

EC

普通高校应用型本科电子与计算机系列规划教材

JAVA 程序设计

刘发久 编著

JAVA
Programming



大连理工大学出版社
Dalian University of Technology Press

EC

普通高校应用型本科电子与计算机系列规划教材

JAVA 程序设计

刘发久 编著

JAVA
Programming



大连理工大学出版社

图书在版编目(CIP)数据

JAVA 程序设计/刘发久编著. —大连:大连理工大学出版社, 2009. 8

(普通高校应用型本科电子与计算机系列规划教材)

ISBN 978-7-5611-5082-5

I. J… II. 刘… III. JAVA 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 150779 号

大连理工大学出版社出版

地址:大连市软件园路 80 号 邮政编码:116023

发行:0411-84708842 邮购:0411-84703636 传真:0411-84701466

E-mail: dulp@dulp.cn URL: <http://www.dulp.cn>

大连理工印刷有限公司印刷 大连理工大学出版社发行

幅面尺寸:185mm×240mm
2009 年 8 月第 1 版

印张:20 字数:426 千字
2009 年 8 月第 1 次印刷

责任编辑:王颖鑫

责任校对:苏丽丽

封面设计:季 强

ISBN 978-7-5611-5082-5

定价:38.00 元

出版说明

随着普通高等教育规模逐年扩大,我国高等教育已迅速进入大众化教育阶段,并成为全世界在校大学生规模最大的国家。在新时期,社会不但需要高校培养主要从事较高层次理论研究、设计和开发的学术型人才,而且更需要培养主要从事技术性和实用性工作的应用型人才。当前,许多用人单位对大学毕业生提出了专业脱离实际、上手慢、不好用的问题。因此,努力提高高校应用型本科人才培养质量,突出应用型本科人才培养特色,是一项意义重大的高等教育改革。在这项改革中,课程的教学改革是基础和关键,而对专业课程的传统教学内容进行改革,则显得尤为重要。

在大连理工大学的支持下,大连理工大学城市学院作为一所新型的独立学院,自建院时起就在各专业积极进行应用型人才培养模式改革,并以课程教学改革为主线,积极进行课程教学内容和教学方法的改革,通过多年来的课程教学改革实践,收到了比较好的效果。为了总结大连理工大学城市学院多年来在培养应用型人才中课程教学改革的实践成果和经验,大连理工大学出版社专门组织编写了这套电子与计算机类专业课程系列规划教材。

这套电子与计算机系列规划教材是由大连理工大学校部教师和城市学院教师共同编写完成的,他们都具有丰富的教学经验和较高的学术水平,以及比较丰富的专业实践经历,尤其更有在城市学院多年课程教学改革实践中积累的丰富教学经验,因此,这套教材可以说是应用型本科教学改革实践的丰硕成果。

这套教材的主要特色有两点,一是面向学生,二是联系实际,并体现在如下几方面:

以技术为基础 教材的编写主要建立在技术这个基点上,而不是理论的分析 and 研究,对于必需的理论,一般只给出或应用其结论,突出专业技术的学习。

以应用为目的 应用型人才培养理所应当要以应用为目的,在教材中突出专业技术的实际应用,使学生真正能够掌握专业技术之应用的真谛。专业课程也只有真正结合应用实际进行讲授,才能使学生真正理解和掌握。

内容安排突出重点 一门课程的教学内容很多,但基本的知识、概念是最主要的。抓住基本概念、基本公式、基本方法,围绕基本,提炼内容,突出重点,强化学生对基本知识的学习和掌握。

尽量降低学习难度 一本教材如果脱离学生的学习实际,那么教学的效果是不会理想的。内容循序渐进,前后衔接,推新温故,表述通俗易懂,深入浅出,是这套教材的编写原则。

这套教材是本社组织的第一套面向高校应用型本科教学的专业课程系列教材,希望这套教材能为应用型本科专业教学发挥积极的作用,同时请各位读者提出宝贵的意见。

联系电话:0411-84708947

信箱:yhl-0032029@163.com

大连理工大学出版社

2009年3月

前 言

Java 语言是一种多用途的面向对象的编程语言。本书注重理论联系实际,通过丰富的示例代码循序渐进地讲解面向对象的概念和编程思想,使读者在最短的时间内掌握 Java 程序设计思想和方法,是一本实用性很强的 Java 教材。

本书在编写过程中着重体现如下特色:

(1)注重 Java 编程的思想和方法,全面介绍基础知识的同时,强调编程的实战能力。

(2)既讲述了 Java 的语法知识,又讲述了一部分常用的程序设计方法以及设计模式,使读者对程序设计有整体的把握。

(3)例题选用经典、实用,试图用最精简的代码说明问题。这些代码既可以作为教学使用,又可以在实际工作时直接使用。

本书的主要内容有:Java 的一些常识性知识,使读者能基本了解“什么是 Java 程序设计语言”;Java 的基本语法知识,包括数据类型、运算符、循环语句、判断语句等,有 C 语言或 C++ 语言编程经验的读者可以跳过本内容;面向对象方面的知识,主要讲解类、接口等知识;常用基础类和工具类,包括字符串处理以及一些常用数据结构的现成类;文件处理以及输入输出流和序列化,使用这部分知识可以使数据长久保存在磁盘中;图形用户界面程序设计,包括窗体、对话框、菜单、鼠标等;数据库应用程序设计,在讲解数据库编程基础知识的同时,给出一个图书管理系统作为示例;网络程序设计,着重讲述网络套接字,并给出一个点对点的聊天程序作为示例;多线程知识,包括线程的创建、同步等;Java 在 Web 上的应用,简要介绍了 JSP 以及 Servlet,并给出一个留言板程序作为示例。

本书的所有代码均在 JDK 6.0 环境下调试通过,可以通过给 d_er_g@163.com 发邮件获取代码。

感谢大连理工大学唐志宏教授、孙承科教授和李彤教授在本书编写过程中的大力支持,感谢张治海老师为本书编写所做的大量工作和提出的宝贵意见,感谢所有在本书编写过程中给予帮助的同事及学生。

由于编者水平有限,仍可能有不足之处,敬请读者提出宝贵意见。

刘发久
2009 年 8 月

目 录

第 1 章 Java 语言概述	1	3.2 类定义	40
1.1 什么是 Java ?	1	3.3 类声明和类体	43
1.2 Java 语言	2	3.3.1 变量声明	43
1.3 Java 开发平台	4	3.3.2 方法定义	44
1.4 Java 程序的编写、编译和 运行过程	6	3.3.3 自定义构造器	47
1.5 Java 语言程序的注释	8	3.4 用类去创建对象	48
1.6 跨平台	9	3.5 继承一个类	50
1.7 面向对象的程序设计	10	3.6 接口	55
1.8 实例讲解与问题研讨	12	3.7 抽象类与最终类	59
本章小结	15	3.8 多态	61
习 题	15	3.9 内部类和匿名类	63
第 2 章 Java 语言基础	17	3.10 泛型	66
2.1 基本符号	17	3.11 修饰符	67
2.2 数据类型和变量	19	3.12 实例讲解与问题研讨	72
2.2.1 数据类型	19	本章小结	77
2.2.2 变量	20	习 题	78
2.3 运算符和表达式	21	第 4 章 常用基础类和工具类	80
2.4 语句和方法	23	4.1 Object 类	80
2.4.1 语句	23	4.2 Number 类	81
2.4.2 方法	24	4.3 StringBuffer 类	86
2.5 控制语句	26	4.4 Exception 类	88
2.5.1 条件语句	26	4.5 Scanner 类	92
2.5.2 循环语句	28	4.6 StringTokenizer 类	93
2.5.3 跳转语句	30	4.7 Vector<E> 泛型类	96
2.5.4 异常处理语句	31	4.8 LinkedList<E> 泛型类	97
2.6 数组	31	4.9 ArrayList<E> 泛型类	100
2.7 实例讲解与问题研讨	34	4.10 HashMap<K,V> 泛型类	103
本章小结	36	4.11 枚举和迭代器	105
习 题	36	4.12 实例讲解与问题研讨	106
第 3 章 Java 面向对象基础	38	本章小结	107
3.1 引言	38	习 题	108

第 5 章 I/O 流和文件管理	110	7.3 Java 数据库管理程序设计	218
5.1 I/O 流类	110	7.4 数据库的数据输入和 打印 GUI 程序设计	219
5.2 File 类	111	7.5 实例讲解与问题研讨	224
5.3 FileWriter 类	111	本章小结	247
5.4 FileReader 类	113	习 题	248
5.5 InputStreamReader 类	116	第 8 章 网络程序设计基础	250
5.6 BufferedReader 类	116	8.1 Java 网络类和接口	250
5.7 PrintWriter 类	117	8.2 InetAddress 类	251
5.8 读取键盘输入的数据	118	8.3 URL 类和 URLConnection 类	254
5.9 JAR 文件	118	8.4 Socket 类和 ServerSocket 类	256
5.10 实例讲解与问题研讨	120	8.5 DatagramPacket 类和 DatagramSocket 类	264
本章小结	122	8.6 MulticastSocket 类	267
习 题	123	8.7 实例讲解与问题研讨	268
第 6 章 图形用户界面程序设计	124	本章小结	270
6.1 视窗类	124	习 题	270
6.2 对话框	129	第 9 章 多线程	272
6.3 组件类	134	9.1 进程与线程概述	272
6.3.1 面板和内部视窗	135	9.2 Java 的进程模型	274
6.3.2 标签、按钮和工具栏	137	9.3 线程类 Thread	275
6.3.3 菜单条、菜单和菜单项	149	9.4 线程接口 Runnable	277
6.3.4 文本行和文本区	151	9.5 线程的生命周期	278
6.3.5 表格和窗格滚动条	160	9.6 线程同步	281
6.3.6 树和窗格拆分	162	9.7 定时器的管理	284
6.3.7 内部视窗和分层容器	166	9.8 实例讲解与问题研讨	287
6.3.8 进度条类	169	本章小结	291
6.4 布局管理类	172	习 题	292
6.5 事件处理	177	第 10 章 Java 在 Web 上的应用	293
6.5.1 事件处理机制	178	10.1 Applet	293
6.5.2 API 中的事件类	181	10.2 Servlet	301
6.5.3 事件监听器接口	182	10.3 JSP	305
6.5.4 常见事件处理	185	10.4 实例讲解与问题研讨	309
6.6 实例讲解与问题研讨	197	本章小结	310
本章小结	209	习 题	311
习 题	210	参考文献	312
第 7 章 数据库应用程序设计	211		
7.1 数据库和 SQL 语句	211		
7.2 JDBC	214		

第 1 章 Java 语言概述

道可道,非常道;名可名,非常名。

——老子

本章概括性地讨论一些有关 Java 的基本问题和面向对象的概念,以便尽快认识 Java 和编写 Java 程序。

1.1 什么是 Java ?

Java 包含两方面内容,一个是 Java 语言,Java 语言是一种多用途的面向对象的编程语言;另一个是 Java 平台,Java 平台是支撑 Java 语言程序开发和运行的环境,包括 Java 虚拟机 JVM 和编程接口 API。

Java 是由 James Gosling 等人创立的,他们创造了 Java 语言及早期的 Java 平台。Sun 公司在 1995 年 5 月 23 日正式宣布了 Java 的诞生。Java 名字来自于印尼的一个盛产咖啡小岛的名字,中文称为爪哇,取名为 Java,寓意为世人献上一杯浓香的热咖啡。

Java 的最初推动力是源自对独立于平台语言的需要,这种语言可创建能够嵌入微波炉、遥控器等各种家用电器设备的软件。用做控制器的 CPU 芯片是多种多样的,但 C 和 C++ 以及其他绝大多数语言的缺点是只能对特定目标进行编译。尽管为任何类型的 CPU 芯片编译 C++ 程序是可能的,但这样做需要一个完整的以该 CPU 为目标的 C++ 编译器,而创建编译器是一项既耗资巨大又耗时较长的工作。因此需要一种简单且经济的解决方案。为了找到这样一种方案,Gosling 等人开始一起致力于开发一种可移植、跨平台的语言,该语言能够生成运行于不同环境、不同 CPU 芯片上的代码。他们的努力最终促成了 Java 的诞生。随着万维网 Web 的出现,Java 被推到计算机语言设计的最前沿,因为 Web 也需要可移植的程序,Web 的迅猛发展导致了 Java 的迅猛发展。

在 1998 年之前,Java 主要用在客户端以及一些移动设备中。1998 年 12 月,Sun 公司发布了 Java 1.2 版本,自 Java 1.2 之后的版本统称为 Java 2。1999 年 6 月,Sun 公司发布了 Java 2 的三个版本:标准版 J2SE、企业版 J2EE 和微型版 J2ME。J2SE 主要用于桌面系统开发,即开发 Java application 程序和 Java applet 程序,是其他两个版本的基础;J2EE 主要用于 Web 开发,即开发 Java Servlets、JSP 和 EJB 程序;J2ME 主要用于手机和 PDA 开发,即

开发 MIDP 应用程序。

关于计算机语言的发展史,一般认为 C 语言是具有里程碑意义的计算机语言,C 语言演变出了 C++ 语言,而 C++ 语言会让位给 Java 语言。Java 继承了其他语言的先进原理,同时又因其独特的环境要求而提出了一些创新性的概念,Java 也改变了人们的编程方式,特别是 Java 对 C++ 使用的面向对象范例的增强和完善。

Java 是一种通用的、面向对象的、跨平台的、并发的和强类型的编程语言,这些特点都会在后续章节逐步讲到。

Java 被广泛地应用到诸如移动通信、智能卡、ATM 机、个人电脑、服务器和大型主机等设备上。目前,Java 已成为全球最具影响力的编程语言和开发平台,Java 全球经济价值已经超过千亿美元。

1.2 Java 语言

Java 语言与其他程序设计高级语言(如 C 语言、C++ 等)稍有一点相似,更有很大的不同。

Java 语言既是编译的又是解释的,编译和解释是分别进行的,Java 平台提供了相应的 Java 编译器和解释器,如图 1-1 所示。首先把用 Java 语言编写的程序保存为 .java 文件,编译器负责把 .java 文件编译成字节码 .class 文件,字节码是 Java 虚拟机可以识别和执行的高度优化的指令集,字节码与具体的计算机无关,称为跨平台,跨平台是 Java 适用 Web 开发的主要原因,也是创立 Java 的起因。每次运行程序时,解释器负责把 Java 字节码文件加载到内存,并解释为具体的计算机上能够执行的程序。编译只需要进行一次,而解释是每执行一次解释一次。

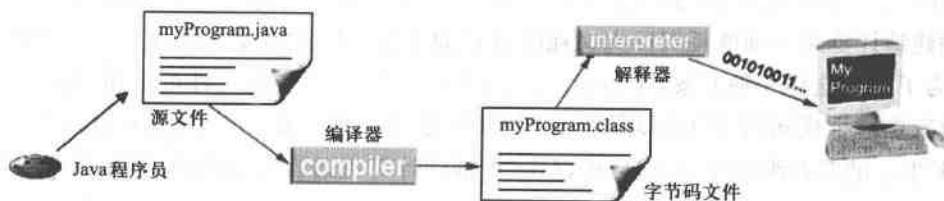


图 1-1 Java 语言程序的编译和解释过程

除了跨平台外,Java 语言是纯面向对象的并且具有强大的 API(Application Programming Interface)支撑,API 是由 Java 平台提供的重要开发资源。

Java 语言能干什么?用 Java 语言结合 Java API 可以写出各种形式的 Java 程序,见表 1-1。

表 1-1 Java 语言的应用

程序形式	编译环境	运行环境	应用
Java application	Java 编译器	Java 虚拟机(JVM)	C/S 架构
Java applet	Java 编译器	Java 的浏览器	嵌入 HTML 中
Java Servlets	Java 编译器	Java 的 Web 服务器	J2EE 架构
JSP(Java Server Page)	脚本语言(无需编译)	Java 的 Web 服务器	嵌入 HTML 中

本书重点讨论 Java application 程序,它是最基本的 Java 语言程序,后面的程序若不声明都是指 Java application 程序,对其他形式 Java 程序都有专著介绍,本书只做一些简单介绍,感兴趣的读者可以参阅相关书籍。

什么是 Java 程序?下面通过一个简单示例,认识一下 Java 程序。

【例 1-1】 这是一个最简单的程序,用 class 定义一个类 HelloWorld,在这个类中定义了一个 main()方法,在这个 main 方法中,用标准的输出方法 System.out.println(),在命令提示符视窗中输出一行文字“Hello World!”,程序如下:

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

运行结果如图 1-2 所示。



图 1-2 例 1-1 的运行结果

关于例 1-1 的几点说明:

①Java 程序是由类 class 构成,class HelloWorld{...}是声明一个类, class 是声明类的关键字, HelloWorld 是类的名字,花括号“{ }”和其中的内容称为类体;HelloWorld 类体从第一行的“{”开始,到最后一行的“}”结束;两个反斜线“//”后面的文字是注释,注释是对代码进行解释和说明,不影响程序的编译和运行。

②类体中的 public static void main(String args[]){...}是定义 main()方法,(String args[])是该方法要求的格式,从第二行的“{”开始,到第四行的“}”结束,{...}称为方法体,方法体中可以包含语句,Java 语言中的方法类似于 C 语言中的函数,main 方法类似于 C 语言中的 main 函数,相同之处都是程序从这里开始执行,不同之处是 C 语言中的 main 函数

由操作系统调用,Java 语言中的 main 方法由虚拟机调用,一个能够执行的程序就必须要有 main 方法。

③方法体中的 `System.out.println("Hello World!");` 是一个语句,这个语句在命令提示符视窗中输出圆括号“()”里的内容。Java 程序是由类组成,类是由变量和方法组成,方法是由语句组成,关于类、变量、方法和语句将在后续章节详细讨论。

要编译和运行例 1-1 这个程序,首先要在计算机上构建 Java 开发平台。

1.3 Java 开发平台

Java 开发平台是 Java 的一个重要组成部分,是一种能够运行 Java 程序并且支撑 Java 程序开发的软件系统,包括 Java 虚拟机和 Java API 两部分,如图 1-3 所示。

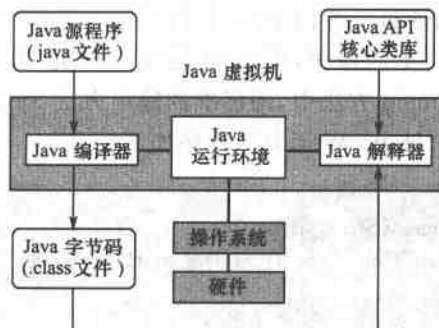


图 1-3 Java 开发平台结构图

JDK 6 (Java™ 2 Platform Standard Edition 6) 是 Java 开发平台的一种具体实现,主要内容见表 1-2。

表 1-2 Java 开发平台

英文名称	英文缩写	中文名称
Java™ SE Development Kit 6	JDK™ 6	Java 开发工具
Java™ SE Runtime Environment 6	JRE™ 6	Java 运行环境

JDK 6 可以从 <http://java.sun.com> 上免费下载,安装方法也可以同时查阅到。

在 Windows 操作系统下安装 JDK,要确定一个安装目录,例如 `d:\jdk`,JDK 安装后的目录结构如图 1-4 所示。

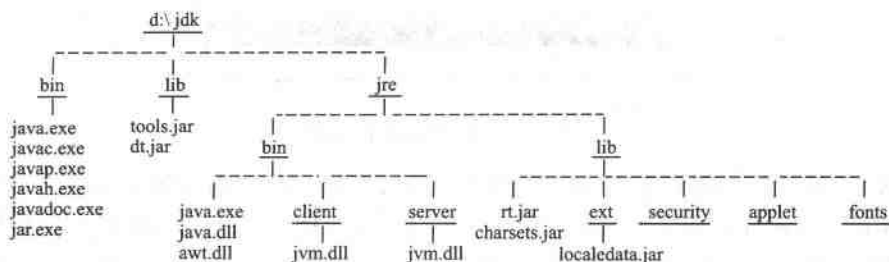


图 1-4 JDK 目录结构

bin 目录保存了 javac.exe、java.exe 等 DOS 下的可执行文件,其中 javac.exe 是编译器,用来将程序文件中的每个类定义编译成一个.class 文件,java.exe 是 Java 虚拟机上的解释器,负责加载.class 文件并将.class 文件解释成为特定的机器码来执行,javadoc.exe 用来将源程序中的文档注释内容生成 HTML 格式的程序说明文档。

lib 目录保存了 tools.jar 和 dt.jar 文件,是开发工具的支持文件。

jre 目录下包括 bin 和 lib 两个子目录,jre\bin 保存了 Java 运行环境,jre\lib 保存了 rt.jar 文件,其中包含了 Java API 核心类的基本包 java 和扩展包 javax,包是存放一些相关类的一个文件目录。

Java API 是 Sun 的核心类库,为 Java 程序设计提供了强大的支撑,下面通过一个示例说明 API 核心类库的用途和用法,这个示例程序创建一个视窗,首先用 import 语句从 API 核心类库中引进视窗类 JFrame,然后声明一个 MyWindow 类,在 MyWindow 类中定义一个 main() 方法,在 main() 方法中用 JFrame 类声明并创建一个视窗对象 win,最后用视窗对象 win 的方法设置视窗的可见性和大小,视窗对象 win 调用的方法是在视窗类 JFrame 中定义的,JFrame 类是 API 提供的,学会使用 API 中的类,是学习 Java 的重要内容。

【例 1-2】 创建一个大家经常见到的视窗对象。

```
import javax.swing.JFrame; //这是从 API 中引进视窗类 JFrame
class MyWindow{
    public static void main(String args[]){
        JFrame win=new JFrame("本窗体就是一个对象!");
        //用 JFrame 类先创建一个视窗对象 win
        win.setVisible(true); //然后设置视窗对象 win 可见
        win.setSize(400,60); //然后设置视窗对象 win 的大小
    }
}
```

运行结果如图 1-5 所示。



图 1-5 例 1-2 的运行结果

在例 1-2 程序中,可以把 JFrame 类看做一种数据类型,把对象 win 看做一个变量,把 `JFrame win=new JFrame("本窗体就是一个对象!");` 比做 `int x=10;` `int` 是一种数据类型, `x` 是一个变量,区别是 win 可以调用方法,例如 `win.setSize(400,60);` 就是用 win 调用 `setSize(400,60)` 方法, `void setSize(int width,int height)` 方法是在 JFrame 类定义的众多方法之一。

关于 Java 开发平台的详细内容,请参阅 Sun 的白皮书或相关资料。

1.4 Java 程序的编写、编译和运行过程

可以采用任何一种文本编辑器来写 Java 程序,比如 Windows 操作系统附件中的记事本或集成开发工具 JCreator、MyEclips 等。建议初学者先用记事本写程序,体会一下程序编写、编译和运行过程之后,再使用集成开发工具。

要建立一个保存程序文件的目录,例如 `d:\mywork`,由于程序文件目录与 JDK 安装目录不同,为了使用 DOS 命令方便,可以通过电脑属性设置系统变量 `path` 指向 `javac.exe` 和 `java.exe` 所在路径 `..\bin`,建立系统变量 `classpath` 指向要运行的 `.class` 文件所在路径。例如, `d:\mywork`,如图 1-6 所示。

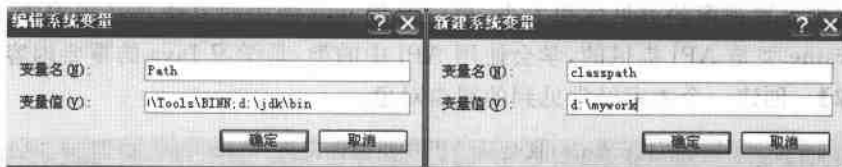


图 1-6 通过电脑属性设置系统变量

也可以在 DOS 视窗中用 DOS 命令 `set` 临时设置系统变量 `path` 和系统变量 `classpath`,如图 1-7 所示。



图 1-7 在 DOS 视窗临时设置系统变量

(1) 编写 Java 程序, 保存程序文件

打开 Windows 操作系统附件中的记事本, 用记事本编写 HelloWorld 程序, 如图 1-8 所示。

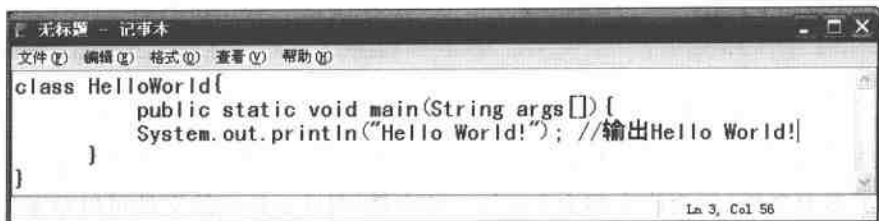


图 1-8 用记事本编写 HelloWorld 程序

在目录 d:\mywork, 保存程序文件 HelloWorld.java (后缀是 .java, C++ 语言程序后缀是 .cpp), 如图 1-9 所示。



图 1-9 保存源程序文件

(2) 编译 Java 程序文件

打开 Windows 操作系统附件中的 DOS 命令视窗, 用编译器 javac.exe 编译 Java 程序 HelloWorld.java, 具体操作过程如图 1-10 所示。

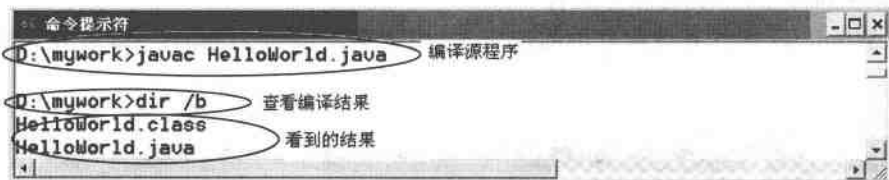


图 1-10 用编译器 javac.exe 编译源文件 HelloWorld.java

编译结果会产生一个 HelloWorld.class 的文件,.class 文件是字节码文件,Java 程序文件中定义了多少个类,就会产生多少个.class 的文件,每个.class 的文件可以单独使用,而 C++ 程序的.cpp 文件不管定义了多少个类,只产生一个.obj 文件,要单独使用一个类必须单独写一个头文件。与 C++ 程序不同,Java 程序不允许在类之外声明变量或定义方法,所以说 Java 程序纯粹是类的或者说 Java 程序是纯面向对象的。

(3) 运行.class 文件

用解释器 java.exe 加载和运行.class 文件 HelloWorld.class,如图 1-11 所示。

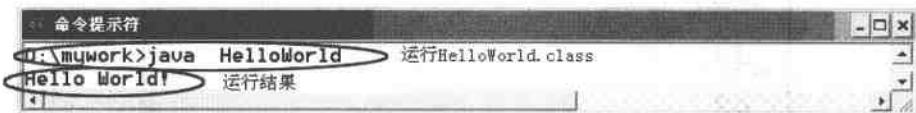


图 1-11 加载和运行.class 文件

1.5 Java 语言程序的注释

注释能够提高程序的可阅读性,编译器不理睬注释,不影响程序编译和运行,Java 程序注释分为单行注释、多行注释和文档注释。

1. 单行注释

单行注释为代码实现的功能添加一个简短的说明,“//”表示注释开始,注释内容到本行结尾,例如:

```
win.setVisible(true); // 设置视图对象 win 可见  
win.setSize(400,60); // 设置视图对象 win 的大小
```

2. 多行注释

使用多行注释(又称区域注释),通常使用区域注释描述文件、数据结构、方法和文件说明,它们通常放在文件的开头和方法中,多行注释以“/*”开头,以“*/”结尾。例如:


```
/* 我们首先用 import 引进 API 核心类库中的一个视图类 Frame, 然后定义一个类 MyWindow, 在类 MyWindow 中定义了一个 main 方法 */
```

3. 文档注释

(1) 文档注释用于描述 Java 的类、接口、构造器、方法, 文档注释是 Java 语言特有的。javadoc 可以把文档注释内容转换成 HTML 网页格式的程序说明文档。文档注释以 “/* *” 开头, 以 “*/” 结尾。例如:

```
/* * 在 main 方法中用 Frame 类声明并创建了一个视图对象 win, 最后用视图对象 win 自身的方法设置了视图的可见性和大小 */
```

(2) javadoc 的使用方法

功能: 把程序中的文档注释内容, 生成 Java 类的说明文档。

用法: javadoc[选项] 源程序文件名, 其中[选项]主要包括:

- d 指定生成的文档存放的位置
- public 仅显示公共类和成员
- protected 显示受保护/公共类和成员(默认)
- package 显示软件包/受保护/公共类和成员
- private 显示所有类和成员
- help 显示命令行选项并退出

【例 1-3】 把例 1-2 中的文档注释内容生成 HTML 格式的说明文档, 操作过程如图 1-12 中第一行所示, 图 1-12 中第二行开始是系统提示信息, 可以在 javadoc 创建的 win\ 目录下找到这些文档。

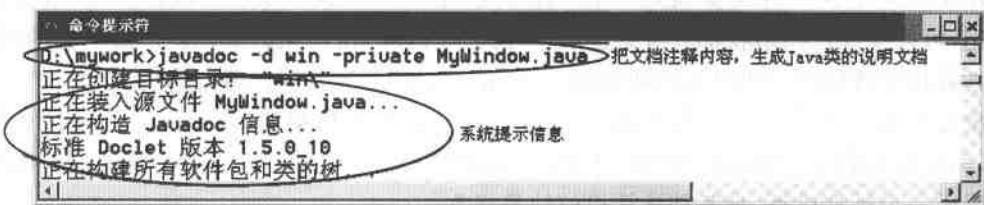


图 1-12 例 1-2 中的文档注释内容生成 HTML 格式的说明文档

1.6 跨平台

跨平台指的是 Java 程序编译后的 .class 文件可以在任何具有 Java 虚拟机的计算机或电子设备上运行, 同一个 .class 文件可以运行在 Windows、Solaris 和 Macintosh 等不同的平台上, 如图 1-13 所示。