

国家骨干高职院校建设成果·计算机项目化系列教材

Java语言程序设计

刘妍东 主 编

郭亚东 副主编

清华大学出版社



国家骨干高职院校建设成果·计算机项目化系列教材

Java语言程序设计

刘妍东 主 编

郭亚东 副主编

清华大学出版社
北 京

内 容 简 介

本书全面介绍编写 Java 程序应具备的知识,主要包括 Java 语言概述、Java 程序设计的基本知识、面向对象编程、异常处理、图形用户界面设计、数据库编程、输入/输出流、多线程编程技术等。

本书中每个知识点都有配套的案例对应讲解,书中大量案例体现了面向对象的编程思想,可以使学生易于理解和掌握。

本书适合初学者和大中专学生入门学习,也适合有一定程序设计语言基础的读者自学,还可以作为 Java 培训班的教材,也可以作为程序设计人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java 语言程序设计/刘妍东主编.--北京:清华大学出版社,2014

国家骨干高职院校建设成果 计算机项目化系列教材

ISBN 978-7-302-34015-7

I. ①J… II. ①刘… III. ①JAVA 语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 234318 号

责任编辑:刘 青 刘翰鹏

封面设计:傅瑞学

责任校对:刘 静

责任印制:何 芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795764

印 装 者:保定市申画美凯印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:14.5 字 数:331 千字

版 次:2014 年 1 月第 1 版 印 次:2014 年 1 月第 1 次印刷

印 数:1~2800

定 价:29.00 元



《中华人民共和国国民经济和社会发展第十二个五年规划纲要》中指出：“在十二五期间，要全面提升信息化水平，加快建设宽带、融合、安全、泛在的下一代国家信息基础设施，推动信息化和工业化深度融合，推进经济社会各领域信息化。”

我国政府不失时机地提出抓紧在经济和社会发展的重要领域和关键环节率先应用信息技术，积极运用信息技术改造和提升传统产业，加快企业信息化步伐，以及加强信息资源的开发利用。预计未来几年，伴随我国经济的持续增长及各行各业自身的稳步发展，信息化投入的增长将会获得更加坚实的物质支持，我国信息化建设将在国民经济各领域更加全面有序地展开。此外，我国行业/企业信息化的政策法律环境将进一步优化，这也将为信息化建设营造良好的社会环境，促使其步入良性循环发展的轨道。

在经济社会各领域的信息化建设中，信息化手段已应用到工业领域和其他经济社会各领域之中，从事与行业相关业务相结合的信息系统应用较广泛，目前已经涉及政府、金融、工业、通信、交通、贸易、物流、能源等领域以及科技、教育、医疗、社保、环保和安全生产等民生领域。各领域所需信息化建设相关的计算机类人才，不仅要精通信息系统建设相关的基础知识，还要了解企业真实项目核心业务系统的应用架构、技术架构和信息架构的规划，掌握设计与开发能力。依托于计算机技术的信息化建设领域技术发展日新月异，这要求专业教育在教学内容上也要保证实时更新，做到“与时俱进”，与社会发展、企业发展保持同步。

本套教材是由北京信息职业技术学院的教师团队根据岗位职业能力、典型工作任务和工作过程分析，结合高等职业学院计算机类专业教学的需要，以及自身多年教学实践成果精心组织编写的，汇集了大量具有代表性的企业真实项目设计和开发案例。教材编者既有在教学一线具有丰富教学经验的教师团队，又有在企业工作多年的一线工程技术人员，结合“学院”与“企业”的优势，系统地总结了一套宝贵的开发经验，学习任务完全按企业工作流程与工作内容进行设计，使学生按照规范的企业工作流程，以“团队合作”方式进行学习，真正做到教材内容与企业的生产实际相结合、学生素质的培养与企业文化建设相结合。

因此，这套教材也是近几年在教育部关于职业教育改革精神推动下的专业教学改革成果，其价值在于它可以真正帮助从事和企业信息化建设相关领域的人员，尽早的接触到企业实际的应用环境和工作场景，尽快掌握工作技能和提高自己的企业文化素质，真正做到学以致用。

前言

FOREWORD

Java 语言是当今社会最实用、最具有影响力的编程语言之一。自 1995 年以来,Java 语言以其独特的跨平台特性,使之能够在不同类型的计算机和操作系统上运行。它的面向对象、分布式和可移植等特性使之应用到社会的各个方面。

Java 应用范围非常广,如动态网站、嵌入式开发和企业级应用等,程序员可以在掌握 Java 语言的基础上再根据自己的爱好来选择深入学习和就业的方向。

本书语言叙述通俗易懂,面向实际应用,采用以实践为主的教学方法,教材中没有过多阐述理论知识和算法,而是将主题案例贯穿到各个章节中,使学生在逐步学习的同时完成一个小型项目;通过案例和项目教学相结合进行教学,即通过案例教学对基本知识点进行讲解,通过课后的技能训练使学生逐步完成一个可以应用的小型项目,使学生以直观的方式学习获得知识、技能和技巧,符合高职学生的学习特点。

本教材介绍了两种开发工具:一是 JCreator,该工具适合单独的可运行的程序,打开、编写、运行方便,建议在前两章使用;二是 Eclipse,这是在进行企业级开发时最流行的开发工具,从第 3 章开始使用 Eclipse 环境可以使学生熟练掌握 Eclipse 的用法,为将来 Java 后续课程打好基础。

本书共分 11 章,每章开始是知识引入,介绍本章学习什么知识,通过什么情景学这些知识,并通过相应的程序和运行结果来介绍典型问题如何解决,涉及哪些知识点。也就是说通过知识引入的例子了解本章所学的所有内容。

第 1 章主要介绍 Java 语言的开发和运行环境,讲解 Java 程序的书写规范、面向对象的概念等。第 2 章主要介绍 Java 语言的基本知识,包括数据类型、运算符、流程控制语句和数组等。第 3 章主要介绍面向对象编程的基本内容,包括类和对象的概念及使用等。第 4 章主要介绍继承和多态,包括子类的定义以及重载、覆盖及隐藏等多态特性。第 5 章介绍抽象类和接口。第 6 章讲述异常的捕获和处理。第 7 章介绍图形用户界面的设计,包括常用组件和组件的事件响应,使读者可以轻松设计出可实际应用的图形界面。第 8 章讲述数据库和 Java 相结合的应用设计,通过 Java 语言来访问数据库系统。第 9 章介绍输入/输出流的基本知识。第 10 章介绍多线程的基本概念。第 11 章介绍一个完整的小型项目的设计及开发。通过这个项目,学生可以了解整个项目的设计和开发过程,使读者



充分掌握和巩固所学知识,掌握 Java 编程技巧,激发学习兴趣,为今后的学习打下坚实的基础。

本书由刘妍东担任主编,郭亚东担任副主编。其中刘妍东编写了本书的第 3~5 章、第 7 章、第 8 章和第 11 章,郭亚东编写了本书的第 1 章、第 2 章、第 6 章、第 9 章和第 10 章。全书由刘妍东统稿,杨显光审稿。由于编者水平有限,书中难免有不妥之处,望各位专家和读者批评指正。

编 者

2013 年 11 月

目 录

CONTENTS

第 1 章 Java 语言概述	1
1.1 什么是 Java 语言	1
1.1.1 Java 语言的发展历史	1
1.1.2 Java 语言的特点	2
1.1.3 Java 虚拟机	3
1.1.4 垃圾收集机制	4
1.2 面向对象编程初步	4
1.2.1 面向对象的概念	4
1.2.2 抽象	5
1.3 Java 程序结构及书写规范	5
1.3.1 Java 程序结构	5
1.3.2 Java 书写规范	6
1.4 Java 开发和运行环境	7
1.4.1 Java 开发运行环境的建立	7
1.4.2 JCreator 开发运行环境	8
1.4.3 MyEclipse 集成开发环境	11
小结	14
习题	14
技能训练	16
第 2 章 Java 基础编程	17
2.1 标识符与关键字	19
2.1.1 标识符	19
2.1.2 关键字	19

2.2	数据类型与常量、变量	20
2.2.1	数据类型	20
2.2.2	常量与变量	21
2.3	运算符与表达式	23
2.3.1	运算符	23
2.3.2	表达式	26
2.4	流程控制语句	27
2.4.1	分支语句	27
2.4.2	循环语句	30
2.4.3	跳转语句	33
2.5	数组	34
2.5.1	一维数组的声明和创建	34
2.5.2	一维数组的使用	36
2.5.3	二维数组和多维数组	37
小结		38
习题		38
技能训练		43
第3章	类和对象	44
3.1	类	47
3.1.1	类的创建	47
3.1.2	成员变量	48
3.1.3	成员方法	49
3.1.4	构造方法	51
3.2	对象	52
3.2.1	对象的创建	52
3.2.2	使用对象	53
3.3	静态修饰符	56
3.3.1	静态域	56
3.3.2	静态方法	57
小结		57
习题		57
技能训练		59
第4章	继承与多态	60
4.1	创建子类	62
4.2	this 和 super	65
4.2.1	this	65

4.2.2	super	67
4.3	Java 的多态性	69
4.3.1	方法的重载	70
4.3.2	方法的覆盖	72
4.4	包	75
4.4.1	创建包	75
4.4.2	包的导入	75
4.5	访问控制修饰符	77
4.6	final 修饰符	79
4.6.1	最终变量	79
4.6.2	最终方法	80
4.6.3	最终类	80
小结		81
习题		81
技能训练		83
第 5 章	抽象类与接口	84
5.1	抽象类	86
5.2	接口	89
5.2.1	接口声明	89
5.2.2	接口的实现	90
5.2.3	接口的继承	91
小结		93
习题		94
技能训练		96
第 6 章	异常处理	97
6.1	异常类	99
6.1.1	异常的概念	99
6.1.2	异常的分类	99
6.2	异常的抛出和处理	101
6.2.1	try ...catch 语句	101
6.2.2	throw 手动抛出异常	104
6.2.3	throws 声明抛出异常	105
6.3	自定义异常	107
小结		109
习题		109
技能训练		114



第 7 章 图形用户界面设计	115
7.1 Swing 和 AWT	117
7.2 框架与面板	119
7.2.1 JFrame	119
7.2.2 JPanel	121
7.3 布局管理器	121
7.3.1 FlowLayout	121
7.3.2 BorderLayout	123
7.3.3 GridLayout	125
7.3.4 CardLayout	126
7.4 组件	128
7.4.1 按钮 JButton	128
7.4.2 标签 JLabel	129
7.4.3 文本框 JTextField	130
7.4.4 密码框 JPasswordField	132
7.4.5 文本区 JTextArea	133
7.4.6 单选按钮 JRadioButton	135
7.4.7 复选框 JCheckBox	136
7.4.8 组合框 JComboBox	137
7.4.9 列表框 JList	138
7.5 事件处理机制	139
7.6 菜单设计	145
7.6.1 普通菜单	146
7.6.2 弹出菜单	148
小结	150
习题	150
技能训练	152
第 8 章 数据库程序设计	153
8.1 JDBC 驱动	153
8.2 ODBC 数据源设置	154
8.3 JDBC API 的使用	157
8.3.1 加载驱动程序	157
8.3.2 JDBC 常用类和接口	157
8.4 用户登录程序	164
小结	166
习题	166

技能训练	167
第 9 章 输入与输出	168
9.1 Java I/O 简介	169
9.2 File 类	172
9.3 字节流	174
9.3.1 FileInputStream 和 FileOutputStream 类	174
9.3.2 DataInputStream 和 DataOutputStream 类	176
9.3.3 BufferedInputStream 和 BufferedOutputStream 类	178
9.4 字符流	179
9.4.1 FileReader 和 FileWriter 类	179
9.4.2 BufferedReader 和 BufferedWriter 类	180
小结	182
习题	182
技能训练	184
第 10 章 多线程技术	185
10.1 线程的基本概念	186
10.2 线程的创建	189
10.2.1 用 Thread 类建立多线程	190
10.2.2 用 Runnable 接口建立多线程	191
10.3 线程的同步	192
10.3.1 定义同步方法	192
10.3.2 使用同步	194
小结	196
习题	196
技能训练	202
第 11 章 人员信息系统设计	203
11.1 系统简介	203
11.1.1 开发背景	203
11.1.2 需求分析	203
11.2 系统数据库的创建	203
11.3 系统数据库事务的处理	204
11.4 登录功能实现	206
11.4.1 登录界面设计	206
11.4.2 登录事务处理	208
11.5 员工查看自己信息的实现	209

11.5.1 员工界面设计	209
11.5.2 员工查看信息实现	210
11.6 管理员操作模块实现	211
11.6.1 管理员操作界面	211
11.6.2 添加员工界面设计	213
11.6.3 添加员工实现	214
11.6.4 查询指定员工界面设计	215
参考文献	217

第 1 章

Java语言概述

学习目标

通过本章的学习,了解 Java 的发展历史、Java 的特点及编程规范,掌握 Java 的开发、运行环境、开发工具的使用及 Java 程序的基本结构,从而对 Java 面向对象的编程思想、类及对象的概念有一定的认识,为后续学习打下基础。

能力目标

- 了解 Java 的发展历史、Java 的特点;
- 能下载、安装、配置 Java 运行环境;
- 掌握软件开发中的书写规范;
- 熟悉 Java 语言的开发和运行环境;
- 能进行简单 Java 程序的编写、编译与运行。

在进行程序开发前,首先要考虑的是选择一种程序开发语言,那么为什么要选择这种语言呢?它有什么样的特点?它的开发环境又是什么?它使用什么样的开发工具呢?带着种种问题,进入 Java 的学习天地,来了解一下 Java 的发展历史、Java 的特点、运行环境及开发工具,并熟悉 Java 程序的结构、Java 程序的编写、编译及运行的整个过程。

1.1 什么是 Java 语言

Java 是目前最流行的一门编程语言,要学习 Java 语言,必须先了解 Java 的整体概况。本章主要介绍 Java 语言的发展历史、安装环境及常用的集成开发工具等。通过本章的学习,使读者对 Java 语言有一个整体的认识。

1.1.1 Java 语言的发展历史

Java 是一种由 Sun 公司开发编程语言。Sun 公司于 1995 年发布了 Java,并直接将其放在互联网上,并开放源代码。而后,Java 语言以其独特的优势迅猛发展,经过短短 10 年多的时间,它已经成为迄今为止最为优秀的面向对象语言之一,其主要发展历程如下。

1995年5月23日,Java语言诞生。

1996年1月,JDK 1.0诞生。JDK指Java开发工具(Java Development Kit),它主要包括Java程序的运行环境和开发工具。

1997年2月18日,JDK 1.1发布。

1998年12月8日,Java 2企业平台J2EE发布。

1999年6月,SUN公司发布Java的3个版本,是针对不同的市场目标和设备进行定位的,标志着Java技术的成熟,其应用扩展到了各个领域。

2000年5月29日,JDK 1.4发布。

2001年9月24日,J2EE 1.3发布。

2002年2月26日,J2SE 1.4发布,自此Java的计算能力有了大幅提升。

2004年9月30日,J2SE 1.5发布,是Java语言的发展史上的又一里程碑事件,为了表示这个版本的重要性,J2SE 1.5更名为J2SE 5.0版。

2005年6月,JavaOne大会召开,SUN公司公开Java SE 6。此时,Java的各种版本已经更名,以取消其中的数字“2”。J2EE更名为Java EE,J2SE更名为Java SE,J2ME更名为Java ME。

2009年12月,Sun公司发布Java EE 6。截至2009年,Java已经发布了一系列的版本,并且每一个版本都有其特有的名字。

目前,Java主要有以下几个版本。

Java SE:标准版,是本书将主要介绍的版本。

Java EE:企业版,通常用于企业级应用系统的开发。

Java ME:精简版,通常用于嵌入式系统开发。

需要说明的是,介绍以上内容的目的是让读者对Java技术的发展历史有一个初步的认识,以便于更好地学习和使用Java语言。

1.1.2 Java语言的特点

Java语言是一种易于编程的、面向对象的语言。它与其他语言不同,与机器硬件无关,是解释执行的,具有可移植性、安全性、健壮性,提供并发机制,可以最大限度地利用网络。对于一些术语,读者现在也许还不太熟悉,但随着学习的深入,将会逐渐对它们有较深入的理解。Java语言的主要特点如下。

1. 面向对象

Java语言是一种完全面向对象的程序设计语言。面向对象就是以对象为中心,通过对象之间的调用来解决问题的编程方法,它不支持类似于C语言面向过程的程序设计技术。面向对象其实是现实世界模型的自然延伸。现实世界中任何实体都可以看做一个对象,任何对象都是某一类事物的实例,任何对象都具有状态和行为两大特征,而且对象之间可通过消息相互作用。

2. 简单性

Java语言是一种面向对象的语言,它虽然简单,却很高效,可通过提供最基本的方法来完成指定的任务,而且只需理解一些基本的概念,就可以用它编写出适合于各种情况的

应用程序。Java 略去了运算符重载、多重继承等模糊的概念,并且通过实现自动垃圾收集大大简化了程序设计者的内存管理工作。它与 C++ 类似,如果读者了解 C++ 和面向对象的概念,就可以很快编写出 Java 程序。

3. 平台无关性

平台无关性是指 Java 能运行于不同的平台。使用 Java 语言编写的应用程序不用修改,就可运行在不同的软硬件平台上。这主要是由编译器将 Java 程序编译为与机器无关的字节码,这些字节代码被定义为不依赖任何硬件平台和操作系统,当运行程序时通过 Java 虚拟机(JVM)将字节码翻译成机器语言来实现。

4. 健壮性

Java 在编译和运行程序时,都要对可能出现的问题进行检查,以消除 bug 的产生。它提供自动垃圾收集来进行内存管理。在编译时,Java 提示出可能出现但未被处理的例外情况,帮助程序员正确地进行选择以防止系统的崩溃。Java 可捕获类型声明中的许多常见错误,防止动态运行时不匹配问题的出现。

5. 可移植性

与平台无关的特性使 Java 程序可以在配备了 Java 解释器和运行环境的任何计算机系统上运行,方便地被移植到网络上的不同机器。同时,Java 的类库中也实现了与不同平台的接口,使这些类库可以移植。另外,Java 编译器是由 Java 语言实现的,Java 运行时系统由标准 C 实现,这使得 Java 系统本身也具有可移植性。

6. 安全性

由于 Java 语言主要用于网络应用程序的开发,所以对安全性有较高的要求。Java 语言通过使用编译器和解释器,在很大程度上避免了病毒程序的产生和网络程序对本地系统的破坏。Java 是强类型的语言,要求显式的方法声明,这保证了编译器可以发现方法调用错误,保证程序更加可靠。Java 解释器在运行时实施检查,可以发现数组和字符串访问的越界问题。Java 不支持指针功能,从而杜绝了内存的非法访问操作。Java 提供了异常处理机制,程序员可以把一组错误代码放在一个地方,这样可以简化错误处理任务便于恢复。

7. 动态性

Java 的动态特性是面向对象程序设计方法的拓展,它允许程序动态地装入运行过程中所需要的类,在类库中可以自由地加入新的方法和实例变量而不会影响用户程序的执行,并且 Java 通过接口来支持多重继承,使之比严格的类继承具有更灵活的方式和扩展性。

8. 多线程

多线程是指在一个程序中可以同时执行多个任务。多线程机制使应用程序能够并发执行,而且同步机制保证了对共享数据的正确操作。通过使用多线程,程序设计者可以分别用不同的线程完成特定的行为,而不需要采用全局的事件循环机制,这样就很容易地实现网络上的实时交互行为。

1.1.3 Java 虚拟机

Java 语言的一个非常重要的特点就是与平台的无关性,而使用 Java 虚拟机是实现这一特点的关键。所谓 Java 虚拟机 JVM,是一台可以存在于不同的、真实的软、硬件环境

下的虚拟计算机,其功能是将字节码解释为真实平台能执行的指令。它的工作原理是:Java 虚拟机处于机器和编译程序之间,在任何平台上都提供给编译程序一个共同的接口。Java 源程序(.java)经过编译器编译后变成字节码(.class),字节码由虚拟机解释执行,虚拟机将每一条要执行的字节码送给解释器,解释器将其翻译成特定机器上的机器码(.exe),然后在特定的机器上运行。Java 正是通过 JVM 技术,实现了与平台无关,“编写一次,到处运行。”因此,任何平台只要安装相应的 Java 虚拟机环境,就能运行 Java 程序。

在此,要弄清楚两个概念,即编译型语言和解释型语言。首先肯定的是,Java 是解释型语言。

注意

(1) 编译型语言:把做好的源程序全部翻译成二进制代码的可运行程序,然后可直接运行这个程序,如 C、C++ 等语言。

(2) 解释型语言:把做好的源程序翻译一句,然后执行一句。Java 很特殊,Java 程序也需要编译,但第一次编译是编译成自己的 .class 文件,当 .class 文件需要执行时,由 Java 虚拟机用解释方式执行字节码。

1.1.4 垃圾收集机制

垃圾收集机制是一种动态存储管理技术,它自动地释放不再被程序引用的对象,按照特定的垃圾收集算法来实现资源自动回收的功能。

在 C++ 中,对象所占的内存在程序结束运行前一直被占用,在明确释放前不能分配给其他对象;而在 Java 中,Java 语言建立了垃圾收集机制。由于 Java 语言是面向对象的编程语言,其程序以类为单位,在程序运行期间会在内存中创建很多类的对象。这些对象在完成任务后,Java 的垃圾回收机制会自动释放这些对象所占用的空间,使回收的内存能被再次利用,提高程序的运行效率。垃圾回收不仅可以提高系统的可靠性、使内存管理与类接口设计分离,还可以使开发者减少了跟踪内存管理错误的时间,从而把程序员从手工回收内存空间的繁重工作中解脱出来。

1.2 面向对象编程初步

Java 是纯面向对象的编程语言。要想学好 Java,首先要建立起面向对象的思想,这对深入理解 Java 语言大有益处。下面我们首先介绍一下面向对象的基本概念。

1.2.1 面向对象的概念

面向对象程序设计是一种全新的程序设计理念,面向对象就是以对象为中心,现实世界中任何实体都可以看作是一个对象,如一辆汽车、一辆卡车、一本书、一支笔、一名学生、一名教师等都是一个对象。任何对象都是某一类事物的实例,人们将其共同的特征抽取出来,就可以将其归纳成一类。例如,汽车、卡车可以归纳为“车类”;书、笔可以归纳为“文具类”;学生、教师可以归纳为“人类”。因此,面向对象程序设计就是设计和定义这些类,定义好的类可以作为一个具体的数据类型进行类的实例化操作。通过类的实例化操

作,就可以得到一系列具有通用特征和行为的对象。对于有关类和对象的概念,将在后续的章节中具体介绍。

1.2.2 抽象

顾名思义,抽象就是抽掉了具体形象的东西。面向对象程序设计的基本要素就是抽象,程序员通过抽象来管理复杂性。例如“人”就可以说是一个相对抽象的概念,因为一个具体的人具有姓名、年龄、身高、文化程度、社会关系等具体特征,但抽象的“人”就是不包含这些具体特征的一个概念,因此说它是抽象的;而具体的人的概念应该是张三、李四等。因此,抽象概念一般是指每一类事物,而不是指某一个事物。面向对象程序设计思想鼓励人们用抽象的观点来看待现实世界。也就是说,现实世界是由一组抽象的对象——类组成的。

1.3 Java 程序结构及书写规范

本节主要通过两个简单的程序实例,使人们对 Java 程序的结构有一个初步的认识。接下来介绍 Java 的程序书写规范,书写规范其实很重要,从现在开始就要注意程序的规范化,以增加程序的可读性。

1.3.1 Java 程序结构

Java 程序可以分为两类,一类是 Java Application,即 Java 应用程序;另一类是 Java Applet,即 Java 小程序。

1. Java Application

Java 应用程序是完整的程序,不需要其他语言的帮助,就可以独立运行。其程序结构如下。

例 1-1 MyApp.java

```
public class MyApp
{
    public static void main(String args[])
    {
        System.out.println("这是 Java Application 程序。");
    }
}
```

2. Java Applet

Java Applet 是嵌在 HTML 编写的 Web 页面中的非独立程序,由 Web 浏览器内包含的 Java 编译器来编译执行。其程序结构如下。

例 1-2 MyApplet.java

```
import java.awt.*;
import java.applet.*;
public class MyApplet extends Applet
{
```