

高等学校教材·计算机教学丛书

Java 程序设计

迟立颖 张银霞 张桂香 李 冰 编著
邓文新 主审



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

高等学校教材·计算机教学丛书

Java 程序设计

迟立颖 张银霞 张桂香 李 冰 编著
邓文新 主审

北京航空航天大学出版社

BEIHANG UNIVERSITY PRESS

内 容 简 介

本书由浅入深地介绍了 Java 的基础知识,并结合具体的实例重点阐述了 Java 面向对象的概念及其程序设计方法,还介绍了图形用户界面(GUI)等知识。全书共分 11 章,内容包括 Java 概述、Java 语言基础、Java 流程控制、数组与字符串、对象和类、异常处理、输入输出、图形用户界面、多线程、Applet 小程序以及 Java 高级应用简介等。

本书既可以作为高校非计算机专业的基础课的教材,又可以作为相关领域的培训教材,对 Java 程序开发人员也具有一定的参考价值。

图书在版编目(CIP)数据

Java 程序设计 / 迟立颖等编著. --北京:北京航空航天大学出版社,2011.5
ISBN 978-7-5124-0410-6

I. ①J… II. ①迟… III. ①Java 语言—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 063412 号

版权所有,侵权必究

Java 程序设计

迟立颖 张银霞 张桂香 李 冰 编著
责任编辑 文幼章

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱: bhpress@263.net 邮购电话:(010)82316936

北京时代华都印刷有限公司印装 各地书店经销

*

开本:787×1092 1/16 印张:19 字数:486 千字

2011 年 6 月第 1 版 2011 年 6 月第 1 次印刷 印数:4 000 册

ISBN 978-7-5124-0410-6 定价:32.00 元



总 前 言

随着科学技术、文化、教育、经济和社会的发展,计算机教学进入了我国历史上最火热的年代,欣欣向荣。就计算机专业而言,全国开办计算机本科专业的院校在2004年之初有505所,到2006年已经发展到771所。另外,在全国高校中的非计算机专业,包括理工农医以及文科(文史哲法教、经管、文艺)等专业,按各自专业的培养目标都融入了计算机课程的教学。过去出版界出版了一大批计算机教学方面的各类教材,满足了一定时期的需求,但是还不能完全适应计算机教学深化改革的要求。

面对《国家科学技术中长期发展纲要(2006年—2020年)》制订的信息技术发展目标,计算机教学也要随之进行改革,以便提高培养质量。教学要改革,教材建设必须跟上。面对各层次、各类型的学校和各类型的专业都要开设计算机课程,就应有多样化的教材,以适应各专业教学的需要。北京航空航天大学出版社是以出版高等教育教材为主的,愿对计算机教学的教材建设做出贡献。

为计算机类教材的出版,北京航空航天大学出版社成立了“高等学校教材·计算机教学丛书”编审委员会。出版计算机教材,得到了北京航空航天大学计算机学院的大力支持。该院有三位教育部高等学校计算机科学与技术教学指导委员会(下称教指委)的成员参加编审委员会的工作。其他成员是北京航空航天大学、北京交通大学等6所院校和中科院计算技术研究所对计算机教育有研究的教指委成员、专家、学者和出版社的领导。

我们组织编写、出版计算机课程教材,以大多数高校实际状况为基点,使其在现有基础上能提高一步,追求符合大多数高校本科教学适用为目标。按照教指委制订的计算机科学与技术本科专业规范和计算机基础课教学基本要求的精神,我们组织身居教学第一线,具有教学实践经验的教师进行编写。在出书品种和内容上,面对两个方面的教学:一是计算机专业本科教学,包括计算机导论、计算机专业技术基础课、计算机专业课等;二是非计算机专业的计算机基础课程的本科教学,包括理工农医类、文史哲法教类、经管类、艺术类等的计算机课程。

教材的编写注重以下几点。

1. 基础性。具有基础知识和基本理论,以使学生在专业发展上具有潜力,便于适应社会的需求。
2. 先进性。融入计算机科学与技术发展的新成果;瞄准计算机科学与技术发展的新方向,内容应具有前瞻性。这样,以使学生扩展视野,以便与科技、社会发展的脉络同步。
3. 实用性。一是适应教学的需求;二是理论与实践相结合,以使學生掌握实用技术。

编写、出版的教材能否适应教学改革的需求,只有师生在教与学的实践中做出评价,我们期望得到师生的批评和指正。



“高等学校教材·计算机教学丛书” 编审委员会成员

主 任 马殿富

副主任 麦中凡

陈炳和

委 员（以音序排列）

陈炳和 邓文新 金茂忠

刘建宾 刘明亮 罗四维

卢湘鸿 马殿富 麦中凡

张德生 谢建勋 熊 璋

张 莉



前言

随着计算机技术应用领域的不断拓宽,Java 作为一种 Internet 流行的程序设计语言备受青睐。Java 语言具有完全面向对象、平台无关性、多线程、安全性强、扩展容易等优点。目前越来越多的高等院校将其作为非计算机专业学生的计算机程序设计课程。

本书由具有多年教学经验的教师编写而成。多年的教学经验告诉我们,编写本书不能采用“使用手册”那样罗列知识的形式,而要循序渐进、由浅入深地结合实例介绍 Java 程序设计的基本方法,可以使读者快速地掌握 Java 的程序设计语言。书中侧重基础知识和基本应用能力的训练,具有通俗易懂、实例丰富、实用性强等特点。

全书共分 11 章。主要包括 Java 概述、Java 语言基础、Java 流程控制、数组与字符串、对象和类、异常处理、输入输出、图形用户界面、多线程、Applet 小程序以及 Java 高级应用简介等。第 1 章介绍 Java 的产生、特点、分类、运行环境以及 Java 程序的基本运行步骤;第 2 章介绍 Java 程序的基本组成、基本的符号集、数据类型、运算符和表达式;第 3 章介绍 Java 的顺序、选择和循环三种基本程序结构;第 4 章介绍一维数组和二维数组的创建及应用、固定长度字符串和变长字符串的各种操作方法;第 5 章以实例引出类和对象的概念,详细介绍类的定义、对象的创建、对象的引用、构造方法、各种访问修饰符、继承和重载、抽象类、接口和包,最后介绍 API 文档的使用;第 6 章介绍异常的概念、分类以及处理机制、异常的抛出、捕获等;第 7 章介绍 Java 中的基本输入输出流类以及文件操作;第 8 章介绍 Java 的图形用户界面,包括容器、各种组件、布局管理器、事件响应机制;第 9 章介绍多线程的概念、实现、基本控制、同步与通信;第 10 章介绍 Applet 的生命周期、利用 Applet 程序接收 HTML 文件中的参数、显示图像、声音、动画等;第 11 章介绍 Java 的数据库设计以及网络程序设计。

本书由迟立颖、张银霞、张桂香、李冰编著。迟立颖负责编写第 2、3、4、7 章,张银霞负责编写第 1、8、10 章,张桂香负责编写第 6、9、11 章,李冰负责编写第 5 章。邓文新教授审阅全书,并提出了宝贵意见。

本书在编写过程中得到了北京航空航天大学出版社的大力支持和帮助,在此表示衷心的感谢。同时对编写过程中参考文献资料的作者一并表示谢意。

由于作者水平及时间有限,书中难免有疏漏之处,敬请读者、专家批评指正。

编著者
2011 年 2 月



目 录

第 1 章 Java 程序设计	1	3.2.2 switch 语句	38
1.1 程序设计语言	1	3.2.3 选择结构语句的嵌套	41
1.2 Java 语言的产生与特点	2	3.3 循环结构语句	43
1.3 Java 程序的开发环境	2	3.3.1 while 循环语句	43
1.4 Java 程序的分类	3	3.3.2 do-while 循环语句	44
1.5 Java Application 的运行步骤	3	3.3.3 for 循环语句	45
1.6 Java Applet 的运行步骤	5	3.4 特殊的控制语句	49
习题一	6	3.4.1 break 语句	49
第 2 章 Java 语言基础	8	3.4.2 continue 语句	51
2.1 Java 程序的符号集	8	习题三	54
2.1.1 关键字	8	第 4 章 数组与字符串	61
2.1.2 标识符	8	4.1 数 组	61
2.1.3 运算符	9	4.1.1 一维数组的声明	61
2.1.4 分隔符	9	4.1.2 一维数组的创建	61
2.1.5 注 释	10	4.1.3 一维数组的引用	62
2.2 Java 程序的基本组成	10	4.1.4 多维数组	64
2.3 Java 的数据类型	12	4.1.5 二维数组的初始化	65
2.3.1 常 量	13	4.1.6 数组的排序与查找	66
2.3.2 变 量	15	4.1.7 数组的复制	70
2.4 运算符和表达式	20	4.2 String 类	71
2.4.1 算术运算符和算术表达式	20	4.2.1 String 类字符串的初始化	71
2.4.2 赋值运算符和赋值表达式	22	4.2.2 String 类字符串的常用方法	73
2.4.3 关系运算符与关系表达式	23	4.3 StringBuffer 类	75
2.4.4 逻辑运算符与逻辑表达式	24	习题四	78
2.4.5 位运算符与位表达式	26	第 5 章 对象和类	88
2.4.6 其他运算符	27	5.1 对象和类的概念	88
2.4.7 运算符的优先级与结合性	27	5.1.1 对 象	88
2.5 数据类型的转换	28	5.1.2 类	88
习题二	29	5.2 类的定义	89
第 3 章 Java 控制语句	33	5.3 创建对象与使用	91
3.1 Java 顺序结构语句	33	5.3.1 创建对象	92
3.1.1 表达式语句	33	5.3.2 对象的使用	92
3.1.2 块语句	33	5.4 构造方法	93
3.2 选择结构语句	34	5.5 访问控制修饰符	94
3.2.1 if 语句	34	5.5.1 公共访问控制符(public)	94



5.5.2 默认控制符	94	6.2.3 直接抛出异常:throw 语句	140
5.5.3 私有访问控制符(private)	94	6.2.4 异常的间接声明抛弃:throws 子语句	141
5.5.4 保护访问控制符(protected)	95	习题六	142
5.6 最终修饰符(final)	96	第 7 章 输入输出	144
5.6.1 最终类	96	7.1 输入输出流	144
5.6.2 最终属性	96	7.1.1 数据流的基本概念	144
5.6.3 最终方法	96	7.1.2 输入输出流类库	144
5.6.4 终结器	96	7.1.3 输入数据流 InputStream 类	145
5.7 静态修饰符(static)	97	7.1.4 OutputStream 类	145
5.7.1 静态属性	97	7.2 基本数据流类	146
5.7.2 静态方法	97	7.2.1 文件数据流	146
5.7.3 静态初始化器	98	7.2.2 缓冲区数据流	147
5.8 继承与重载	99	7.2.3 数据数据流	148
5.8.1 属性的继承与隐藏	100	7.2.4 管道数据流	151
5.8.2 方法的继承、覆盖与重载	101	7.2.5 对象流	153
5.8.3 this 和 super	102	7.3 文件管理	155
5.8.4 构造方法的重载与继承	104	7.3.1 File 类	155
5.9 抽象修饰符(abstract)	106	7.3.2 随机文件的访问	157
5.10 接口	107	习题七	160
5.10.1 接口的定义	108	第 8 章 图形用户界面	165
5.10.2 接口的使用	108	8.1 容器组件	166
5.11 包	109	8.1.1 框架(frame)	166
5.11.1 包的创建	109	8.1.2 面板(Panel)	167
5.11.2 包的引用	110	8.2 标签、按钮与文本框	168
5.11.3 包中类的编译与运行	111	8.2.1 标签(Label)	168
5.12 Java API 文档的使用	112	8.2.2 按钮(Button)	169
5.12.1 Java API 文档的使用	112	8.2.3 文本框	169
5.12.2 Java 中常用几个包	114	8.3 布局管理器	171
5.12.3 Java 中常用的几个类	114	8.3.1 流式布局管理器(FlowLayout)	171
习题五	123	8.3.2 边界布局管理器(BorderLayout)	172
第 6 章 异常处理	131	8.3.3 卡式布局管理器(CardLayout)	173
6.1 异常的概念	131	8.3.4 网格布局管理器(GridLayout)	176
6.1.1 异常的定义和分类	131	8.3.5 网格箱布局管理器(GridBagLayout)	177
6.1.2 Java 异常处理机制	132	8.3.6 布局管理器综合举例	178
6.1.3 异常类及继承关系	133	8.4 事件处理机制	181
6.2 异常的处理	135		
6.2.1 异常的直接捕获与处理:try - catch - finally 语句	135		
6.2.2 嵌套的 try - catch - finally 语句	139		



8.4.1 Java 事件处理机制	181	9.4 多线程的同步与通信	245
8.4.2 事件响应程序设计的基本步骤	182	9.4.1 线程同步	245
8.4.3 Java 中常见的事件类、监听接口及其方法	183	9.4.2 线程通信	247
8.4.4 Java 中的适配器类	189	习题九	249
8.5 文本、图形的基本操作	194	第 10 章 Applet 小程序	251
8.5.1 绘制文本	194	10.1 Applet 小程序的基本原理	251
8.5.2 字体(Font)	195	10.2 Applet 的生命周期	251
8.5.3 颜色(Color)	196	10.3 利用 Applet 接收 HTML 传递的参数	254
8.5.4 绘制图形	198	10.3.1 控制窗口特征的 Applet 标记属性	254
8.6 复选框、单选按钮、组合框和列表框	202	10.3.2 向 Applet 传递参数的<PARAM> 参数标记	255
8.6.1 复选框(Checkbox)	202	10.4 绘制文字与图形	256
8.6.2 单选按钮	204	10.5 播放声音	256
8.6.3 组合框(Choice)	206	10.6 图像绘制	258
8.6.4 列表框(List)	207	10.6.1 图像的加载	258
8.7 菜单组件	210	10.6.2 图像的显示	258
8.7.1 菜单项(MenuItem)	210	10.6.3 在 Application 中绘制图像	260
8.7.2 菜单(Menu)	211	10.6.4 图像和声音的结合	261
8.7.3 菜单条(MenuBar)	212	10.7 动画	262
8.8 对话框组件	214	习题十	264
8.8.1 对话框(Dialog)	214	第 11 章 Java 高级应用简介	267
8.8.2 文件对话框(FileDialog)	217	11.1 JDBC 数据库设计	267
8.8.3 JOptionPane 类实现的对话框	220	11.1.1 数据库与 JDBC 数据库概述	267
习题八	223	11.1.2 SQL 语言	268
第 9 章 多线程	229	11.1.3 JDBC 连接数据库	269
9.1 线程的概念	229	11.1.4 JDBC API 的主要类和接口介绍	271
9.1.1 程序、线程和进程	229	11.1.5 使用 JDBC 的编程步骤	276
9.1.2 线程的概念模型	229	11.2 网络程序设计	279
9.2 多线程的实现	230	11.2.1 URL 类和 URLConnection 类	279
9.2.1 通过实现 Runnable 接口创建线程	231	11.2.2 InetAddress 类	284
9.2.2 继承 Thread 类实现多线程	232	11.2.3 Socket 类和 ServerSocket 类	285
9.2.3 创建线程的两种方法比较	233	11.2.4 DatagramSocket 类和 DatagramPacket 类	288
9.3 多线程的基本控制	234	习题十一	292
9.3.1 线程优先级和调度	234	参考文献	294
9.3.2 线程的生命周期	235		
9.3.3 线程的控制	237		
9.3.4 线程组	244		



第 1 章

Java 程序设计

在众多的程序设计语言中,Java 作为一种面向对象的程序设计语言,备受青睐。本章主要介绍各种计算机程序设计语言的特点、Java 的产生及特点、Java 的开发环境、Java 程序的分类、各类 Java 程序的基本运行步骤。



1.1 程序设计语言

程序设计语言是人与计算机交流的工具。计算机中运行的各种程序均是由各类程序设计语言编制而成的。编制程序的过程就如同使用某种自然语言写作文一样,不过这个“作文”要按照某种程序设计语言的语法编写,并且要在计算机上运行。因此,要编程必须学习程序设计语言,不同的程序设计语言适合编写不同类别的程序,自从程序设计语言诞生到现在已经出现了几十上百种,按特点基本可以分为以下三类。

1. 面向机器的语言

面向机器的语言是与机器相关的,用户必须熟悉计算机的内部结构以及其对应的指令序列才可以使用。面向机器的语言又分为两类:机器语言和汇编语言。

机器语言是以二进制代码组成的机器指令集合。这种语言编制的程序运行效率极高,但程序很不直观,编写很简单的功能就需要大量代码,重用性差,而且编写效率较低,很容易出错。

汇编语言比机器语言直观。它用助记符来代替二进制代码,编程工作相对机器语言简化,使用起来方便了很多,错误也相对减少,但不同的指令集的机器仍有不同的汇编语言,程序重用性也很低。

2. 面向过程的语言

现代应用程序开发多数都是使用高级语言。高级语言是与机器不相关的一类程序设计语言,比较接近人类的自然语言。因此,使用高级语言开发的程序可读性较好,便于维护。同时,由于高级语言并不直接和硬件相关,其编制出来的程序的移植性和重用性较好。

高级语言又分为面向过程的语言和面向对象的语言两种。

所谓面向过程(procedure oriented)的程序设计就是以要解决的问题为核心,分析问题中所涉及的数据及数据之间的逻辑关系(数据结构),进而确定解决问题的方法(算法)。因此,面向过程的程序设计语言注重高质量的数据结构和算法,研究采用什么样的数据结构来描述问题,以及采用什么样的算法来高效地解决问题。由于面向过程的程序设计语言是以要解决的问题为核心编程,因此如果问题稍微发生改变,就需要重新编写程序。在 20 世纪 70 年代和 80 年代,大多数流行的高级语言都是面向过程的程序设计语言,如 Basic、Fortran、Pascal 和 C 等。



3. 面向对象的语言

面向对象(object oriented)的基本思想就是以一种更接近人类一般思维的方式去看待世界,把世界上的任何一个个体都看成是一个对象。每个对象都有自己的特点,并以自己的方式做事,不同对象之间存在着交往,因此构成了大千世界,而世界上的对象又分为不同的类别。面向对象的程序设计就是通过定义类来描述自然界中的类别,通过创建类的对象来模拟自然界中的对象,对象的特点就是它的属性,而对象能做的事就是它的方法。常见的面向对象的程序设计语言包括 C++ 和 Java 等。

1.2 Java 语言的产生与特点

1. 产生

1991, Sun 公司的 James Gosling 等人为了开发家电产品的软件,设计了一种平台无关的面向对象的计算机语言,最初定名为 Oak(橡树),在 1995 年注册时,发现 Oak 已被使用,因此改为 Java(爪哇,盛产咖啡的一个岛屿的名称)。Java 诞生之后,以其强大的功能迅速成为网络编程的首选语言。

2. 特点

Java 作为一种平台无关的、适合于分布式环境的面向对象的编程语言,具有安全性、动态执行、多线程等特点。

平台无关是指 Java 程序不需要更改可以在各种操作系统上运行。

分布式环境是指可以将数据或计算分散到网络中的不同主机上,即 Java 可以实现网络中的不同主机上的数据或操作的编程。Java 不仅适合于网络编程,对于单机的程序设计功能也很强大。

Java 的安全性可从两个方面得到保证:在 Java 中摒弃了指针类型,增加了数组越界检查功能以及无用内存的回收功能;另外,Java 的运行环境提供了字节码校验器、类装载器和文件访问限定功能。字节码校验器检测字节码是否被改动过,类装载器将类装载到与其他程序分开的内存区域,文件访问限定可以限制该程序不能访问计算机的其他资源。

动态执行是指 Java 允许程序动态地装入运行过程中所需要的类。

多线程机制可以使 Java 程序并发执行,从而提高系统的效率。

1.3 Java 程序的开发环境

比较典型的 Java 开发运行环境有 Sun 公司的 Java 开发工具集 JDK(Java Development Kit)、Borland 公司的 JBuilder、微软公司的 Visual J++。由于 JDK 简便易学,文档资料齐全,比较适合初学者学习,而 JBuilder 和 Visual J++ 为集成开发环境,适合于有一定基础的程序开发人员使用。本书使用 JDK 作为 Java 开发工具。

1. JDK 的版本

JDK 可以免费从 Oracle 公司的网站 <http://www.oracle.com> 下载(由于 Sun 公司被 Oracle 公司收购,原 Sun 公司的网址 <http://www.sun.com> 被自动转向 Oracle 公司首页)。JDK 的版本主要有 JDK1.1, JDK1.2, JDK1.3, JDK1.4, JDK5.0, JDK6.0。从 JDK5.0 开始提



供了泛型等非常实用的功能,其运行效率大大提高。

JDK 通常有三个版本:标准版 J2SE(Java 2 Standard Edition),是常用的一个版本,从 JDK 5.0 开始,改名为 Java SE;企业版 J2EE(Java 2 Enterprise Edition),用于开发 J2EE 应用程序。从 JDK 5.0 开始,改名为 Java EE;J2ME(Java 2 Micro Edition),用于移动设备、嵌入式设备上的 java 应用程序开发,从 JDK 5.0 开始,改名为 Java ME。

在下载 JDK 时,建议同时下载 Java Documentation,这是 Java 帮助文档,对于开发 Java 程序大有益处。

2. JDK 的安装与设置

本书采用 J2SDK1.4.2 版本作为程序开发环境,安装路径默认为 c:\j2sdk1.4.2,安装后的目录包括:

bin 目录:Java 开发工具,包括 Java 编译器、解释器等;

demo 目录:一些实例程序;

lib 目录:Java 开发类库;

jre 目录:Java 运行环境,包括 Java 虚拟机、运行类库等。

开发环境安装之后需要做系统设置,右击“我的电脑”,在快捷菜单中选择“属性”菜单项,在弹出的对话框中单击“环境变量”,在弹出的页面中对“系统变量”做如下设置:

path= c:\j2sdk1.4.2\bin,

java_home= c:\j2sdk1.4.2(安装路径)

Classpath=.;C:\j2sdk1.4.2\lib\tools.jar; C:\j2sdk1.4.2\lib\dt.jar; C:\j2sdk1.4.2\bin;

设置成功后,即可开始 Java 程序的开发与调试。



1.4 Java 程序的分类

Java 程序根据程序结构与执行机理的不同可以分为两类:Java Application(应用程序)和 Java Applet(小程序)。Java Application 程序可以直接编译运行,而 Java Applet 程序需要嵌入到 HTML 语言编写的 Web 页面中,通过页面调用该程序而运行。



1.5 Java Application 的运行步骤

Java Application 程序的运行要经过编辑、编译、解释运行三个步骤,如图 1-1 所示。

1. 编辑

编辑是通过编辑软件将源程序存成扩展名为 java 的磁盘文件的过程。JDK 环境并不提供专门的编辑软件,用户可以借助 DOS 环境下的 EDIT 或是 Windows 下的记事本等软件进行编辑。

例如,有如下程序,其功能是在屏幕上显示“这是我的第一个 Java 应用程序。”

```
import java.io.*;
public class MyFirstJavaProgram
```

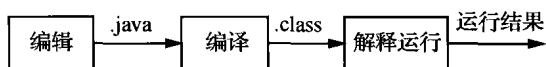


图 1-1 Java Application 程序的运行步骤



```
{  
    public static void main (String args [ ])  
    {  
        System.out.println("这是我的第一个 Java 应用程序");  
    }  
}
```

Java 程序是由类为基本单位组成的。类以 class 为标志, class 后面是类名。一个程序中可以有多个类, 但是只能有一个类的前面允许加 public(公共的), 这个类称为主类。

Java 源程序的名字必须与主类名完全相同, 扩展名必须为 java。注意 Java 是严格区分大小写, 因此大小写也要一致。

这里, 将以上程序通过记事本存为 MyFirstJavaProgram.java 的源文件。

2. 编译

编译的过程是利用编译程序将源程序翻译成扩展名为 class 的字节码文件的过程。

编译的命令格式为:

javac 源文件名.java

在命令提示符状态下, 键入 javac MyFirstJavaProgram.java (↵代表回车键)。

如果源程序没有错误, 屏幕上没有显示信息, 直接返回命令提示符, 磁盘上会增加一个扩展名为 class 的字节码文件, 用 dir 命令会发现磁盘上增加了一个 MyFirstJavaProgram.class 的文件, 如图 1-2 所示。如果源程序有错误, 编译器会指出错误位置及原因, 例如, 将源程序中的语句 System.out.println("这是我的第一个 Java 应用程序"); 后面的分号去掉, 会出现如图 1-3 所示界面, 表明共有一处错误, 第 7 行 } 位置有误, 希望是分号, 此时需要返回编辑状态重新修改源程序, 然后再编译。

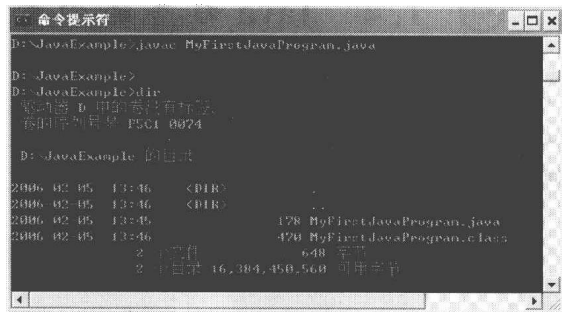


图 1-2 无语法错误的编译效果



图 1-3 有语法错误的编译效果

3. 解释运行

运行的过程是通过解释程序逐一解释字节码文件, 完成程序的执行过程。

运行的命令格式为:

java 字节码文件名

注意: 字节码文件名的后面不要加扩展名 class。

在命令提示符状态下, 键入 java MyFirstJavaProgram (↵)。这时屏幕上会出现如图 1-4 所示的运行结果(显示“这是我的第一个 Java 应用程序。”)。

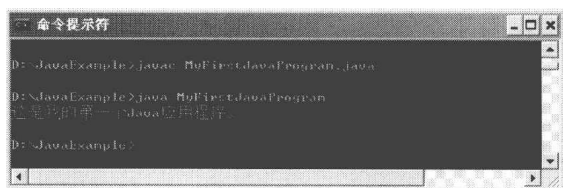


图 1-4 正确的运行结果



1.6 Java Applet 的运行步骤

Java Applet 程序的运行要经过编辑、编译、建立网页文件、运行网页文件四个步骤,如图 1-5 所示。

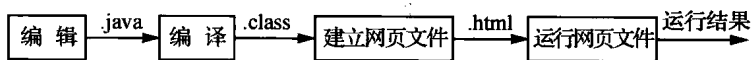


图 1-5 Java Applet 程序的运行步骤

例如,有如下 Applet 小程序,其功能是在屏幕上显示“这是我的第一个 Java 小程序。”

```

import java.applet. * ;
import java.awt. * ;
public class MyFirstJavaApplet extends Applet
{
    public void paint(Graphics g)
    {
        g.drawString( " 这是我的第一个 Java 小程序。 " ,40,50);
    }
}
  
```

1. 编辑 Applet 源程序

利用记事本将以上源程序存为 MyFirstJavaApplet.java 文件。

2. 编译 Applet 源程序

在命令提示符下键入

```
javac MyFirstJavaApplet.java
```

如果无误,将生成字节码文件 MyFirstJavaApplet.class。

3. 编辑嵌入字节码文件的 HTML 网页文件

利用记事本建立扩展名为 HTML 的文件(MyFirstApplet.html)。该文件名不必与类名一致,内容如下:

```

<HTML>
<HEAD>
<TITLE>我的 Java 小程序</TITLE>
</HEAD>
<BODY>
  
```



```
<APPLET CODE =MyFirstJavaApplet.class   WIDTH=200   HEIGHT=100>
</APPLET>
</BODY>
</HTML>
```

4. 运行 HTML 文件

JDK 提供了一个名为 appletviewer 的小程序浏览工具。其命令格式如下：

appletviewer HTML 文件

在命令行提示符下,键入 **appletviewer MyFirstApplet.html**,如图 1-6 所示。
屏幕上会出现一个小窗口,内容如图 1-7 所示。

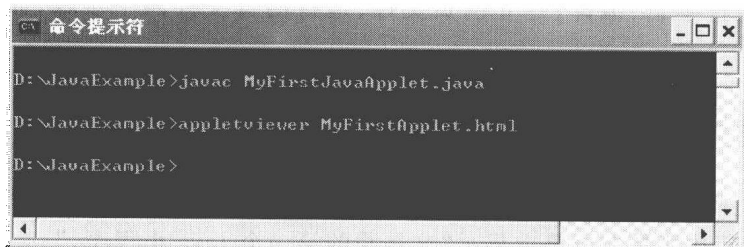


图 1-6 运行 HTML 文件

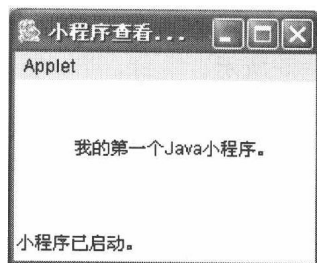


图 1-7 用 appletviewer 查看
Java Applet 程序的运行结果

当然,用户如果利用其他浏览器(比如 Internet Explorer 等)直接打开 MyFirstApplet.html 文件也可以看到对应的运行结果,如图 1-8 所示。



图 1-8 用 IE 浏览器察看 Java Applet 程序的运行结果



习题一

一、选择题

1. 以下不属于 Java 特点的是()。

A. 平台无关性

B. 安全性

C. 多线程

D. 静态性



2. 下列说法中正确的是()。
- A. Java 是不区分大小写的
 - B. Java 语言以方法为程序的基本单位
 - C. Applet 是一类特殊的 Java 程序,它嵌入 HTML 中,随主页发布到互联网上
 - D. Java 的源文件名和类名不允许相同
3. main 方法是 Java Application 程序执行的入口点。关于 main 方法的方法头以下哪项是合法的()。
- A. `public static void main ( )`
 - B. `public static void main ( String args[] )`
 - C. `public static int main (String [] arg )`
 - D. `public void main (String arg[])`
4. 编译 Java 源程序文件产生的字节码文件的扩展名为()。
- A. java
 - B. class
 - C. html
 - D. exe
5. 以下关于 Java Application 程序结构特点的描述中,错误的是()。
- A. 一个 Java Application 程序可以由一个或多个文件组成
 - B. Java 程序中声明有 public 类时,则 Java 程序的文件名必须与 public 类的类名相同,扩展名为 java
 - C. 在组成 Java Application 程序的多个类中,有且仅有一个主类
 - D. 在一个 Java Application 程序,可以声明多个 public 类
6. 若 Java 程序中定义了 3 个类,编译后可生成()个字节码文件。
- A. 4
 - B. 3
 - C. 2
 - D. 1

二、填空题

1. 通常所说的高级程序设计语言,根据程序设计思想不同,可以分为()和()两类。
2. 根据程序的构成和运行环境的不同,Java 源程序分为两大类:()程序和()程序。
3. 在应用程序中必须而且只能有一个()方法,程序将从该方法开始执行。
4. 有一段 Java 应用程序,它的主类名是 abc,那么保存它的源文件名应该是()。

三、简答题

- 1. 如何区分 Java 应用程序和 Applet 小程序。
- 2. 试述 Java 应用程序的基本运行步骤。
- 3. 试述 Java Applet 程序的基本运行步骤。
- 4. 分别用 Java 应用程序和 Applet 小程序实现在屏幕上显示“Java 是一种面向对象的程序设计语言”。



第 2 章

Java 语言基础

就像世界上任何一种人类的自然语言一样,程序世界里的每一种程序设计语言也都有自己的符号集、编写程序的语法规则等。本章将详细介绍 Java 语言的基础知识。

2.1 Java 程序的符号集

符号是构成程序的基本单位。Java 语言采用国际化的 Unicode 字符集。在这种字符集中,每个字符用 2 字节即 16 位表示。这样,整个字符集中共包含 65 535 个字符。其中,前面 256 个表示 ASCII 码,使其对 ASCII 码具有兼容性;后面 21 000 个字符用来表示汉字等非拉丁字符。Unicode 字符集只用在 Java 平台内部,当涉及打印、屏幕显示、键盘输入等外部操作时,仍由具体计算机系统决定其表示方法。

Java 程序中的基本符号包括关键字、标识符、运算符、分隔符、注释符等。

2.1.1 关键字

关键字通常也称为保留字,是系统预定义的具有专门意义和用途的符号。比如表示类型的关键字 `class`,`int` 等,控制语句中的关键字 `if`,`while` 等。表 2-1 列出了 Java 语言的全部关键字。

表 2-1 Java 的关键字表

<code>abstract</code>	<code>boolean</code>	<code>break</code>	<code>byte</code>	<code>byvalue *</code>	<code>case</code>	<code>cast *</code>
<code>catch</code>	<code>char</code>	<code>class</code>	<code>const *</code>	<code>continue</code>	<code>default</code>	<code>do</code>
<code>double</code>	<code>else</code>	<code>extends</code>	<code>false</code>	<code>final finally</code>	<code>float</code>	<code>for</code>
<code>future *</code>	<code>generic *</code>	<code>goto *</code>	<code>if</code>	<code>implements</code>	<code>import</code>	<code>inner *</code>
<code>instanceof</code>	<code>int</code>	<code>interface</code>	<code>long</code>	<code>native</code>	<code>new</code>	<code>null</code>
<code>operator *</code>	<code>outer *</code>	<code>package</code>	<code>private</code>	<code>protected</code>	<code>public</code>	<code>rest *</code>
<code>return</code>	<code>short</code>	<code>static</code>	<code>super</code>	<code>switch</code>	<code>synchronized</code>	
<code>this</code>	<code>throw</code>	<code>throws</code>	<code>transient</code>	<code>true try</code>	<code>var *</code>	
<code>void</code>	<code>volatile</code>	<code>while</code>				

其中 * 标记的关键字是被保留,但当前尚未使用

有些关键字,如 `cast`,`const`,`future`,`generic`,`goto`,`inner`,`operator`,`outer`,`rest`,`var` 等都是 Java 保留的没有意义的关键字。严格说来它们不是关键字,只是符号。

需要特别注意的是,由于程序设计语言的编译器在对程序进行编译的过程中,对关键字作特殊对待,所以,在程序中不能用关键字给对象命名。

2.1.2 标识符

在 Java 语言中,标识符用于标识一个对象(变量、常量、方法和类等)。标识符是以字母、