

JAVA开发



实用技术

JAVA KAIFA
SHIYONG JISHU

主 编 / 梁勇强 蒙峭缘

层次递进 任务驱动
案例经典 图文并茂

JAVA开发



实用技术

JAVA KAIFA
SHIYONG JISHU

主 编 / 梁勇强	蒙峭缘
副主编 / 肖志军	李治强
编 委 / 张远夏	龙法宁
孙小雁	李超建

图书在版编目 (C I P) 数据

JAVA 开发实用技术 / 朱晓姝总主编; 梁勇强, 蒙峭缘主编. —成都: 西南交通大学出版社, 2017.2
普通高等学校“十三五”应用型人才培养规划教材
ISBN 978-7-5643-5268-4

I. ①J… II. ①朱… ②梁… ③蒙… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2017) 第 025834 号

普通高等学校“十三五”应用型人才培养规划教材

JAVA 开发实用技术

总主编 朱晓姝
主 编 梁勇强 蒙峭缘

责任编辑	穆 丰
封面设计	墨创文化
出版发行	西南交通大学出版社 (四川省成都市二环路北一段 111 号 西南交通大学创新大厦 21 楼)
发行部电话	028-87600564 028-87600533
邮政编码	610031
网 址	http://www.xnjdcbs.com
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	13.5
字 数	320 千
版 次	2017 年 2 月第 1 版
印 次	2017 年 2 月第 1 次
书 号	ISBN 978-7-5643-5268-4
定 价	32.00 元

课件咨询电话: 028-87600533

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 语言自诞生之日起就被称为一种革命性的程序设计语言。它具有简单、面向对象、平台无关、多线程、安全、动态等优点,是最流行的编程语言之一,同时也是大专院校面向对象程序设计课程教学的首选语言。

目前,介绍 JAVA 程序设计的教程类书籍很多,但是这些教程类书籍不同程度地存在下面的缺点:(1) 教学内容照顾面过宽,缺乏针对性。考虑到不同院校的不同需求,目前的教程类书籍内容都尽量地全面,这样做的好处很明显,就是任课教师可以在教学中根据实际需要灵活选择教学内容。但是不足也很明显,任课教师未必能够完整领会编写者的意图,最后选择而拼凑的教学内容连贯性、完整性可能不好;(2) 教学案例实际意义不大,相关内容的教学案例没有关联。教学案例的实用性弱,往往迫使任课教师再花时间去说明知识点的实用性,教学案例没有体现与知识点的联系,教师就需要花费一部分时间先让学生理解不同案例相关的背景知识,学生也不能通过案例更直观地理解知识点。

编写本书的目的是为地方本科院校计算机相关专业的 JAVA 面向对象程序设计课程教学提供一本针对性的教程:(1) 内容选择方面。在考虑相对完整性的基础上,精选学生今后从事 JAVA 程序设计最基础、最常用的知识和技术;(2) 案例设计方面。尽量选择与学生知识背景相关的案例,相关联的知识点尽量使用关联的案例,或者说是同一案例的不同版本。全书共分为 10 章,其中第 1 章介绍 JAVA 语言产生和发展的历史,介绍 JAVA 开发工具包的组成,还有 JAVA 应用程序的类型和基本的开发步骤;第 2 章介绍 JAVA 的基本数据类型、数组、枚举,包括标示符和关键字、基本数据类型、运算符和表达式等;第 3 章介绍 JAVA 的结构化程序设计;第 4 章和第 5 章介绍 JAVA 的面向对象程序设计,其中第 4 章介绍 JAVA 中的类和对象,第 5 章介绍 JAVA 中的继承、接口等机制;第 6 章介绍 JAVA 中的常用实用类;第 7 章介绍 JAVA 的文件操作;第 8 章介绍 JAVA 的图形用户界面设计;第 9 章介绍 JAVA 中的多线程;第 10 章介绍 JAVA 操作数据库的方法。如

果读者之前已经学习了 C 语言程序设计，可以跳过本书的第 2 章到第 3 章。

本书由长期从事 JAVA 面向对象程序设计教学的一线教师编写，其中李治强老师编写了第 1 章和第 2 章，肖志军老师编写了第 3 章，梁勇强老师编写了第 4 章和第 10 章，蒙峭缘老师编写了第 5 章至第 8 章，张远夏老师编写了第 6 章，龙法宁老师编写了第 9 章。全书由孙小雁、李超建、张捷三位老师校对。

本书编写工作得以顺利完成，除了编写者的辛勤劳动之外，还得到了许多部门和领导的大力支持和帮助，热心的教师也提出了不少宝贵意见，在这里表示衷心的感谢！

由于编者的水平有限，书中的不足之处难免，敬请各位热心的专家和读者批评指正！

本书编写组

2016 年 11 月

目 录

第1章 概 述	1
1.1 JAVA 语言的产生	1
1.2 JAVA 的特性	1
1.3 JAVA 的平台简介	3
1.4 JAVA 程序的开发过程	3
1.5 Application 和 Applet 程序	7
1.6 JAVA 程序的注释	10
1.7 本章小结	10
第2章 基本数据类型、数组、枚举	13
2.1 程序分析	13
2.2 常量与变量	14
2.3 标识符与关键字	14
2.4 数据类型	15
2.5 数组与方法	20
2.6 枚举类型	26
2.7 本章小结	28
第3章 JAVA 的结构化程序设计	30
3.1 顺序结构	30
3.2 选择结构	32
3.3 循环结构	39
3.4 break 语句	46
3.5 continue 语句	47
3.6 返回语句 return	48
3.7 本章小结	49

第4章 类与对象	51
4.1 面向对象程序设计简介	51
4.2 类	56
4.3 对 象	67
4.4 访问权限	72
4.5 本章小结	74
第5章 类的继承及接口	76
5.1 类的继承	76
5.2 抽象类	83
5.3 内部类、匿名类及最终类	86
5.4 接口	88
5.5 本章小结	91
第6章 常用实用类	93
6.1 字符串	93
6.2 字符串与基本数据之间的互相转换	102
6.3 Date 类	104
6.4 Calendar 类	104
6.5 Random 类	106
6.6 向量 (Vector) 类	108
6.7 本章小结	110
第7章 输入输出流	112
7.1 数据流的基本概念	112
7.2 字节流初步	113
7.3 文件操作	116
7.4 随机文件的操作	120
7.5 字符流 (Reader 类和 Writer 类)	123
7.6 本章小结	127
第8章 Swing 图形用户界面	128
8.1 JAVA Swing 概述	128
8.2 JFrame	130
8.3 常用组件、容器及布局	131
8.4 布局	139

8.5	事件处理	144
8.6	菜单	155
8.7	DocumentEvent 事件	159
8.8	ItemEvent 事件	161
8.9	其他常用事件	163
8.10	对话框	172
8.11	本章小结	184
第 9 章	多线程	186
9.1	线程的概念	186
9.2	JAVA 线程的创建与启动	187
9.3	本章小结	193
第 10 章	JDBC 数据库操作	194
10.1	数据库访问模型	194
10.2	一个简易教务系统数据库	196
10.3	访问数据库的实现	198
10.4	简易教务管理系统的功能简介	202
10.5	本章小结	206
参考文献		207
附录	KeyEvent 类的常量值	208

第 1 章 概 述

1.1 JAVA 语言的产生

1991 年, Sun Microsystems 公司成立 Green 项目组, 目的是为家用消费电子产品开发一个分布式代码系统, 用户将信息发送给家用电器, 可以对它们进行控制和信息交换。Green 项目组最初采用 C++ 语言开发该系统, 但它太复杂而且安全性差, 不能满足要求, 于是项目组研究设计了一种基于 C++ 的语言 Oak (橡树)。

Oak 项目组在 Mark Ardreessen (其开发了 Mosaic 和 Netscape) 的启发下, 用 JAVA 编写了 Web 浏览器 (称为 HotJava) 以及 applet 在 Web 上应用, 开启了 JAVA 进军 Internet 的新局面。由于商标冲突, 1995 年 5 月, Sun 公司正式发布 JAVA 的第一个办公版本, Oak 语言更名为 java 语言。

作为新一代的面向对象的程序设计语言, JAVA 语言的平台无关性直接威胁到 Wintel 联盟的垄断地位。一些著名计算机公司相继购买了 JAVA 语言的使用权, 如 Apple、DEC、IBM、Microsoft、Netscape、Novell、Oracle、SGI 等。

JAVA 语言是当今主流开发语言之一, 由于它具有跨平台性、安全性和开发简单等特点, 已被广泛应用于电子商务、信息系统和无线终端等许多领域。可预想, 未来每个人的生活中都会包含有 JAVA 的一部分。

注: JAVA 是印度尼西亚的一个重要的盛产咖啡的岛屿, 中文名叫爪哇, 开发人员为这种新的语言起名为 JAVA, 其寓意是为世人端上一杯热咖啡。

1.2 JAVA 的特性

1.2.1 简单性

JAVA 语言的一些语法与 C/C++ 语言很相似, 有 C/C++ 语言基础的读者很容易学习和使用它。JAVA 语言适用范围不同, 摒弃了 C++ 语言中一些安全性差、过于复杂、容易混淆、不适合网络应用等缺陷, 不支持全程变量, 不支持 goto 语句, 也不支持指针, 并提供了规范的语法格式和自动的垃圾回收机制, 使程序开发简单化。

1.2.2 面向对象

JAVA 语言是一种纯的面向对象程序设计语言, 它是一种效率高、易理解且更加符合人们思维习惯的程序设计语言。本书在后面章节将详细介绍类、对象、接口、封装、继承和多态等基本概念。

1.2.3 分布式

JAVA 语言从诞生起就和网络联系在一起，支持 Internet 应用开发，在基本的 JAVA 应用编程接口中有一个网络应用编程接口(`java.net`)，它提供了用于网络应用编程的类库，如 `URL`、`URLConnection`、`Socket` 和 `ServerSocket`，也内置了 `TCP/IP`、`HTTP`、`FTP` 协议类库等。JAVA 的 RMI(远程方法调用)机制也是开发分布式应用的重要手段。

1.2.4 安全性

JAVA 平台采用了域管理方式的安全机制，无论是本地代码还是远程代码都可通过配置的策略，设定可以访问的资源域。当 JAVA 字节码进入专门处理该内容的程序(解释器)时，首先必须经过字节码校验器的检查，然后 JAVA 解释器将决定程序中类的内存布局。随后，类装载器负责把来自网络的类装载到单独的内存区域，避免应用程序之间相互干扰破坏。最后，客户端用户还可以限制从网上装载的类访问某些文件系统。上述机制结合起来，使得 JAVA 成为安全的编程语言。

1.2.5 平台无关性

JAVA 源文件在 JAVA 虚拟机(`Java virtual Machine`, `JVM`)上编译成字节码文件，然后可以在具有 JAVA 运行环境(`Java Running Environment`, `JRE`)的任何操作系统上运行。`JRE` 由 `JVM`、类库和一些核心文件组成。字节码最大的特点就是可以跨平台，即经常听说的“一次编译，到处运行”，正是这一特性成为 JAVA 得以迅速普及的原因。

1.2.6 多线程

JAVA 语言支持多个线程的并发执行，这些线程独立地执行各自的 JAVA 代码、处理公共数据区和私有栈的数据，也提供多线程之间的同步机制(`synchronized`)。在同一个 `JVM` 进程中，程序的多任务是通过线程来实现的。当有新的线程产生时，操作系统不分配新的内存，而是让新线程共享原有进程的内存块。JAVA 语言提供了多线程编程的扩展点，并给出了功能强大的线程控制 API。

1.2.7 动态性

JAVA 语言的一个重要特点就是能适应环境的动态变化。由后面的章节可知，JAVA 程序是一个或多个类组成的，类是 JAVA 的基本单元。类要么是用户自己定义的，要么由类装载器(`Class Loader`)从类库中动态地载入到运行环境中，也可以通过网络来载入所需要的类。

1.2.8 垃圾回收机制

JAVA 语言的另一个重要特点就是具有一个垃圾回收器。类的实例对象和数组所需的内存空间是在堆上分配的(基本数据类型所需的内存空间在栈上分配的)，解释器承担了

实例对象内存分配工作，分配时就开始记录其占用的空间，使用完毕后便将其回收到垃圾箱中。

1.3 JAVA 的平台简介

JAVA 语言在网络编程方面应用得很广，它具有简单、多变、面向对象、跨平台等特点，具有很好的移植性和安全性，这些给网络编程带来了很多便利。

1999 年 6 月，Sun 公司推出的 JDK1.3 将 java 平台划分为 J2SE、J2ME 和 J2EE 三大平台，分别用于桌面、嵌入式和企业级应用。2004 年，Sun 公司发布 J2SE5.0，随后又发布 JAVASE6，并取消 JAVA2 名称，JAVA 三大平台更名为 JAVA SE、JAVA ME 和 JAVA EE。

JAVA SE—— JAVA Standard Edition, JAVA 标准版，它提供了标准的 JAVA Development Kit，主要用于桌面应用程序和低端的服务器应用程序的开发。

JAVA ME—— JAVA Micro Edition, JAVA 微型版，它提供了一种很小的开发环境，主要用于嵌入式产品等开发，如智能手机、平板电脑、可穿戴设备和智能家电等。

JAVA EE—— JAVA Enterprise Edition, JAVA 企业版，提供了企业级开发的各种技术，主要用于企业级应用开发。

其中，JAVA SE 是标准 JAVA 语言，是学习 JAVA ME、JAVA EE 的基础。

1.4 JAVA 程序的开发过程

JAVA SE 提供两种类型的程序，即应用程序（Application）和小应用程序（Applet），两者都必须在 JAVA 虚拟机（JAVA virtual machine, JVM）上运行。

JVM 是用软件模拟的计算机，它定义了指令集、寄存器集、类文件结构栈、垃圾收集堆、内存区域等，提供了跨平台的基础框架。

JAVA Application 程序（*.java）经过编译生成字节码文件（*.class），再由解释器解释执行。而 Applet 程序则是将编译后产生的字节码文件作为对象嵌入到网页文件（*.html 或 *.htm）中，在浏览器中运行。

JAVA 程序开发的一般过程：

第 1 步，编辑 JAVA 源程序。初学者可以使用文本编辑器（如记事本或 Edit）来编写源文件，也可以使用 eclipse、NetBeans 等开发工具来编写。

第 2 步，编译 JAVA 源程序。使用 JAVA 编译器（javac.exe），将 JAVA 源文件编译成字节码文件。

第 3 步，解释执行字节码文件。使用 JAVA 解释器（java.exe）来执行字节码文件。

程序开发过程如图 1.1 所示。

字节码文件是 JAVA 虚拟机中可执行的文件格式，是与平台无关的二进制码，执行时由解释器解释成本地机器码，解释一句，执行一句。JAVA 编译器针对不同平台的硬件提供了不同的编译代码规范，使得 JAVA 软件能够独立于平台。

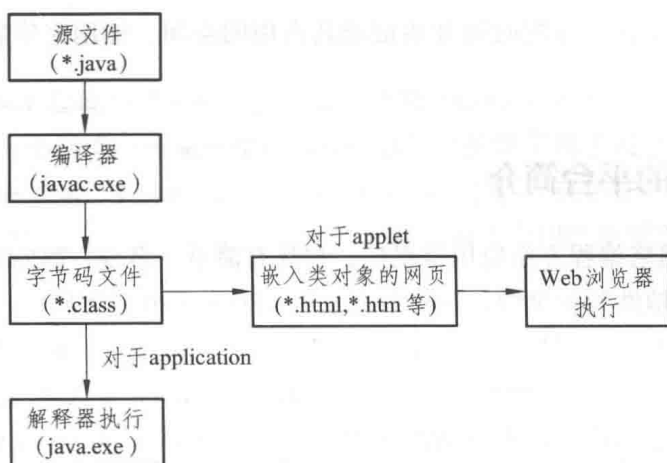


图 1.1 JAVA 程序的开发过程

1.4.1 JAVA 工具包 JDK

JAVA 不仅提供了一个丰富的语言环境和运行环境，而且还提供了一个开源的 JAVA 开发工具集（JAVA development kit, JDK）。读者可以利用这个工具集来开发 JAVA 程序。其官方网站提供了下载链接：

（1）在浏览器中输入：

<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html> 出现如图 1.2 所示的页面。

（2）选中 “Accept License Agreement”；

（3）选择 JDK 安装包，如果您的操作系统是 32 位的，则点击

Windows x86 89.09 MB jdk-7u25-windows-i586.exe

安装下载的 JDK 安装包。在安装过程，可以设置安装路径和选择组件。注意安装包含有名称为 i586 是针对 32 位的操作系统，而 x64 是针对 64 位的操作系统。

JDK 帮助文档（.chm 或.html 格式）是重要的编程手册，当遇到不熟悉的类或语法知识时，可以根据类名或方法名等检索，懂得类及其继承关系、API 使用说明，以及相关的例子。读者可以在官方网站下载最新版本，建议与 Jdk 同时下载使用。

1.4.2 JDK 工具库主要程序

安装完成 JDK 后，在 bin 目录下有各种工具，用来执行编译后产生的代码（即.class 文件或 binary 代码），常用的工具有：

（1）Javac.exe：JAVA 编译工具，将.java 源文件编译为.class 文件。

（2）java.exe：JAVA 解释工具，用于启动 JAVA 的应用程序，执行命令后 JavaSE 平台将相应程序载入 JVM 运行。

（3）appletviewer.exe：小应用程序浏览器，一种执行网页文件上的 JAVA 小应用程序的 JAVA 浏览器。



图 1.2 JDK 下载页面

(4) javadoc.exe: 命令产生类的 API 说明文档 (.html 格式), API 说明文档包括类、接口描述及继承关系, 以及属性、方法描述与参数描述等。

(5) jar.exe : 是一种压缩工具, 可以压缩所有格式的文件。

1.4.3 JDK 环境配置

1. 设置 Windows XP/NT/2000

(1) 在“我的电脑”图标上单击右键, 选择“属性”菜单, 将出现“系统属性”界面 (见图 1.3)。点“高级”选项卡, 单击“环境变量”按钮, 将出现“环境变量”界面 (见图 1.4)。

(2) 在“系统变量”框中选择“path”, 单击“编辑”按钮, 出现“编辑系统变量”对话框, 如果是安装在默认的路径, 就在其中“变量值”栏的变量值前添加“C:\Program Files\Java\jdk1.7.0_25\bin;”。



图 1.3 系统属性

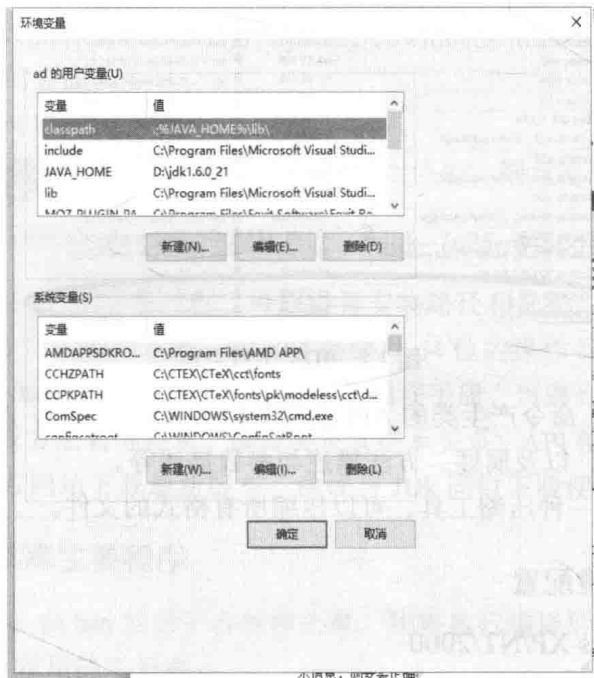


图 1.4 环境变量

(3) 在“系统变量”框中选择“classpath”，单击“编辑”按钮，出现“编辑系统变量”对话框，如果是安装在默认的路径，就在其中“变量值”栏的变量值前添加“.; C:\Program Files\Java\jdk1.7.0_25\lib;”。

(4) 设置完成后，打开 DOS 窗口，在命令提示符下输入“java”或“javac”，回车后，

如果出现其用法参数提示信息，则安装正确。

注意：安装路径上最好不要有空格（如默认安装路上就有空格）、中文等字符，否则，在编译时可能出现乱码问题，或路径不能识别等问题，导致不能正确执行。环境变量设置时，是在系统变量中设置，还是在用户变量中设置，又或者是在 DOS 命令行设置，是根据用户的具体需求而定的。

2. 设置 Windows 7

(1) 在桌面“计算机”图标上单击右键，选择“属性”菜单，将出现系统对话框界面，如图 1.5 所示。



图 1.5 系统对话框

- (2) 点左侧控制面板主页下面的“高级系统设置”选项，出现系统属性设置界面。
- (3) 点“高级”选项卡，再单击“环境变量”按钮，将出现“环境变量”设置界面。
- (4) 与设置 Windows XP/NT/2000 一样，设置环境变量。

1.5 Application 和 Applet 程序

JAVA SE 主要提供两种程序的开发,即应用程序(Application)和小应用程序(Applet)。

1.5.1 一个简单的 Application 程序

【例 1.1】 JAVA 应用程序在屏幕上输出"Hello World !"。

(1) 编辑源程序。

```
public class Example1_1 {  
    //程序的起点，所有程序从 main 开始执行  
    public static void main(String args[]){
```

```
System.out.println("Hello World!"); //在屏幕上打印"Hello World!"
```

```
}
```

将文件以“Example1_1.java”为文件名保存到硬盘(如 d:\java)。

(2) 编译源程序：将 JAVA 源程序编译转换成字节码，需在 DOS 命令提示符中输入：

```
d:\java >javac Example1_1.java
```

如果源程序没有语法错误，则编译后会产生和源程序相对应的几个.class 文件，例 1.1 中只有一个类编译产生一个字节码文件 Example1_1.class；如果编译出现错误，可根据错误提示信息修改源程序，重新编译。

(3) 解释执行字节码文件，在命令提示符中输入：

```
d:\java >java Example1_1
```

结果如图 1.6 所示。

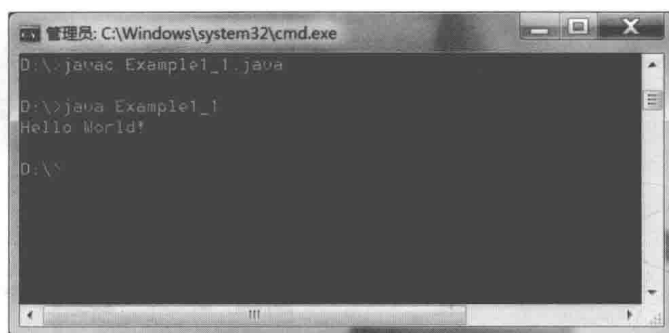


图 1.6 例 1.1 程序的运行结果

程序说明：

(1) 一个 JAVA 源文件可以由一个或多个类构成。在例 1.1 中只有一个类名为 Example1_1 的类构成。

(2) public class Example1_1 是类首。其中，class 是用来定义类的关键字。public 是定义访问权限（可见性）的关键字，则说明其他类都能访问公共类 Example1_1。除去类首部分即类体，即类首后面的这对大括号以及它们之间的内容。

(3) public static void main (String args[]) 是类体中的一个方法，它后面的两个花括号以及它们之间的内容叫做方法体。一个 JAVA 应用程序必须有且只有一个类含有 main() 方法，这个类称为它的主类。main() 方法是程序开始执行的位置，即 JAVA 解释器的入口。在一个 JAVA 应用程序中 main() 方法必须说明为 public static void。

(4) String args[] 声明一个字符串类型的数组 args，它是 main() 方法的参数。

(5) 给源文件命名时，如果源文件中有多类，那么只能有一个类是 public 类，同时该 public 类必须为主类，JAVA 应用程序的源文件名与主类名相同（严格区分大小写），扩展名为.java（不区分大小写）。如果源文件没有 public 类，则源文件的主名只要和某个类的名字相同即可。因此在例 1.1 中的源文件必须保存为 Example1_1.java。

1.5.2 Applet 小应用程序

小应用程序是一种作为对象嵌入到网页中的程序,在支持 JVM 的 Web 浏览器中运行。

【例 1.2】 JAVA 小应用程序输出 “Hello java!”。

(1) 编写 JAVA 源程序,保存文件名为 Example1_2.java

```
import java.applet.*;
import java.awt.*;

public class Example1_2 extends Applet{
    public void paint (Graphics g){
        g.setColor(Color.red);
        g.drawString("Hello java!", 30, 40);
    }
}
```

(2) 编译 JAVA 源程序,在 DOS 命令行中输入:

```
d:\java > javac Example1_2.java
```

回车后产生字节码文件 Example1_2.class。

(3) 编写如下 HTML 文件,保存文件名为 HelloApplet.html

```
<HTML>
<HEAD>
<TITLE> Applet 小应用程序 </TITLE>
</HEAD>
<BODY>
<APPLET code="Example1_2.class" width=300 height=200></APPLET>
</BODY>
</HTML>
```

(4) 在 DOS 命令行中输入:

```
d:\java> appletviewer HelloApplet.html
```

这时弹出一个显示内容为 “Hello java!” 的小窗口,如图 1.7 所示。

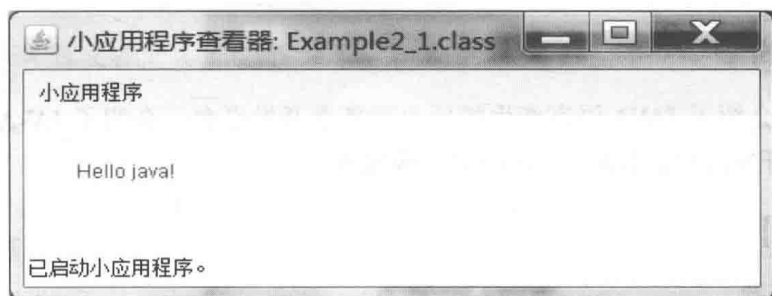


图 1.7 例 1.2 运行结果