



普通高等院校“十三五”规划教材

JAVA程序设计

JAVA CHENGXU SHEJI

韦鹏程 肖 丽 邹晓兵◎主编

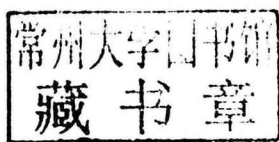


电子科技大学出版社

普通高等教育“十三五”规划教材

JAVA 程序设计

主 编 韦鹏程 肖 丽 邹晓兵
副主编 陈 宇 冉 维 吴小波 宋 科



电子工业出版社

图书在版编目 (CIP) 数据

Java 程序设计/韦鹏程, 肖丽, 邹晓兵主编. — 成都: 电子科技大学出版社, 2017. 6
ISBN 978-7-5647-4632-2

I. ①J… II. ①韦…②肖…③邹… III. ①JAVA 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2017) 第 134574 号

内容简介

Java 语言具有面向对象、与平台无关、简单稳定、支持多线程等特点, 已经成为当下最重要的网络编程语言之一。本书分为 12 个章节, 主要包括 Java 语言概述、基本语法规则、面向对象核心技术、数组和集合、异常处理、文件和数据流、图形用户界面、线程、网络编程和数据库程序设计。

本书以初学者为起点, 每个章节循序渐进地向读者介绍了 Java 相关核心知识, 并设计任务驱动环节, 使读者在读程序、学知识、写程序的过程中, 逐渐掌握面向对象的 Java 程序设计思想和技能。

本书适合作为高校计算机相关专业“Java 程序设计”或者“面向对象程序设计”课程的教材, 也可作为相关技术人员学习 Java 语言的参考用书。

Java 程序设计

主 编 韦鹏程 肖 丽 邹晓兵

出 版: 电子科技大学出版社 (成都市一环路东一段 159 号电子信息产业大厦 邮编: 610051)

策划编辑: 汤云辉

责任编辑: 汤云辉

主 页: www.uestcp.com.cn

电子邮箱: uestcp@uestcp.com.cn

发 行: 新华书店经销

印 刷: 北京市彩虹印刷有限责任公司

成品尺寸: 185mm×260mm 印张 21 字数 472 千字

版 次: 2017 年 6 月第一版

印 次: 2017 年 6 月第一次印刷

书 号: ISBN 978-7-5647-4632-2

定 价: 39.80 元

■ 版权所有 侵权必究 ■

◆ 本社发行部电话: 028—83202463; 本社邮购电话: 028—83201495。

◆ 本书如有缺页、破损、装订错误, 请寄回印刷厂调换。

前 言

Java 语言诞生于 20 世纪 90 年代初期，是一种完全的面向对象的程序设计语言，尤其适合用来编写跨平台应用软件。因此，伴随着计算机平台的多样化以及“互联网+”时代的来临，Java 语言已然成为主流的网络编程语言之一。其课程体系也成为各高校计算机相关专业的技术主线之一。

本书作者曾编写过彻底采用任务驱动教学模式的 Java 程序设计教材，希望通过各个任务的实施让读者轻松掌握 Java 程序设计思想和技能。从教材使用期间的教学反馈中发现，单纯的任务驱动教学模式并不是特别适合高校教育和初学者学习。因此，本书对内容体系和结构进行了调整，采用了“任务驱动+知识讲解”的综合模式。这种综合模式既对 Java 语言的核心技术知识进行了循序渐进的讲解，给予读者充分的理论知识储备，又通过任务实施强化了对学生的实践能力的培养，符合社会对应用型人才培养的需求。同时，综合模式下的学习可以为后续课程奠定面向对象程序设计的理论基础。

本书尽可能避免了对 Java 技术的贪多求全，在内容选取上做到了适度够用，重点讲解 Java 核心知识。本书共分 12 章。第一章讲解了 Java 语言概述，主要介绍了 Java 产生的背景，工作原理以及 Java 程序的一般开发过程。第二章则简要地介绍了 Java 的基本语法结构，这对于有程序设计基础的读者非常容易。第三章到第五章，从浅入深地介绍了面向对象技术，讲述了类和对象、继承和多态、接口以及访问权限等核心内容，使读者有了一个最基本的面向对象编程思想。第六章讲解 Java 中数组与集合的使用。第七章介绍了 Java 的异常处理机制。第八章介绍了 Java 独特的输入输出流技术。第九章简要介绍了已经不再流行的 Java GUI 图形界面编程，目的是让相关学习者了解 Java 的事件处理机制。第十章介绍了线程的相关知识和多线程编程。第十一章介绍了最基础的 Java 网络编程。第十二章主要讲解如何使用 Java 平台的 JDBC 工具包实现数据库程序设计。

作为计算机相关专业的教材，在本课程基础之上，还可以进一步学习 JSP 网络编程，JBuilder 数据库编程或者移动应用开发技术。

本教材由重庆第二师范学院的韦鹏程教授、肖丽博士和邹晓兵博士担任主编，由陈宇、冉维、吴小波、宋科等老师担任副主编，该教材得到重庆市交互式教育工程技术研究中心、重庆第二师范学院交互式儿童电子产品协同创新中心和重庆第二师范学校计算机科学与技术重点学科支持，在此表示感谢。

由于编者水平所限，书中难免还存在一些缺点和错误，希望读者批评指正。

目 录

第 1 章	Java 语言概述	1
1.1	Java 语言的发展	2
1.2	Java 程序运行过程	4
1.3	Java 程序开发过程	5
1.4	使用 Eclipse 开发 Java 程序	13
第 2 章	数据运算与流程控制	18
2.1	从圆形开始	18
2.2	变量和常量	20
2.3	数据类型及其运算	23
2.4	流程控制	35
第 3 章	类与对象	44
3.1	面向对象编程	44
3.2	类的描述	46
3.3	类与对象的创建	51
3.4	构造函数的使用	58
3.5	static 的合理应用	63
3.6	方法调用与重载	68
3.7	任务实现	73
第 4 章	继承、多态与内部类	79
4.1	组合	79
4.2	继承	82
4.3	多态	91
4.4	抽象类	98
4.5	接口	101
4.6	内部类	105
4.7	任务实现	110
第 5 章	访问权限	117
5.1	包	117
5.2	使用包	118
5.3	访问权限修饰词	119
5.4	任务实现	125
第 6 章	数组和集合	130
6.1	数组	130



6.2 集合	139
6.3 任务实现	158
第 7 章 Java 的异常处理	163
7.1 异常的概念	163
7.2 异常处理	167
7.3 创建用户自定义异常类	178
7.4 任务实现	180
第 8 章 输入输出流	182
8.1 File 类	182
8.2 字节流	185
8.3 字符流	190
8.4 对象流	195
8.5 任务实现	200
第 9 章 图形用户界面 (GUI)	215
9.1 GUI 编程基础	215
9.2 事件处理模型	216
9.3 常用组件与容器	217
9.4 布局管理	224
9.5 事件类型与适配器	227
9.6 任务实现	231
第 10 章 线程	240
10.1 线程概述	240
10.2 线程的创建	243
10.3 线程的控制	253
10.4 线程的同步	261
10.5 任务实现	271
第 11 章 网络编程	275
11.1 URL 类	275
11.2 InetAddress 类	279
11.3 Socket 编程	282
11.4 任务实现	287
第 12 章 数据库程序设计	298
12.1 JDBC 概述	298
12.2 JDBC API 介绍	299
12.3 JDBC 编程	305
12.4 任务实现	315
参考文献	329

第 1 章

Java 语言概述

Java 语言是一种面向对象的编程语言。它的语法结构与 C 语言和 C++ 语言非常类似，但是其语法结构更为简单，没有直接的指针操作，其内存自动管理。它提供了大量丰富的类库来满足网络化、多线程、面向对象系统的需要。Java 语言简单易学，安全稳定，具有通用性、平台无关性。

Java 作为最热门的编程语言之一，广泛应用于个人计算机、数据中心、游戏控制台、科学超级计算机、移动电话和互联网等领域。

通过最热门的 TIOBE 编程语言社区排行榜^①的数据分析，从 2009 年至今，Java 语言一直位于排行榜的前三位。Java 语言所在的甲骨文公司宣称全球有三十亿台设备运行该语言。



图 1-1 甲骨文公司的统计数量

^① TIOBE 编程语言社区排行榜按月更新，作为编程语言流行趋势的一个指标，其排名基于互联网上有经验的程序员、课程和第三方厂商的使用情况，排名数据来源于著名的搜索引擎（如 Google、MSN、Yahoo!、Wikipedia、YouTube 以及 Baidu 等）。

1.1 Java 语言的发展

1.1.1 Java 语言的出现

Java 是一种跨平台的、面向对象的程序设计语言，由当时任职 Sun 微系统的詹姆斯·高斯林（James Gosling）等人于 20 世纪 90 年代初开发。它最初被命名为 Oak，目标设置在家用电器等小型系统的编程语言，用来解决诸如电视机、电话、闹钟、烤面包机等家用电器的控制和通讯问题。

随着互联网的发展，Sun 看到了 Oak 在计算机网络上的广阔应用前景，改造了 Oak，并在 1995 年 5 月以“Java”名称正式发布。Java 的特点跨平台、动态 Web、Internet 计算等使得它伴随着互联网的迅猛发展而发展，逐渐成为重要的网络编程语言。Java 技术不断更新，自面世后就非常流行，发展迅速，在全球云计算和移动互联网的产业环境下，Java 更展示出显著优势和广阔前景。

1.1.2 Java 语言的发展

在这里，我们将 1995 年作为 Java 发展的起点。

1995 年：Java 诞生

1996 年：JDK^① 1.0

1997 年：JDK 1.1

1998 年：Java 2，JDK 1.2

1999 年：发布 Java 三个版本：标准版（J2SE）、企业版（J2EE）、微缩版（J2ME）^②

2000 年：Java 3，JDK 1.3

2001 年：Nokia 宣布到 2003 年将出售 1 亿部支持 Java 的手机；发布 J2EE 1.3

2002 年：Java 4，JDK 1.4

2004 年：Java 5，JDK 1.5

2005 年：SUN 公司宣布，Java 三个版本更名为 Java SE、Java EE 和 Java ME

2006 年：Java 6，JDK 1.6

2010 年：SUN 公司被 Oracle 公司收购

2011 年：Java 7，JDK 1.7

① JDK：Java Development Kit，Java 开发工具包。

② Java 三大平台。J2SE：（Java 2 Platform Standard Edition）Java SE 标准版，适合桌面应用；J2EE：（Java 2 Platform Enterprise Edition）Java EE 企业版，适合网络应用；J2ME：（Java 2 Platform Micro Edition）Java ME 微型版，适合手机应用。本书中的应用主要针对 Java SE 标准版。

2014 年: Java 8, JDK 1.8

Java 的每次更新, 都进行了不同程度地改进, Java 发展的主要推动力来自 JCP 制定的一系列 JSR^①。

1.1.3 Java 语言的特点

1. 简单易学 (Simple)

简单易学, 这是相对于其他流行的面向对象的编程语言而言的, 例如 C++。Java 语言以 C++ 语言为蓝本, 但是进行了大量的简化和改进, 例如 C++ 中非常复杂的指针和多重继承, 在 Java 中取消了指针, 使用接口替代了多重继承。Java 使用自动内存分配和垃圾收集, 而 C++ 需要程序员手动完成这些操作。所以有程序员把 Java 称为“C++-”。

Java 语言简单、语法清晰, 使用 Java 语言开发程序易写易读, 是设计和开发应用程序非常有效的语言。

2. 面向对象 (Object oriented)

面向对象的程序设计语言 (Object-oriented programming, 简称 OOP), Java 语言从设计时就被定位为 OOP, OOP 用对象来构建世界, 每样东西都可以作为一个对象, 例如圆是一个对象、人是一个对象。Java 语言就是通过创建对象、操作对象来工作的。Java 语言以对象为基本粒度, 其下包含属性和方法。对象的说明用属性表达, 而通过使用方法来操作这个对象。

程序开发中的核心问题之一是如何重复利用代码, OOP 通过封装很好地解决了这一问题。Java 作为一种面向对象的语言继承了面向对象的诸多好处, 如代码扩展、代码复用等。

3. 分布式 (Distributed)

分布式计算是指网络环境中的多台计算机协同工作。Java 语言使得分布式计算更加容易。Java 语言支持 Internet 应用的开发, Java 应用编程接口中有网络应用编程接口, 它提供了用于网络应用编程的类库。

Java 的远程方法调用 (Remote Method Invocation, 简称 RMI) 机制也是开发分布式应用的重要手段。使用这种机制, 某一台计算机上的对象可以调用另外一台计算机上的对象来获取远程数据。

4. 解释型 (Interpreted)

我们需要一个解释器来运行 Java 程序。因为 Java 程序是在 Java 虚拟机 (Java Virtual Machine, 简称 JVM) 中执行的, 执行的是编译之后的字节码。字节码是独立于机器平台的, 可以在任何装有 Java 解释器的机器上运行。

其他程序的编译器, 例如 C++, 将源程序编译为机器指令, 这些机器指令只能运行在本地机器上, 如果要在其他机器上运行, 则需要重新编译。例如在 Windows 系统中编译的 C++ 程序只能在 Windows 平台上运行。但是 Java 源程序编译之后生成的字

^① JCP (Java Community Process), 大量的公司与开发者共同参与的 Java 社区过程; JSR (Java Specification Requests), JCP 制定的不同的 Java 语言的规范, 通过这些规范来完善和增强 Java 语言。



字节码可以在任何安装有 Java 解释器的平台上运行, Java 解释器会将字节码翻译为目标机器可以识别的机器语言。

5. 可移植性 (Portable)

可移植性来自于解释性和 Java 体系结构的中立性, 另外, Java 系统本身也具有很强的可移植性, Java 语言自身是没有任何平台特征的。例如 Ada 语言在不同的平台上其整型变量的取值范围是不同的, 而 Java 语言在任何平台上都是一样的。

6. 多线程 (Multithreaded)

多线程在操作系统中已得到了最成功的应用。多线程是指允许一个应用程序同时存在两个或两个以上的线程, 用于支持事务并发和多任务处理。Java 除了内置的多线程技术之外, 还定义了一些类、方法来建立和管理用户定义的多线程。

7. 安全性 (Secure)

安全性可以分为四个层面, 即语言安全性、编译安全性、运行安全性、可执行代码安全性。语言安全性指 Java 的数据结构是完整的对象, 这些封装过的数据类型具有安全性。编译时要进行 Java 语言和语法的检查, 保证每个变量对应一个相应的值, 编译后生成 Java 类。运行时 Java 类需要类加载器载入, 并经由字节码校验器校验之后才可以运行。Java 类在网络上使用时, 对它的权限进行了设置, 保证了被访问用户的安全性。Java 对通过网络下载类具有一个安全防范机制, 如分配不同的名字空间以防替代本地的同名类、字节代码检查, 并提供安全管理机制让 Java 应用设置安全哨兵。

8. 丰富的类库 (Class and library)

Java 中丰富的类库资源是 Java 语言的一大特色, 是 Java 程序设计的基础。类库是 Java 语言提供的已经实现的标准类的集合, 是 Java 的应用程序接口 (Application Programming Interface, 简称 API^①)。在程序设计中, 合理和充分利用类库提供的类和接口, 不仅可以完成字符串处理、绘图、网络应用、数学计算等多方面的工作, 而且可以大大提高编程效率, 使程序简练、易懂。根据实现的功能不同, 可以划分为不同的集合, 每个集合组成一个包。Java 丰富的类库包括语言包、实用程序包、I/O 包、网络包、图形用户界面包等。

1.2 Java 程序运行过程

1.2.1 Java 的运行环境

Java 的运行环境 (Java Runtime Environment, 简称 JRE)。JRE 由 Java 虚拟机 (Java Virtual Machine, 简称 JVM) 和应用程序编程接口 API 组成。

JVM 是在某台计算机上由软件模拟出来的系统, 类似在各个平台上建立起一个虚

^① API, 应用程序编程接口 (Application Programming Interface), 是一些预先定义的函数, 目的是提供应用程序与开发人员基于某软件或硬件的以访问一组例程的能力, 而又无需访问源码, 或理解内部工作机制的细节。

拟系统。JVM 有自己的指令系统，它读取并处理经过编译之后的字节码文件，最终将字节码文件处理为 CPU 能识别的指令。不同的操作平台对应不同的虚拟机，例如 Windows、Unix、Dos 等操作平台上的虚拟机各不相同。

运行 Java 程序时，JVM 装载程序的 class 文件所使用的 Java API class 文件、所有被装载的 class 文件（包括从应用程序中和从 Java API 中提取的）和所有已经装载的动态库（包含本地方法）共同组成了在 Java 虚拟机上运行的整个程序。在一个平台能够支持 Java 程序以前，必须在这个特定平台上明确地实现 API 的功能。为访问主机上的本地资源，Java API 调用了本地方法。由于 Java API class 文件调用了本地方法，Java 程序就不需要再调用它们了。通过这种方法，Java API class 文件为底层主机提供了具有平台无关性、标准接口的 Java 程序。对 Java 程序而言，无论平台内部如何，Java API 都会有同样的表现和可预测的行为。正是由于在每个特定的主机平台上明确地实现了 Java 虚拟机和 Java API，因此，Java 程序自身就能够成为具有平台无关性的程序。

1.2.2 Java 的编译与运行

编写的 Java 源文件，源文件的后缀名必须为 “.Java”。通过编译器将其编译为字节码文件，字节码的扩展名为 “.class”。执行是字节码文件在任何带有 Java 虚拟机的平台上运行的过程。运行的过程就是解释的过程，解释是将字节码中的语言翻译为目标机器语言，翻译一步运行一步。具体过程如图 1-2 所示。

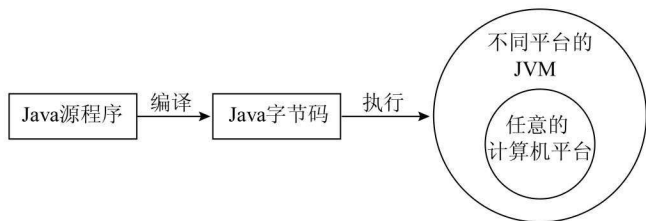


图 1-2 Java 程序的编译和运行过程

1.3 Java 程序开发过程

1.3.1 Java 开发环境搭建

1. 下载

进入 oracle 公司主页：<http://www.oracle.com/index.html>。在主页选择“Downloads” — “Java SE”。如图 1-3 所示。

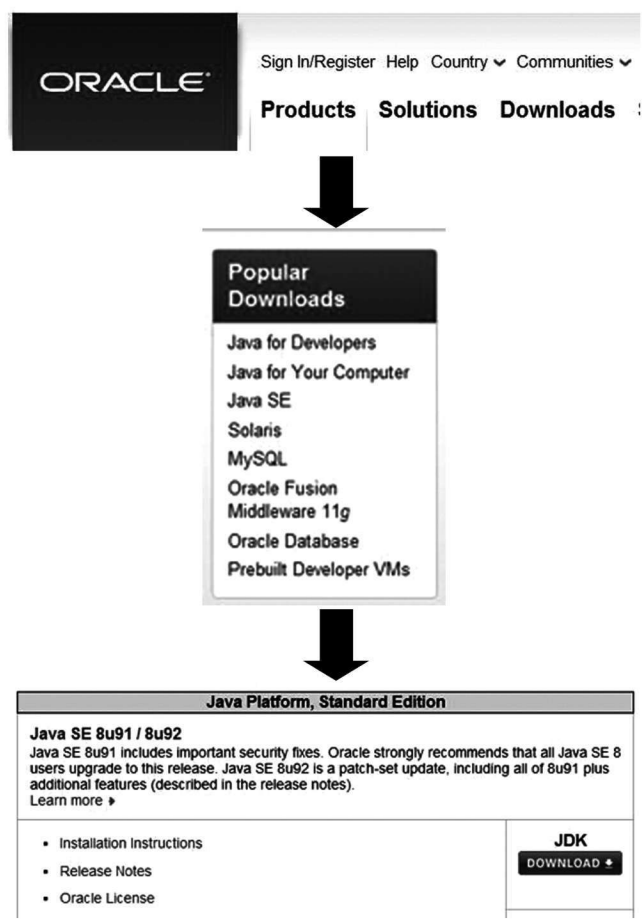


图 1-3 通过 oracle 公司主页进入 Java 下载页面

进入具体的下载页面之后，勾选“Accept License Agreement”，然后根据自己的操作系统选择相匹配的 JDK 版本。以 Windows 操作系统为例，当操作系统为 32 位时，选择 Windows X86 对应的版本；当操作系统为 64 位时，选择 Windows X64 对应的版本。如下图所示。

Java SE Development Kit 8u92		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
☒ Accept License Agreement ☐ Decline License Agreement		
Product / File Description	File Size	Download
Linux x86	160.26 MB	jdk-8u92-linux-i586.rpm
Linux x86	174.94 MB	jdk-8u92-linux-i586.tar.gz
Linux x64	158.27 MB	jdk-8u92-linux-x64.rpm
Linux x64	172.99 MB	jdk-8u92-linux-x64.tar.gz
Mac OS X	227.32 MB	jdk-8u92-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	139.47 MB	jdk-8u92-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	98.93 MB	jdk-8u92-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	140.35 MB	jdk-8u92-solaris-x64.tar.Z
Solaris x64	96.76 MB	jdk-8u92-solaris-x64.tar.gz
Windows x86	188.43 MB	jdk-8u92-windows-i586.exe
Windows x64	193.66 MB	jdk-8u92-windows-x64.exe

图 1-4 选择对应的版本

2. 安装

以下的安装步骤与变量配置以 64 位的 Windows7 操作系统环境为例。下载完毕后，双击安装包。



图 1-5 安装界面



图 1-6 选择安装路径



图 1-7 完成安装

3. 环境变量配置

在桌面的计算机图标上点击鼠标右键，选择“属性”。在弹出的面板中选择“高级系统设置”。



图 1-8 属性面板

在“系统属性”面板中选择“环境变量”。单击系统变量中的“新建”按钮。新建系统变量“Java_Home”，变量值为Java的安装路径，根据个人不同的安装情况路径会有所不同。

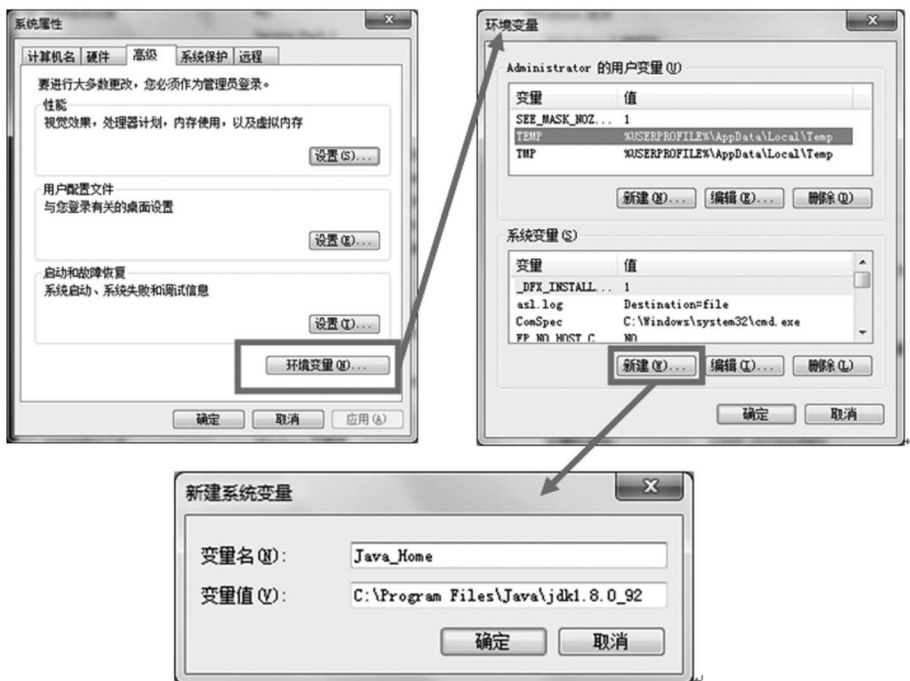


图 1-9 新建系统变量“Java_Home”

修改系统变量“Path”的变量值。回到环境变量面板中，选择“Path”变量，点击“编辑”，在弹出的对话框的最前面添加代码“%Java_Home%\bin;%Java_Home%\jre\bin;”。



图 1-10 修改系统变量 Path 的变量值

新建系统变量“ClassPath”，设置变量值为“.;%Java_Home%\bin;%Java_Home%\lib\dt.jar;%Java_Home%\lib\tools.jar”。



图 1-11 新建系统变量 ClassPath

检查环境变量是否搭建成功。打开“运行”对话框，输入“cmd”。

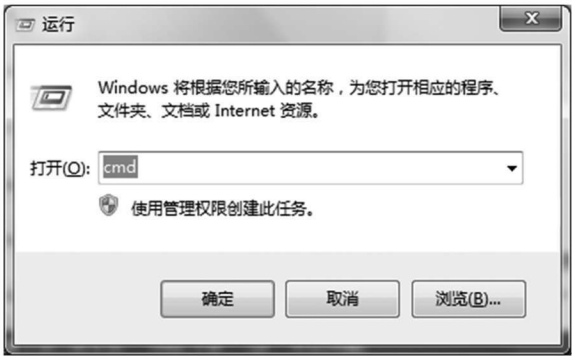


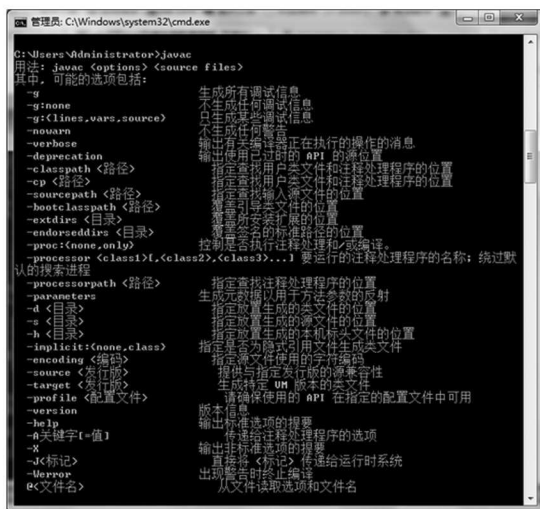
图 1-12 运行对话框

在弹出的对话框中，输入“Java”，如果配置成功，会出现下图中的内容。



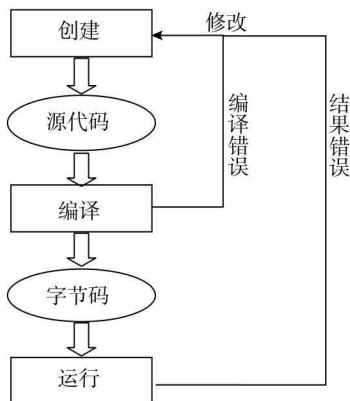
图 1-13 “Java”命令相关内容

继续输入“Javac”，如果环境变量配置成功，会有下图中的内容。



1.3.2 Java 程序开发流程

Java 程序的开发就是反复地创建和修改源代码、编译和执行的过程。如下图所示，如果编译的时候出现错误，我们要返回第一步修改源代码，然后继续编译它。如果运行结果出现错误或者结果不正确，我们也需要返回第一步修改源代码。



1.3.3 一个小程序

下面我们通过一个小程序的创建、编译和执行来说明该过程。

(1) 打开附件中的记事本工具，在其中输入下图中所示代码，请注意大小写。

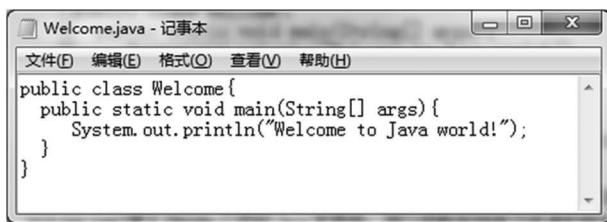


图 1-16 第一个程序

该程序的内容输出为“Welcome to Java world!”。



相关知识

保存的文件名必须与记事本中公用类的名称一致，扩展名必须为“Java”。例如本例中的公用类名为“Welcome”，所以该文件保存的名字为“Welcome.Java”。

(2) 打开开始菜单，在“运行”对话框中输入“cmd”并确定。



图 1-17 进入 command 面板

(3) 进入“Welcome.Java”程序所在的目录。在这里保存在“E: \ book”目录中。

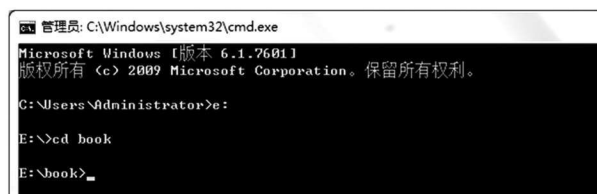


图 1-18 进入程序所在目录

(4) 输入编译命令：“Javac Welcome.Java”并回车。

```
E:\book>javac Welcome.java
```

图 1-19 编译程序

(5) 输入运行命令：“Java Welcome”并回车。显示出正确的运行结果“Welcome to Java world!”。

```
E:\book>java Welcome
Welcome to Java world!
```

图 1-20 运行程序