

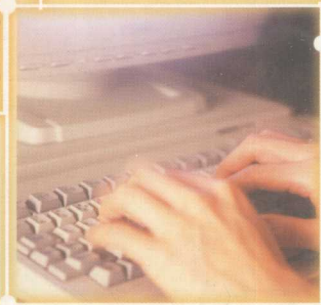
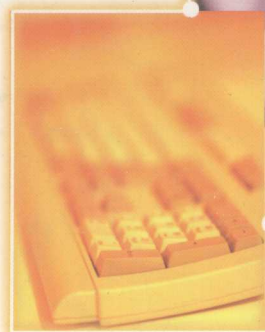


普通高等教育“十五”国家级规划教材
教育部高职高专规划教材

高职高专计算机专业系列教材

Java 程序设计

朱喜福 编著



清华大学出版社

普通高等教育“十五”国家级规划教材
教育部高职高专规划教材
高职高专计算机专业系列教材

Java 程序设计

朱喜福 编著

清华大学出版社

北 京

内 容 简 介

本书从 Java 语言最基本的概念开始讲述,包括 Java 语言的数据类型、运算符、表达式与流程控制、数组和方法等。然后对 Java 面向对象程序设计的基本概念,如类、对象、接口、继承和多态等进行了深入浅出的讲解。接下来介绍了 Java 的编程应用,包括图形绘制和图像显示、图形用户界面中的基本控制组件、容器和布局、常用的对话框和菜单设计的应用等。随后对 Java 语言的特点,如异常处理、多线程应用等做了详细介绍;最后对 Java 的输入输出处理、Java 小程序等进行了深入说明。

本书内容比较详尽,实例丰富,注重培养解决实际问题的能力。每章都附加了大量有针对性的练习和编程实训题,便于教师教学和检验学生的学习效果。

本书既可作为高等院校 Java 程序设计课程的教材和教学参考书,也可作为 Java 编程人员的参考书。

版权所有,翻印必究。举报电话:010-62782989 13901104297 13801310933

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Java 程序设计/朱喜福编著. —北京:清华大学出版社,2004

(高职高专计算机专业系列教材)

ISBN 7-302-08678-8

I. J… II. 朱… III. Java 语言—程序设计—高等学校:技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 047673 号

出 版 者:清华大学出版社

<http://www.tup.com.cn>

社 总 机:010-62770175

地 址:北京清华大学学研大厦

邮 编:100084

客户服务:010-62776969

组稿编辑:谢 琛

文稿编辑:张龙卿

印 刷 者:北京四季青印刷厂

装 订 厂:三河市化甲屯小学装订二厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:24 字数:551 千字

版 次:2004 年 7 月第 1 版 2004 年 7 月第 1 次印刷

书 号:ISBN 7-302-08678-8/TP·6219

印 数:1~5000

定 价:29.80 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或(010)62795704

序

1999年10月,教育部高教司主持召开了全国高职高专教材工作会议,会议要求尽快组织规划和编写一批高质量的、具有高职高专特色的基础和专业教材。根据会议精神,在清华大学出版社的支持下,于2000年1月在上海召开了由来自全国各地的部分高职、高专、成人教育及本科院校的代表参加的“高职高专计算机专业培养目标和课程设置体系研讨会”。与会的专家和教师一致认为,在当前教材建设严重滞后同高职教育迅速发展的矛盾十分突出的情况下,编写一套适应高等职业教育培养技术应用性人才要求的、真正具有高职特色的、体系完整的计算机专业系列教材十分必要而且迫切。会议成立了高职高专计算机专业系列教材编审委员会,明确了高职计算机专业的培养目标,即掌握计算机专业有关的基本理论、基本知识和基本技能,尤其要求具有对应用系统的操作使用、维护维修、管理和初步开发的能力。

根据上述目标,编委会拟定了本套教材的编写原则。在教材内容安排上,以培养计算机应用能力为主线,构造该专业的课程设置体系和教学内容体系;从计算机应用需求出发进行理论教学,强调理论教学与实验实训密切结合,尤其突出实践体系与技术应用能力的实训环节的教学;教材编写力求内容新颖、结构合理、概念清楚、实用性强、通俗易懂、前后相关课程有较好的衔接。与本科教材相比,本套教材在培养学生的应用技能上更有特色。

根据目前各高职高专院校计算机专业的课程设置情况,编委会确定了首批出版的十几本教材。这些教材的作者多是在高职高专院校或本科院校的职业技术学院任教的、具有多年教学经验的教师,每本书均由计算机专业的资深教授或专家主审把关。我们还将在此基础上,陆续征集出版第二、三批教材,力争在3到5年内完成一套完整的高职高专计算机专业教材。

应当说明的是,凡是高等职业教育、高等专科教育和成人高等院校的计算机及其相关专业均可使用本套教材。各学校可以根据实际需要,在教学中适当增删一些内容、实训项目和练习题,从而更有针对性地帮助学生掌握计算机专业知识,并形成相关的应用能力。

由于各地区各学校在教学水平、培养目标理解等方面有所不同,加上这套教材编写时间仓促,难免会出现这样或那样的错误,敬请各学校在使用过

前言

计算机网络技术是计算机领域中发展最迅速的新技术之一,对社会和经济发展做出了极大的贡献。计算机网络的普及和推广,将会使越来越多的人能够利用网上资源,享受不断发展和完善的网络服务,提高工作效率和水平。随着我国经济和社会发展进一步与国际接轨,各种新技术的引入和应用开发正在蓬勃发展和深入。在程序设计应用领域,面向对象的编程技术正在逐步替代面向过程的程序设计技术,成为计算机应用开发的主流,尤其在网络应用开发中,面向对象的 Java 语言编程和相关的 Java 技术应用正在以前所未有的速度发展,网上购物、网上银行、远程教学、虚拟课堂、网络游戏等都在纷纷使用 Java 技术。

近年来,全国各地高校与计算机相关的很多专业都开设了 Java 程序设计或与 Java 技术相关的课程,尤其是计算机和其他相关专业的高职高专院校更是将 Java 语言作为学习和掌握面向对象编程技术的必修课。

本书作者现为北京联合大学师范学院计算机系副教授,近 4 年来先后给本科生开设《Java 程序设计》和《Java how to program》双语课等,并在高职高专计算机技术应用和计算机网络专业教授《Java 程序设计》课程。在这几年的教学过程中特别注意了解和认识高职高专学生的知识基础、能力水平、学习兴趣和特点等,并积累了一定的教学经验,深感学生理解和掌握 Java 面向对象的编程技术、应用庞大的 Java 类库编写解决实际问题的程序等都有一定的困难。因此,在讲授和编写《Java 程序设计》教材时,根据学生的特点,在内容取舍和编排、讲述语言和方法等都做了较多的探讨和实践。本书内容的取舍在做到有一定的广度和深度的同时,力求适合高职高专学生使用;知识点的讲解和编排由浅入深,循序渐进,通俗易懂;例题的选择既考虑到利于知识的理解和掌握,又考虑到学生的学习兴趣 and 编程应用;习题的编写具有较强的针对性,以帮助学生巩固所学知识和提高实际编程能力。

本书适合作为大、中专院校的低年级学生学习 Java 语言和 Java 编程技术的教材。学习本书不需要高级语言编程的基础,可以根据学生的不同程度安排每星期 3~4 课时进行讲授,全书讲授课时可安排 54~72 课时。

第 1 章讲述 Java 应用程序和 Java 小程序、Java 语言的基本数据类型

等,并对 Java 语言的方法、类和对象进行初步说明,可安排 3~4 课时。

第 2 章讲述 Java 语言的运算符、表达式与流程控制等,可安排 3~4 课时。

第 3 章讲述 Java 语言的数组、方法和方法重载等,可安排 3~4 课时。

第 4 章讲述 Java 面向对象程序设计的基本技术及在程序设计中的具体应用,可安排 6~8 课时。

第 5 章简单介绍 Java 类库,对一些常用类在 Java 编程中的应用进行了说明,可安排 3~4 课时。

第 6 章讲述 Java 的图形用户界面编程,包括字体和颜色的设置、图形绘制和图像显示、Graphics2D 画图,标签(Label)、按钮(Button)、文本区(TextField 和 TextArea)、复选框(Checkbox)、列表(List)、下拉列表(Choice)、画布(Canvas)、滚动条(Scrollbar)等基本控件及相应的事件处理,容器组件及各种布局、对话框和菜单设计等。考虑到课时安排,对 Swing GUI 组件没有做专门的介绍,通过第 6 章的学习将使很容易自学 Swing GUI 组件的相关内容。第 6 章可安排 12~16 课时。

第 7 章讲述 Java 的异常和异常处理,可安排 3~4 课时。

第 8 章讲述 Java 的多线程及其应用,可安排 6~8 课时。

第 9 章讲述 Java 的输入输出处理,可安排 6~8 课时。

第 10 章讲述 Java 小程序及 Applet 中的多媒体应用,可安排 3~4 课时。

考虑到课程教学和学生的实际情况,本书略去了 Java 的套接字(Socket)通信、数据报通信编程和 JDBC 数据库连接等内容,但通过全书内容的学习和编程练习,能使具备深入学习 Java 的许多必备知识,比较深入全面地掌握面向对象编程技术,并打下比较扎实的利用 Java 类库编写和开发 Java 程序的基础,为进一步学习掌握 Java 技术提供帮助。本书也可作为对 Java 编程感兴趣的读者学习 Java 的入门参考书。

参加本书编写的人员还有魏绍谦、操静涛、王永平、李湛、屈敬文和马涛等。由于作者水平有限,书中难免有错误和不妥之处,恳请广大读者、教师和同行专家批评指正。

作者

2003 年 12 月

目 录

第 1 章	Java 语言入门	1
1.1	Java 语言的特点	1
1.2	Java 语言的发展过程	1
1.3	两类 Java 程序	2
1.4	Java 程序中的关键字、标识符和分隔符	7
1.5	Java 语言的数据类型	10
1.5.1	数据类型的分类	10
1.5.2	基本数据类型	10
1.5.3	常量和变量	11
1.5.4	类型转换	15
1.6	Java 语言的方法	16
1.7	Java 语言的复合数据类型——类和对象的初步说明	19
	习题	24
第 2 章	运算符、表达式与流程控制	29
2.1	Java 语言的运算符和表达式	29
2.1.1	操作数、运算符和表达式的概念	29
2.1.2	Java 语言的运算符	29
2.1.3	Java 语言运算符的优先级和结合性	38
2.2	Java 语言的流程控制	39
2.2.1	语句	39
2.2.2	循环语句	40
2.2.3	分支语句	44
2.2.4	控制转移语句	49
	习题	53
第 3 章	数组和方法	62
3.1	Java 语言的数组	62
3.1.1	数组的概念	62

3.1.2	一维数组	62
3.1.3	二维数组	69
3.2	Java 语言的方法和方法重载	71
3.2.1	方法定义和使用的进一步说明	71
3.2.2	main() 方法	80
3.2.3	方法的重载	84
习题		86
第 4 章	面向对象的程序设计	90
4.1	面向对象程序设计的基本概念	90
4.2	类	92
4.2.1	类的定义	92
4.2.2	成员的访问权限修饰符	95
4.2.3	构造函数	97
4.2.4	对象创建和对象使用的进一步说明	100
4.2.5	类的静态属性和静态方法	106
4.2.6	类中变量和方法的进一步说明	108
4.3	继承和多态	112
4.3.1	父类和子类——继承和继承的实现	112
4.3.2	子类对父类构造函数的继承和调用	115
4.3.3	子类对父类的变量隐藏和方法覆盖	117
4.3.4	父类和子类对象的转换	119
4.3.5	抽象类和最终类	121
4.3.6	根类和数学类	125
4.3.7	接口	130
4.4	内部类和匿名类	135
4.4.1	内部类	135
4.4.2	匿名类	137
4.5	包的创建和引用	138
习题		140
第 5 章	Java 类库和常用类简介	148
5.1	Java 类库简介	148
5.2	Java 语言基础类库	150
5.2.1	数据类型类	150
5.2.2	System 类	152
5.3	字符串类——String 类和 StringBuffer 类	154
5.3.1	String 类	155

5.3.2	StringBuffer 类	161
5.4	Java 常用工具类库	164
5.4.1	Random 类	165
5.4.2	StringTokenizer 类	166
5.4.3	Vector 类	167
习题		172
第 6 章 Java 图形用户界面		173
6.1	图形用户界面概述和 java.awt 包简介	173
6.2	字体和颜色的设置、图形绘制和图像显示	175
6.2.1	设置字体	175
6.2.2	设置颜色	177
6.2.3	字符数组和字节数组的显示	179
6.2.4	Java 的图形绘制	180
6.2.5	图像显示	184
6.3	Graphics2D 画图简介	186
6.3.1	直线、矩形和椭圆	186
6.3.2	用渐变颜色画填充图形	188
6.3.3	图形的旋转、平移和缩放	189
6.4	标签、按钮、文本区与动作事件	191
6.4.1	标签	192
6.4.2	按钮	193
6.4.3	文本框	196
6.4.4	文本区	199
6.5	Java 应用程序的图形界面	203
6.5.1	容器和基本控制组件	203
6.5.2	应用程序图形界面的实现	203
6.5.3	Canvas 类和应用程序中的图像显示及图形绘制	207
6.6	单选框、复选框、列表和下拉列表	211
6.6.1	复选框和单选框	211
6.6.2	列表和下拉列表	215
6.7	滚动条	221
6.8	鼠标和键盘事件	224
6.8.1	鼠标事件	224
6.8.2	键盘事件	228
6.9	容器的布局	230
6.9.1	容器的嵌套	230
6.9.2	FlowLayout 布局	233

101	6.9.3 BorderLayout 布局	235
101	6.9.4 GridLayout 布局	237
101	6.9.5 CardLayout 布局	238
101	6.9.6 GradBagLayout 布局	241
101	6.9.7 NULL 布局	246
101	6.10 对话框的创建	247
	6.10.1 对话框	247
101	6.10.2 文件对话框	249
101	6.11 菜单设计	253
101	6.11.1 菜单条、菜单和菜单项	253
101	6.11.2 创建二级菜单和复选菜单项	258
101	6.11.3 弹出式菜单	262
101	习题	264
101	第7章 Java 的异常处理	267
101	7.1 Java 异常	267
101	7.1.1 异常的概念	267
101	7.1.2 Java 的异常类	270
101	7.1.3 Java 的异常处理	271
101	7.2 自定义异常	279
101	习题	282
101	第8章 多线程	283
101	8.1 多线程的概念	283
101	8.1.1 进程	283
101	8.1.2 线程和多线程	284
101	8.1.3 Java 中的多线程程序——Thread 类的初步了解	284
101	8.1.4 线程的休眠——sleep()方法	289
101	8.1.5 线程的等待——join()方法	290
101	8.2 Java 的线程类 Thread 和 Runnable 接口	291
101	8.2.1 Runnable 接口	291
101	8.2.2 Thread 类	293
101	8.3 线程的生命周期和线程的基本控制	296
101	8.3.1 线程的状态和生命周期	296
101	8.3.2 多线程的基本控制	297
101	8.4 线程的互斥锁和线程的同步	298
101	8.5 多线程应用的综合实例	303
101	习题	308

第 9 章	Java 的输入/输出流	310
9.1	Java 的输入/输出类库简介	310
9.1.1	流的概念	310
9.1.2	字节输入/输出流——InputStream 类和 OutputStream 类	311
9.1.3	字符输入/输出流——Reader 类和 Writer 类	313
9.2	字节流的顺序输入/输出	314
9.2.1	文件输入/输出流	314
9.2.2	过滤流——缓冲流、数据流、回压流和打印流	317
9.2.3	字节数组流	324
9.2.4	SequenceInputStream 类	326
9.2.5	对象流	327
9.2.6	管道流	329
9.3	文件类和文件的随机读取	333
9.3.1	文件类	333
9.3.2	文件的随机访问类	337
9.4	字符流的顺序输入/输出	340
9.4.1	InputStreamReader 类和 OutputStreamWriter 类	341
9.4.2	字符文件输入/输出流 FileReader 类和 FileWriter 类	344
9.4.3	BufferedReader 类和 BufferedWriter 类	344
9.4.4	LineNumberReader 类	351
	习题	352
第 10 章	Java Applet	354
10.1	Java 小程序——Applet 类	354
10.1.1	Applet 的运行和安全机制	354
10.1.2	Applet 的生命周期方法	355
10.1.3	HTML 文档中的<Applet>标记符	357
10.2	Applet 的界面设计	360
10.3	Applet 中的多媒体	361
10.3.1	绘制图形和显示图像	361
10.3.2	播放声音	363
10.3.3	显示动画	365
	习题	370

第 1 章

Java 语言入门

1.1 Java 语言的特点

Java 语言是一种面向对象的、能跨平台使用的、安全的、解释执行的网络编程语言;Java 语言支持多线程;Java 语言具有内存“垃圾”自动收集机制,它扫描内存,并自动释放那些不再使用的内存块。

1.2 Java 语言的发展过程

1996 年,Java 1.0 版正式推出。Java 的早期版本称为 JDK(Java Development Kit),如:JDK 1.0、JDK 1.02、JDK 1.1、JDK 1.1.5 等。

1999 年升级为 1.2 版,并改称为 Java 2。2000 年 9 月升级为 1.3 版;2001 年 7 月推出了 Java 的新版本 1.4 版。Java 2 分为三个体系。

- J2SE(Java 2 Software Development Kit, Standard Edition): Java 2 开发的标准版,适用于一般计算机的 Java 程序开发和程序运行平台。本书所讲的程序都是在该环境下编译运行的,所使用的具体版本名称是 j2sdk1.4.0_01-win(可在网上免费下载)。如果没有改变安装时默认的安装路径,则被安装在 c:\j2sdk1.4.0_01 目录下,该目录下有一个子目录 bin,编译和运行 Java 程序的命令都在该目录下。
- J2EE(Java 2 Software Development Kit, Enterprise Edition): 企业 e_Business 架构和 Web Service 服务开发及应用平台。
- J2ME(Java 2 Software Development Kit, Micro Edition): 移动商务,如无线通信、PDA 等小型电子设备的开发工具及应用平台。

1.3 两类 Java 程序

Java 程序分为 Java 应用程序(Java Application)和 Java 小程序(Java Applet)两类。Java 应用程序是一种能在支持 Java 的平台上,通过 Java 解释器(例如 Java. exe)独立运行的程序;Java 小程序不能独立运行,它是通过支持 Java 的浏览器来运行的。

1. 一个简单的 Java 应用程序

(1) 编辑源程序

例 1.1 下面是一个简单的 Java 应用程序。打开 Windows 下的记事本,键入下面的程序(本书中所有程序中的行号“1:,2:,…”是为了便于解释说明程序而加上的,不是程序的内容)。注意程序中各字母的大小写不要弄错,因为 Java 语言严格区分大小写。将该程序保存为 C:\j2sdk1.4.0_01\bin 目录下的,文件名为 FirstApp. java 的文件,注意文件名与类(class)名完全一致。

程序: FirstApp. java

```
1: public class FirstApp
2: {
3:     public static void main (String[] args)
4:     {
5:         System.out.println("欢迎学习 Java 语言!");
6:     }
7: }
```

(2) 编译源程序

进入 Windows 系统的 DOS 命令行界面,并进入 Java 2 的安装目录下的 bin 子目录(不同的计算机安装目录可能不同)。例如,该程序编译和运行的路径是 C:\j2sdk1.4.0_01\bin>,键入 javac FirstApp. java 并回车后,如果源程序没有错误,屏幕上没有输出, javac 命令将编译源文件 FirstApp. java,并在该目录下生成 FirstApp. class 文件,这是一个二进制数据表示的文件,通常称为字节码文件(byte-code)。

如果源程序有错误,编译时将会给出错误提示,通常根据提示可改正程序中的错误。例如,键入程序时若把第 5 行语句中的 System 的第一个字母小写,编译该程序时输出如图 1.1 所示结果,提示该程序在第 5 行有一个错误。把该行的第一个字母改为大写字母后程序能编译通过,生成名为 FirstApp. class 的字节码文件。

(3) 运行程序

编译后产生的字节码文件(文件后缀名为 class)格式是一种结构中立的、与机器无关的中间文件格式。字节码文件并不能作为程序直接运行,但只要安装有 Java 运行系统的计算机都能执行这种中间代码,字节码通过 Java 虚拟机(Java Virtual Machine, JVM,一个由应用程序仿真的虚拟计算机)解释执行。

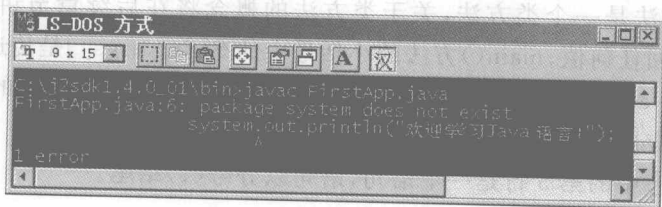


图 1.1 编译程序的输出

运行 Java 应用程序是运行编译后形成的字节码文件(本例是 FirstApp.class 文件),如果程序没有编译通过,则不能运行。可以通过 C:\j2sdk1.4.0_01\bin>目录下的 java.exe 命令运行 Java 应用程序,即:

C:\j2sdk1.4.0_01\bin>java FirstApp

回车后即可运行该应用程序,图 1.2 是该程序的运行结果,在 DOS 界面输出一行文字“欢迎学习 Java 语言!”。

注意 解释执行字节码文件时不要写后缀名.class,否则会出现错误。

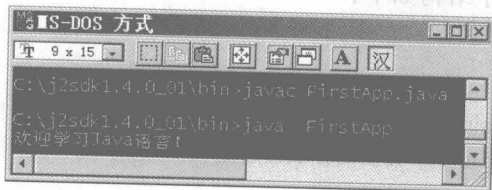


图 1.2 程序的运行结果

(4) 程序说明

Java 语言中的基本程序单位是类,第 1 行是类的声明,使用关键字 class 声明一个类,其后面是类的名称 FirstApp,类名由用户指定。关键字 public 表示这个类的访问特性是公共的。这里所说的“关键字”是指 Java 中有其特殊含义的英文单词,用户只能根据 Java 中的规则使用它。

一个源程序文件中可以声明多个类,但仅允许有一个公共的类,对于包含多个类的应用程序,应把包含 main()方法的类声明为 public 类,其他类不能用关键字 public 修饰。程序文件名必须与公共类的名称完全一致(包括大小写),如例 1.1 的源程序文件名为 FirstApp.java。

第 2 行和第 7 行是一对大括号,其中内容是类的定义,通常称为类体。类体中声明类的成员变量和类的成员方法。

第 3 行是该类成员方法 main()的方法头声明,在第 4 行和第 6 行的一对大括号中定义 main()方法的方法体。一个类中可以声明多个方法,但 Java 应用程序必须有一个 main()方法,并且方法的声明格式要与例 1.1 相同,应用程序从主方法 main()开始运行,通过主方法再使用(调用)其他的方法,在例 1.1 程序中调用了 println()方法输出一串文本字符,也称字符串。在 main()的方法头声明中,public 表示访问权限;static 的含义是

静态的,指明该方法是一个类方法,关于类方法的概念将在后续章节讲述;void 则指明 main()方法不返回任何值;main()方法的小括号中的 String args[]是传递给 main()方法的参数,参数名为 args,它是类 String 类型的数组。这些概念将在后面的章节详细说明,这里只需要了解应用程序的基本格式。

例 1.1 的程序中的第 5 行是一个语句,语句以分号(;)结尾。

```
System.out.println("欢迎学习 Java 语言!");
```

该语句的功能是调用 println()方法实现屏幕输出功能,在 DOS 命令行输出一行文本“欢迎学习 Java 语言!”后光标换行。

应用程序在命令行输出时还有另外一个方法 print(),该方法与 println()方法的不同之处在于:输出结果后光标不换行,仍停留在当前行。

(5) Java 应用程序的程序框架结构

从上面说明可知,Java 程序是由类(class)构成的。Java 应用程序的主类(public class)中一定包含 main()方法,它是应用程序执行的入口。Java 应用程序(Java Application)是一种能在支持 Java 的平台上独立运行的程序,通过 Java 虚拟机解释执行。

Java 应用程序的程序结构如下:

```
public class 类名
{
    public static void main(String args[])
    { //本对大括号中是 main()方法的方法体的定义
        ...;
    }
    //其他方法定义
}
```

2. 一个简单的 Java 小程序

Java 小程序的源程序编辑和编译过程与 Java 应用程序是一样的。但 Java 小程序不能独立运行,它通过支持 Java 浏览器来运行。因此,Java 小程序要通过标记符 <APPLET></APPLET>把编译形成的字节码文件嵌入到 html 文档中,通过浏览器运行 html 文档即可运行小程序。

例 1.2 下面是一个简单的 Java 小程序。与例 1.1 类似,键入并保存下面的程序在目录 C:\j2sdk1.4.0_01\bin>下,文件名为 FirstApplet.java,通过 javac 命令编译该程序,则在该目录下生成 FirstApplet.class 的字节码文件。

程序: FirstApplet.java

```
1: import java.awt.Graphics;
2: import java.applet.Applet;
3: public class FirstApplet extends Applet
4: {
5:     public void paint(Graphics g)
```

```

6:      {
7:          g.drawString("欢迎学习 Java 语言!",20,20);
8:      }
9:  }

```

(1) 编写 html 文档

运行 Applet 必须有一个 HTML 文件,并通过标记符<APPLET></APPLET>将字节码文件 FirstApplet.class 嵌入到该网页文件中,将网页文件保存为 page1.htm(网页文件可任意命名)。

标记符<APPLET></APPLET>在 HTML 中最基本的形式为:

```
<APPLET CODE="FirstApplet.class" WIDTH=200 HEIGHT=60></APPLET>
```

CODE 属性指明了所要加载的后缀名为.class 的小程序的文件名,WIDTH 和 HEIGHT 属性给出了 Applet 显示区的大小,它们以像素为单位。

下面是网页文件 page1.htm 的源文件。

```

<HTML>
<HEAD><TITLE>Java 小程序</TITLE></HEAD>
<BODY>
<APPLET CODE="FirstApplet.class" WIDTH=200 HEIGHT=60></APPLET>
</BODY>
</HTML>

```

注意 HTML 标记语言并不区分大小写,因此除嵌入的 FirstApplet.class 字节码文件大小写必须与编译后文件名一致外,其他标记符大小写都可以,但习惯上将标记符大写。

(2) 运行 Java 小程序

通过支持 Java 的网页浏览器运行 html 文档即可运行小程序,图 1.3 是通过 IE 打开网页 page1.htm 运行 Java 小程序 FirstApplet.class 的结果。

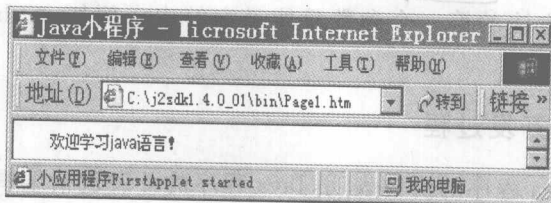


图 1.3 通过 IE 打开网页运行 Java 小程序

也可以通过 C:\j2sdk1.4.0_01\bin>目录下的 appletviewer 命令运行网页 page1.htm,查看 Java Applet 的运行结果,即:

```
C:\j2sdk1.4.0_01\bin>appletviewer page1.htm
```

回车后即可得到图 1.4 所示的结果。