡。

随着 Internet 的发展, Java 语言和浏览器融合产生了 Applet 技术。目前, Applet 技术基本上已经被 Flash 技术替代, 但是, Applet 技术使 Sun 公司的该研发小组获得了新生。

Java 语言发展大事记:

1995 年 5 月, Sun 公司正式向外界发布 Java 语言, Java 语言正式诞生。

1996 年 1 月, JDK1.0 发布。

1997年2月, JDK1.1发布。

1998年12月JDK1.2发布。JDK 1.2的发布是语言发展的里程碑, Java首次被划分为J2SE、J2EE和J2ME。此后, Sun公司将Java改称Java2, Java语言也开始被国内开发者学习和使用。

2000年5月, JDK1.3发布。

2002年2月, JDK1.4发布。

2004年10月, JDK 1.5发布, 同时Sun公司将JDK1.5改称J2SE5.0.60, 同时, J2EE改称Java EE. 1256。

2006年6月, JDK 1.6发布, 也称作JavaSE. 6.0, 改称Java SE, J2ME改称Java ME。

2011年7月, JDK1.7发布。

1.1.2 Java 的特点及优势

Java是一种优秀的程序设计语言, 其最大的优点就是平台无关性, 相同的代码在Windows、Linux、Solaris、Mac OS等平台上都可运行, 从而实现“一次编写, 到处运行”。除此之外, Java还具有面向对象、可靠性、安全性、多线程等特性。

1. 平台无关性

Java的平台无关性是指用Java编写的应用程序不用修改即可在不同的软/硬件平台上运行, 即Java应用程序的运行不受平台约束, 一次编译, 可实现多平台运行。

2. 面向对象

面向对象编程思想是软件工程学的一次革命, 大大提升了人们的软件开发能力, 是软件工程学发展的里程碑。

Java语言使用类和对象的概念, 实现了对象的封装。类提供了一类相似对象实体的原型, 使用继承来实现子类 and 父类之间的联系, 并通过类的多态、抽象和接口等技术使程序易于扩充和维护。

3. 可靠性和安全性

Java的最初设计目的是应用于电子类消费产品, 因此要求有较高的可靠性。Java虽然源于C++, 但消除了许多C++的不可靠因素, 可以避免许多编程错误。由于Java主要用于网络应用程序开发, 因此对安全性也有较高的要求。Java通过自己的安全机制防止病毒程序的传播和下载程序对本地系统的威胁和破坏。

4. 多线程

Java从两方面支持多线程。一方面, Java环境本身是多线程的, 由若干个系统线程负责无用存储单元的回收、系统维护等系统级操作; 另一方面, Java语言内置多线程控制, 可以大大简化多线程应用程序的开发。



1.1.3 Java 的运行机制

Java 程序的运行必须经过编写、编译和运行三个步骤。其中，编写是指在 Java 开发环境中输入程序代码，形成扩展名为“.Java”的 Java 源文件。编译是指使用 Java 编译器对源文件进行错误排查的过程，编译后将生成扩展名为“.class”的字节码文件。运行是指使用 Java 虚拟机将字节码文件翻译成机器代码，执行并显示结果。

Java 虚拟机 (Java Virtual Machine, JVM) 是 Java 平台无关性的基础，JVM 中的 Java 解释器用于解释经 Java 编译器编译的字节码文件。Java 编程人员在编写完 Java 源文件后，通过 Java 编译器将 Java 源程序编译为字节码文件。机器只要安装了 Java 虚拟机，就可以运行该程序，与字节码文件在什么平台上生成无关。此外，Java 采用基于 IEEE 标准的数据类型，通过 JVM 保证数据类型的一致性和平台无关性。

字节码文件是一种与具体硬件环境及操作系统无关的中间代码，是一种二进制文件，是 Java 源文件由 Java 编译器编译后生成的目标代码文件。编程人员和计算机都无法直接读懂字节码文件，字节码文件必须由专用的 Java 虚拟机解释执行。

运行 Java 程序时，首先启动 JVM，然后由 JVM 负责解释执行 Java 字节码文件，并且 Java 字节码文件只能通过 JVM 运行。通过 JVM 即可把 Java 字节码文件和具体的硬件平台以及操作系统环境分隔，只要在不同的平台上安装了针对具体平台的 JVM，Java 程序就可以运行，而不用考虑当前具体的硬件平台及操作系统环境，也不用考虑字节码文件是在何种平台上生成的。JVM 把不同软硬件平台的具体差别隐藏起来，从而真正实现了二进制代码级的跨平台移植。

1.1.4 JDK 简介

Java 开发工具箱 (Java Development Kit, JDK) 是 Sun 公司免费为 Java 程序员提供的 Java 开发工具。JDK 通过命令行运行，主要包括以下常用工具。

(1) `javac.exe`: Java 程序编译器，用于将源代码编译成字节码文件，以“.class”为扩展名存入 Java 工作目录中。执行命令格式如下：

```
javac [选项] 文件名
```

(2) `java.exe`: Java 解释器，用于执行字节码文件。该命令中的类名参数必须是一个完整定义的名字 (包括该类所在包的包名、类名和包名之间的分隔符“.”)。执行命令格式如下：

```
java [选项] 类名 [程序参数]
```

(3) `javadoc.exe`: Java 文档生成器，用于根据 Java 源文件和包生成 HTML 格式的文档。

(4) `javap.exe`: Java 类分解器，对“.class”文件提供字节码的反汇编，并打印，默认情况下，打印类的公共域、方法、构造方法和静态初值。执行命令格式如下：

```
javap [选项] 类名
```

(5) `appletviewer.exe`: Java Applet 浏览器。执行命令格式如下：

```
appletviewer [-debug] URL
```

【任务实施】

1. 实施思路

- (1) 安装和配置 JDK;
- (2) 安装和配置 Eclipse。

2. 实施步骤

JDK 中包含了 Java 开发中必需的工具和 Java 程序的运行环境（即 JRE）。JDK 的安装文件可以从 <http://java.sun.com/javase/downloads/index.jsp> 下载，目前的 JDK 版本为 6.0，JDK 的安装文件有 3 个不同操作系统的版本，在这里使用 Windows 版本的 JDK，下载文件为“jdk-6u2-windows-i586-p.exe”。

如果需要 6.0 之前版本的 JDK，也可以在 Sun 的官方网站进行下载，具体的链接地址为 <http://java.sun.com/javase/downloads/previous.jsp>，在这个地址中提供了 JDK1.3 以后的各个版本 JDK 安装文件的下载服务。

下面对 JDK 的安装过程进行详细的说明：

(1) 双击运行下载的安装文件“jdk-6u2-windows-i586-p.exe”，就可以进入图 1-1 所示的 JDK 安装界面。

(2) 图 1-1 所示的安装 JDK 的欢迎界面在几秒钟之后会自动切换到图 1-2 所示的 JDK 安装协议选择界面。

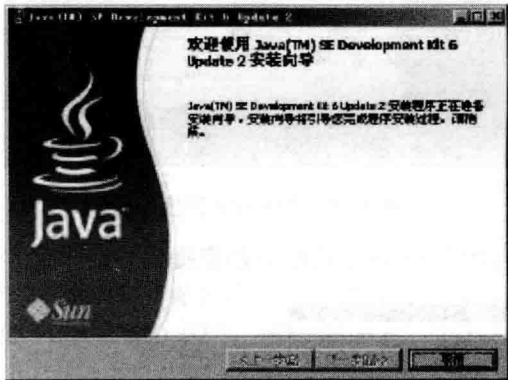


图 1-1 安装 JDK 的欢迎界面

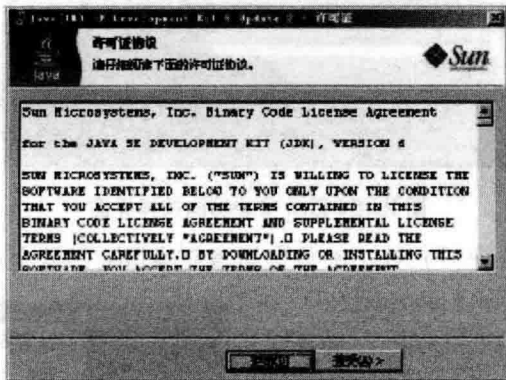


图 1-2 JDK 安装协议选择界面

(3) 在图 1-2 所示界面中，提供的是 JDK 的安装许可证协议信息，在这里只能选择“接受”，如果选择“拒绝”就会退出 JDK 的安装，在这里直接单击“接受”按钮，进入图 1-3 所示的 JDK 安装功能选择界面。

(4) 在图 1-3 所示界面中，可以选择需要安装的功能，在这里接受默认的安装即可，在默认的安装中已经提供了基本的 Java 开发和运行环境。单击“下一步”按钮就可以进入图 1-4 所示的 JDK 安装进度界面。

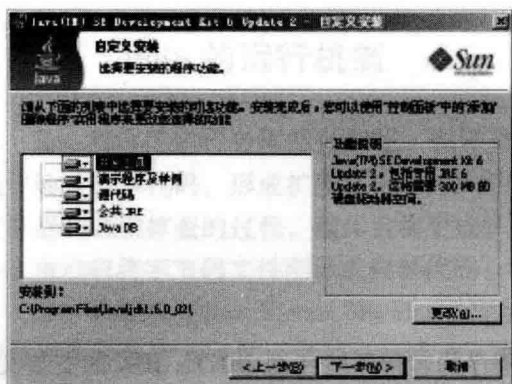


图 1-3 JDK 安装功能选择界面

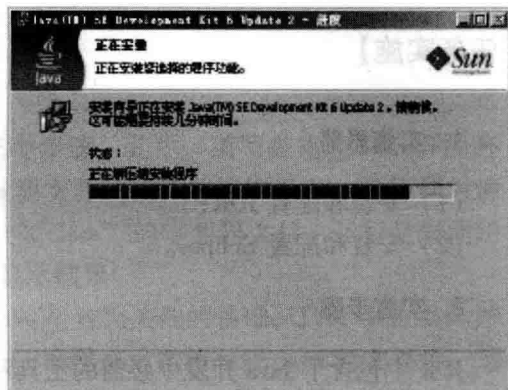


图 1-4 JDK 安装进度界面

(5) 在 JDK 安装进度结束以后，会自动进入图 1-5 所示的 JRE 安装功能选择界面，在 JDK 的默认安装中，会自动安装 JRE 的功能模块。

(6) 在图 1-5 所示的安装界面中，可以选择 JRE 的功能模块和安装位置，在这里选择默认的设置即可，单击“下一步”按钮，可以进入图 1-6 所示的 JRE 安装进度界面。

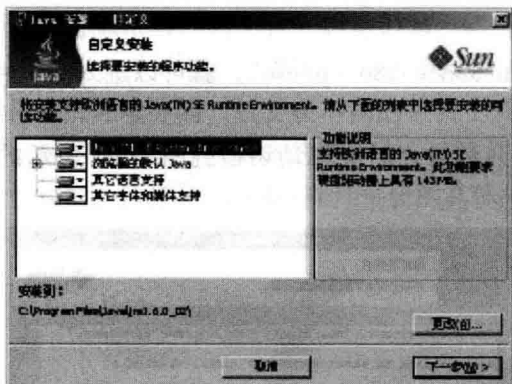


图 1-5 JRE 安装功能选择界面

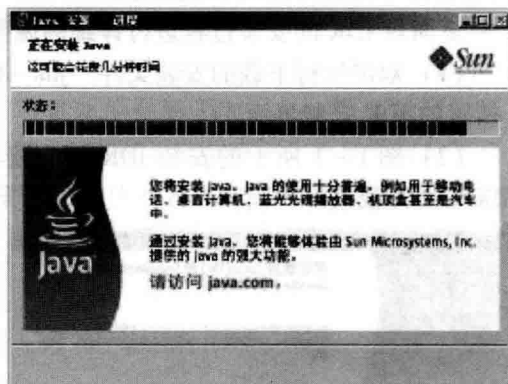


图 1-6 JRE 安装进度界面

(7) 在 JRE 安装进度结束以后，就会自动进入图 1-7 所示的结束信息提示界面。

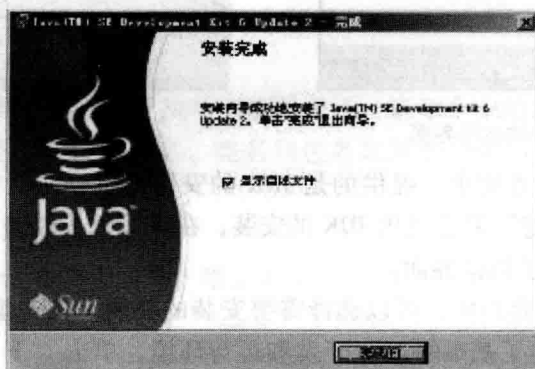


图 1-7 结束信息提示界面

在图 1-7 所示的界面中单击“完成”按钮，就可以完成 JDK 的安装。到这一步为止，JDK 和 JRE 的安装工作已经全部结束，但是现在还不能马上使用 JDK 中提供的开发工具，

JDK 安装结束之后，必需设置必要的环境变量才能正常使用开发工具。

(8) JDK 环境变量配置。

在 JDK 安装结束之后，必须进行环境变量的设置，才可以使用 JDK 提供的开发工具。下面对环境变量的设置步骤进行详细的介绍。

①用鼠标右键单击“我的电脑”图标，在弹出的菜单中选择“属性”选项可以打开“系统属性”窗口，选择“高级”|“环境变量”选项，可以得到图 1-8 所示的环境变量设置界面。

②在图 1-8 所示的界面中，可以对系统的环境变量进行设置，图中上半部分是用户变量界面，下半部分是系统变量界面，在系统变量界面中，选择“Path”这个变量，然后单击“编辑”按钮，就可以进入图 1-9 所示的“Path”变量编辑界面。



图 1-8 环境变量设置界面



图 1-9 “Path” 变量编辑界面

③在图 1-9 所示的界面中，可以修改“Path”变量的值，在这里需要在“Path”变量中添加 JDK 的路径信息，例如在安装 JDK 的时候，选择的安装路径为“C:\Program Files\Java\jdk 1.6.0_02”，所以需要在“Path”变量值的最前面添加“C:\Program Files\Java\jdk1.6.0_02\bin;”。需要注意的是，在引号的内容中，最后的逗号一定不能缺少。

经过上面的处理，JDK 的环境变量已经设置完成，下面需要测试 JDK 的安装配置是否成功。可以在 DOS 命令行中测试 JDK 是否安装成功。选择“开始”|“运行”命令，会弹出运行程序的选择界面，输入“cmd”并回车，就可以进入 DOS 界面，在这个界面中输入“java - version”，如果得到图 1-10 所示的界面，说明 JDK 的安装配置已经成功。

(9) 安装和配置 Eclipse。

访问“<http://www.eclipse.org/downloads/>”下载相关软件，解压缩之后，Eclipse 即可使用。在 Eclipse 安装目录下双击“eclipse”可执行文件即可启动 Eclipse。