

实例讲解  
实训强化  
培养技能  
面向就业

全国高等职业教育计算机类规划教材 · 实例与实训教程系列

# Java程序设计

◎ 杨旭超 主编    ◎ 王伟民 刘 冰 孙 麒 诸葛燕 副主编



- ◆ 遵循“循序渐进”和“理论联系实际”的教学原则
- ◆ 彰显“面向对象程序设计”的思想、方法和技巧
- ◆ 突出重点、分散难点，揭示知识之间的内在联系
- ◆ 以学生为本位、面向应用，推动理论向实践转化的能力教育



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

2000-2001 2001-2002  
 2002-2003 2003-2004  
 2004-2005 2005-2006  
 2006-2007 2007-2008

# Java程序设计



**Abstract**

全国高等职业教育计算机类规划教材·实例与实训教程系列

# Java 程序设计

杨旭超 主 编

王伟民 刘 冰 孙 麒 诸葛燕 副主编

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书内容分为三个部分：第一部分主要介绍 Java 语言的开发环境、基本数据类型和基本语法知识，便于读者了解与掌握 Java 语言基础知识，为后续学习打基础。第二部分介绍 Java 语言面向对象的编程思想、方法和技巧，内容包括 Java 面向对象技术的基础理论、主要原则和思维方式，类与对象机制、包、接口、系统类库、最终类、抽象类、Java 常用工具，Applet 与图形用户界面、事件处理、多媒体编程等。第三部分内容包括 Java 语言的异常处理、多线程编程、网络编程、数据库编程等。

本书可以作为高等院校、高职高专类院校计算机专业教学用书，也可作为初学者自学用书。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有，侵权必究。

## 图书在版编目(CIP)数据

Java 程序设计 / 杨旭超主编. —北京：电子工业出版社，2009.1

全国高等职业教育计算机类规划教材·实例与实训教程系列

ISBN 978-7-121-07884-2

I. J… II. 杨… 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（2008）第 184103 号

策划编辑：左 雅

责任编辑：程超群

印 刷：北京市海淀区四季青印刷厂

装 订：三河市万和装订厂

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×1 092 1/16 印张：22.25 字数：566.4 千字

印 次：2009 年 1 月第 1 次印刷

印 数：4 000 册 定价：32.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 [zlts@phei.com.cn](mailto:zlts@phei.com.cn)，盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

服务热线：（010）88258888。

## 序

20 世纪 90 年代以来,以计算机和通信技术为推动力的信息产业在我国获得了前所未有的发展,全国各企事业单位对信息技术人才求贤若渴,高等教育计算机及相关专业毕业生供不应求。随后几年,我国各高等院校、众多培训机构相继开设计算机及相关专业,积极扩大招生规模,不久即出现了计算机及相关专业毕业生供大于求的局面。纵观近十年的就业市场变化,计算机专业毕业生经历了“一夜成名、求之不得”的宠幸,也遭遇了“千呼百应,尽失风流”的冷落。

这个时代深深地镌刻着信息的烙印,这个时代是信息技术人才尽情展示才能的舞台。目前我国的劳动力市场,求职人数过剩,但满足企业要求的专业人才又很稀缺。这种结构性的人才市场供求矛盾是我国高等教育亟待解决的问题,更是“以人为本,面向人人”为目标的职业教育不可推卸的责任。

电子工业出版社,作为我国出版职业教育教材最早的出版社之一,是计算机及相关专业高等职业教材重要的出版基地。多年来,我们一直在教材领域为战斗在职业教育第一线的广大职业院校教育工作者贡献着我们的力量,积累了丰富的职业教材出版经验。目前,计算机专业高等教育正处于发展中的关键时期,我们有义务、有能力协同全国各高等职业院校,共同探寻适合社会发展需要的人才培养模式,建设满足高等职业教育需求的教学资源——这是我们出版“全国高等职业教育计算机类规划教材·实例与实训教程系列”的初衷。

关于本系列教材的出版,我们力求做到以下几点:

(1) 面向社会人才市场需求,以培养学生技能为目标。工学结合、校企结合是职业教育发展的客观要求,面向就业是职业教育的根本落脚点。本系列教材内容体系的制定是广大高职教育专家、一线高职教师共同智慧的结晶。我们力求教材内容丰富而不臃肿、精简而不残缺,实用为主、够用为度。

(2) 面向高职学校教师,以方便教学为宗旨。针对每个课程的教学特点和授课方法,我们为其配备相应的实训指导、习题解答、电子教案、教学素材、阅读资料、程序源代码、电子课件、网站支持等一系列教学资源,广大教师均可从华信教育资源网([www.huaxin.edu.cn](http://www.huaxin.edu.cn))免费获得。

(3) 面向高职学校学生,以易学、乐学为标准。以实例讲述理论、以项目驱动教学是本系列教材的显著特色。这符合现阶段我国高职学生的认知规律,能够提高他们的学习兴趣,增强他们的学习效果。

这是一个崭新的开始,但永远没有尽头。高等职业教育教材的建设离不开广大职业教育工作者的支持,尤其离不开众多高等职业院校教师的支持。我们诚挚欢迎致力于职业教育事业发展的有识之士、致力于高等职业教材建设的有才之士加入到我们的队伍中来,多批评,勤点拨,广结友,共繁荣,为我国高等职业教育的发展贡献我们最大的力量!

电子工业出版社高等职业教育分社

# 前 言

Java 语言作为一种优秀的现代高级编程语言,除继承了 C++ 语言以及其他高级编程语言的优点之外,还体现和发展了当代软件技术的新成果。特别是 Java 语言内置了多线程和网络应用的功能,使得 Java 语言很好地顺应了 Internet 的高速发展,从而成为一种跨平台、分布式、适于网络应用的面向对象的新型编程语言。目前,由于 Java 语言无可比拟的优点,Java 语言的应用领域更加广泛,深得软件开发人员的青睐,许多高等院校也将《Java 程序设计》课程作为计算机专业的基础课或必修课。

本书是基于高等院校课程改革的需要进行编写的,书中在考虑学生能力需求的基础上,基础理论知识“以实用为准则,以够用为尺度”,教学内容增加实践性、应用性,旨在推动理论向实践转化的能力教育。

本书内容分为三部分:

第一部分主要介绍 Java 语言的开发环境、基本数据类型和基本语法知识,便于读者了解与掌握 Java 语言基础知识,为后续学习打基础。

第二部分介绍 Java 语言面向对象的编程思想、方法和技巧,主要内容包括 Java 面向对象技术的基础理论、主要原则和思维方式,类与对象机制、包、接口、系统类库、最终类、抽象类、Java 常用工具,Applet 与图形用户界面、事件处理、多媒体编程等。

第三部分内容包括 Java 语言的异常处理、多线程编程、网络编程、数据库编程等。

教材内容由浅入深、循序渐进,书中所用实例通俗易懂,体现程序设计中最基本的思想和方法。教材内容注重理论与实践的结合,对一些必需的、基础的和常用的 Java 语言概念或规则,尽可能分散穿插在算法设计的过程中进行渗透,有利于分散知识难点,符合教学规律。

教材旨在培养学生良好的程序设计风格,拓展学生的逻辑思维能力,提高学生分析问题、解决问题的能力以及实际动手的能力。本书可以作为高等院校、高职高专类院校计算机专业教学用书,也可作为初学者自学用书。

本书由杨旭超主编,王伟民、刘冰、孙麒、诸葛燕为副主编。书中第 8、10 章由王伟民编写,第 4、5、9 章由刘冰编写,第 11、12 章由孙琪编写,第 1 章由诸葛燕编写,第 2、3、6、7 章由杨旭超编写,河南商业高等专科学校蔡中民老师参加了以上章节部分内容的编写。全书由杨旭超和王伟民统稿。

由于编者水平有限,时间仓促,教材内容可能存在不妥和错误,敬请批评指正。

编 者

2008 年 11 月

## 《Java 程序设计》读者意见反馈表

尊敬的读者：

感谢您购买本书。为了能为您提供更优秀的教材，请您抽出宝贵的时间，将您的意见以下表的方式（可从 <http://edu.phei.com.cn> 下载本调查表）及时告知我们，以改进我们的服务。对采用您的意见进行修订的教材，我们将在该书的前言中进行说明并赠送您样书。

姓名：\_\_\_\_\_ 电话：\_\_\_\_\_

职业：\_\_\_\_\_ E-mail: \_\_\_\_\_

邮编：\_\_\_\_\_ 通信地址：\_\_\_\_\_

1. 您对本书的总体看法是：

☐ 很满意    ☐ 比较满意    ☐ 尚可    ☐ 不太满意    ☐ 不满意

2. 您对本书的结构（章节）：☐ 满意    ☐ 不满意    改进意见\_\_\_\_\_

3. 您对本书的例题    ☐ 满意    ☐ 不满意    改进意见\_\_\_\_\_

4. 您对本书的习题    ☐ 满意    ☐ 不满意    改进意见\_\_\_\_\_

5. 您对本书的实训    ☐ 满意    ☐ 不满意    改进意见\_\_\_\_\_

6. 您对本书其他的改进意见：

7. 您感兴趣或希望增加的教材选题是：

请寄：100036 北京万寿路 173 信箱高等职业教育分社收

电话：010-88254571    E-mail: gaozhi@phei.com.cn

# 目 录

|                           |      |
|---------------------------|------|
| 第 1 章 Java 语言基础 .....     | (1)  |
| 1.1 Java 语言概述 .....       | (1)  |
| 1.2 Java 开发运行环境 .....     | (2)  |
| 1.2.1 Java 开发工具 SDK ..... | (2)  |
| 1.2.2 Java 程序开发步骤 .....   | (2)  |
| 1.2.3 初识 Java 应用程序 .....  | (3)  |
| 1.2.4 初识 Java 小应用程序 ..... | (4)  |
| 1.3 简单的 Java 程序设计 .....   | (6)  |
| 1.3.1 标识符与关键字 .....       | (6)  |
| 1.3.2 简单数据类型 .....        | (7)  |
| 1.3.3 运算符与表达式 .....       | (9)  |
| 1.3.4 流程控制语句 .....        | (13) |
| 1.3.5 Java 的输入/输出初步 ..... | (21) |
| 1.3.6 简单数组应用 .....        | (24) |
| 本章小结 .....                | (28) |
| 课外阅读 Eclipse 简介 .....     | (29) |
| 复习思考题 .....               | (30) |
| 上机操作题 .....               | (33) |
| 第 2 章 类与对象 .....          | (35) |
| 2.1 对象 .....              | (35) |
| 2.2 类 .....               | (36) |
| 2.2.1 类的建立 .....          | (36) |
| 2.2.2 类的声明 .....          | (36) |
| 2.2.3 类的成员方法 .....        | (37) |
| 2.2.4 类的构造函数 .....        | (40) |
| 2.3 类的实例化 .....           | (41) |
| 2.4 成员变量与局部变量 .....       | (43) |
| 2.5 方法调用与方法重载 .....       | (48) |
| 2.6 对象应用 .....            | (50) |
| 2.6.1 对象的内存模型 .....       | (50) |
| 2.6.2 对象的使用 .....         | (52) |
| 2.7 面向对象编程举例 .....        | (57) |
| 2.8 static 关键字 .....      | (61) |
| 2.9 包 .....               | (67) |

|                                       |              |
|---------------------------------------|--------------|
| 2.9.1 包的创建与作用 .....                   | (67)         |
| 2.9.2 包的引用 .....                      | (69)         |
| 2.9.3 环境变量 CLASSPATH 的设置 .....        | (75)         |
| 2.10 访问权限 .....                       | (75)         |
| 2.10.1 共有变量和共有方法 .....                | (75)         |
| 2.10.2 私有变量和私有方法 .....                | (76)         |
| 2.10.3 友好变量和友好方法 .....                | (78)         |
| 2.10.4 受保护的成员变量和方法 .....              | (80)         |
| 2.10.5 public 类与友好类 .....             | (82)         |
| 2.11 类的封装性 .....                      | (82)         |
| 本章小结 .....                            | (84)         |
| 课外阅读 类 .....                          | (85)         |
| 复习思考题 .....                           | (85)         |
| 上机操作题 .....                           | (86)         |
| <b>第3章 类的继承性与多态性</b> .....            | <b>(89)</b>  |
| 3.1 类的继承性 .....                       | (89)         |
| 3.1.1 创建子类 .....                      | (89)         |
| 3.1.2 子类的继承性 .....                    | (91)         |
| 3.1.3 关键字 super 的使用 .....             | (95)         |
| 3.1.4 类成员的覆盖 .....                    | (101)        |
| 3.2 类的多态 .....                        | (105)        |
| 3.2.1 成员方法的重载 .....                   | (105)        |
| 3.2.2 构造方法的重载 .....                   | (107)        |
| 3.3 类与类之间的数据类型转换 .....                | (110)        |
| 本章小结 .....                            | (111)        |
| 课外阅读 类的继承性和多态性 .....                  | (111)        |
| 复习思考题 .....                           | (112)        |
| 上机操作题 .....                           | (113)        |
| <b>第4章 最终类、抽象类、接口</b> .....           | <b>(115)</b> |
| 4.1 最终类 (final) 与抽象类 (abstract) ..... | (115)        |
| 4.1.1 final 类和 final 方法 .....         | (115)        |
| 4.1.2 abstract 类和 abstract 方法 .....   | (115)        |
| 4.2 接口 .....                          | (117)        |
| 4.2.1 接口的声明与实现 .....                  | (117)        |
| 4.2.2 事项接口的继承与组合 .....                | (119)        |
| 4.2.3 接口的多态 .....                     | (120)        |
| 本章小结 .....                            | (121)        |
| 复习思考题 .....                           | (121)        |
| 上机操作题 .....                           | (122)        |

|                                 |       |
|---------------------------------|-------|
| <b>第 5 章 常用类</b>                | (124) |
| 5.1 字符串类 String                 | (124) |
| 5.1.1 String 字符串的定义             | (124) |
| 5.1.2 String 字符串的常用方法           | (125) |
| 5.1.3 使用 StringTokenizer 类分析字符串 | (130) |
| 5.2 字符串缓存类 StringBuffer         | (131) |
| 5.2.1 StringBuffer 字符串的定义       | (132) |
| 5.2.2 StringBuffer 的常用方法        | (132) |
| 5.3 String 类与 StringBuffer 类的区别 | (135) |
| 5.4 main()方法的参数                 | (136) |
| 5.5 数学类 Math                    | (137) |
| 5.6 随机数类 Random                 | (139) |
| 5.7 向量类 (Vector)                | (140) |
| 本章小结                            | (142) |
| 复习思考题                           | (142) |
| 上机操作题                           | (143) |
| <b>第 6 章 GUI 图形用户界面</b>         | (146) |
| 6.1 AWT 类库与 Swing 类库            | (146) |
| 6.1.1 容器、窗口、面板                  | (146) |
| 6.1.2 文本、字体、颜色                  | (153) |
| 6.1.3 标签组件                      | (157) |
| 6.2 Java Applet                 | (159) |
| 6.2.1 Applet 的生命周期              | (159) |
| 6.2.2 Applet 程序结构               | (161) |
| 6.2.3 按钮组件                      | (162) |
| 6.3 布局管理器                       | (166) |
| 6.3.1 顺序布局                      | (167) |
| 6.3.2 边界布局                      | (168) |
| 6.3.3 卡片布局                      | (169) |
| 6.3.4 网格布局                      | (170) |
| 6.3.5 手工布局                      | (171) |
| 6.4 GUI 标准组件                    | (172) |
| 6.4.1 文本框                       | (172) |
| 6.4.2 文本区                       | (175) |
| 6.4.3 选项框                       | (178) |
| 6.4.4 复选框与选项按钮                  | (179) |
| 6.4.5 列表框                       | (181) |
| 6.4.6 滚动条                       | (182) |
| 6.5 Java 的事件处理机制                | (183) |
| 6.5.1 事件处理概述                    | (183) |

|                                             |              |
|---------------------------------------------|--------------|
| 6.5.2 窗口事件 .....                            | (185)        |
| 6.5.3 适配器类 .....                            | (191)        |
| 6.5.4 鼠标事件 .....                            | (191)        |
| 6.5.5 焦点事件 .....                            | (197)        |
| 6.5.6 键盘事件 .....                            | (200)        |
| 本章小结 .....                                  | (207)        |
| 课外阅读 Applet 标签属性及 HTML 和 Applet 的参数传递 ..... | (208)        |
| 复习思考题 .....                                 | (210)        |
| 上机操作题 .....                                 | (212)        |
| <b>第 7 章 菜单与对话框 .....</b>                   | <b>(215)</b> |
| 7.1 菜单 .....                                | (215)        |
| 7.1.1 建立菜单 .....                            | (215)        |
| 7.1.2 菜单的功能设置 .....                         | (218)        |
| 7.2 自定义对话框 .....                            | (225)        |
| 本章小结 .....                                  | (227)        |
| 复习思考题 .....                                 | (227)        |
| 上机操作题 .....                                 | (227)        |
| <b>第 8 章 Java 图形图像处理 .....</b>              | <b>(229)</b> |
| 8.1 简单图形 .....                              | (229)        |
| 8.2 画布 (Canvas) .....                       | (233)        |
| 8.3 在 Applet 中加载图像 .....                    | (234)        |
| 8.4 在 Applet 中加载数字音频 .....                  | (235)        |
| 课外阅读 随机画图 .....                             | (237)        |
| 复习思考题 .....                                 | (238)        |
| 上机操作题 .....                                 | (239)        |
| <b>第 9 章 异常处理 .....</b>                     | <b>(241)</b> |
| 9.1 异常处理概述 .....                            | (241)        |
| 9.2 异常处理的语法规则 .....                         | (242)        |
| 9.2.1 try-catch-finally 语句 .....            | (242)        |
| 9.2.2 throw 语句和 throws 语句 .....             | (245)        |
| 9.3 创建自定义异常 .....                           | (248)        |
| 本章小结 .....                                  | (249)        |
| 复习思考题 .....                                 | (249)        |
| 上机操作题 .....                                 | (250)        |
| <b>第 10 章 Java 多线程程序设计 .....</b>            | <b>(253)</b> |
| 10.1 多线程 .....                              | (253)        |
| 10.1.1 线程的概念 .....                          | (253)        |
| 10.1.2 创建线程 .....                           | (254)        |
| 10.2 线程的状态与生命周期 .....                       | (256)        |
| 10.3 通过实现 Runnable 接口创建线程 .....             | (257)        |

|                                 |       |
|---------------------------------|-------|
| 本章小结 .....                      | (259) |
| 课外阅读 多线程运用 .....                | (260) |
| 复习思考题 .....                     | (261) |
| 上机操作题 .....                     | (262) |
| <b>第 11 章 网络编程</b> .....        | (264) |
| 11.1 TCP/IP 协议 .....            | (264) |
| 11.2 基于 TCP 的 Socket 网络编程 ..... | (265) |
| 11.2.1 Socket 通信 .....          | (265) |
| 11.2.2 基于 TCP 的通信程序设计 .....     | (268) |
| 11.3 基于 UDP 的 Socket 网络编程 ..... | (275) |
| 11.3.1 Datagram 通信 .....        | (275) |
| 11.3.2 基于 UDP 的简单通信程序设计 .....   | (278) |
| 本章小结 .....                      | (281) |
| 课外阅读 网络相关知识 .....               | (281) |
| 复习思考题 .....                     | (282) |
| 上机操作题 .....                     | (283) |
| <b>第 12 章 数据库编程</b> .....       | (287) |
| 12.1 JDBC 概述 .....              | (287) |
| 12.1.1 JDBC 的主要类和接口 .....       | (287) |
| 12.1.2 JDBC 驱动程序 .....          | (290) |
| 12.1.3 JDBC URL 的标准语法 .....     | (291) |
| 12.2 JDBC 开发数据库应用 .....         | (291) |
| 12.2.1 开发步骤 .....               | (291) |
| 12.2.2 实现过程 .....               | (292) |
| 12.2.3 数据库应用程序 .....            | (293) |
| 本章小结 .....                      | (300) |
| 复习思考题 .....                     | (301) |
| 上机操作题 .....                     | (302) |
| <b>参考答案</b> .....               | (305) |
| <b>参考文献</b> .....               | (343) |

# 第1章 Java 语言基础

## 应知目标

了解 Java 语言的特点，熟悉 Java 语言开发环境及 Java 程序开发过程，熟练掌握 Java 语言的基础知识。

## 应会目标

熟悉 Java 语言开发环境，能够正确运用 Java 语言的基础知识编写简单程序，并能够正确运用 Java 语言开发工具编译和运行程序。

## 1.1 Java 语言概述

Java 语言始创于 1995 年 6 月，由美国 Sun 公司创建。Java 语言集中体现和充分利用了若干当代软件技术新成果，从而成为一种跨平台、分布式、适于网络应用的面向对象的现代高级编程语言。

Java 具有以下特点：

### 1. 平台无关性

Java 语言的平台无关性是指用 Java 语言编写的应用程序不用修改就可在不同的软、硬件平台上运行。

平台无关有两种：源代码级的平台无关性和目标代码级的平台无关性。C 和 C++ 具有一定程度的源代码级平台无关，Java 主要靠 Java 虚拟机（Java Virtual Machine, JVM）在目标代码级实现平台无关性。

JVM 可以看做是一种抽象机器，它附着在具体操作系统上，是驻存在计算机内存中的虚拟计算机。当 Java 源程序被编译为字节码程序后，JVM 中的 Java 解释器将解释执行 Java 字节码程序。任何一台计算机只要配置了 Java 解释器，就可以运行这个程序，而不管这种字节码是在何种平台上生成的。JVM 是 Java 平台无关性的基础。

Java 语言独特的运行机制使得它具有良好的二进制级的可移植性。利用 Java，开发人员可以编写出与具体平台无关、普遍适用的应用程序，大大降低了开发、维护和管理开销。

### 2. 面向对象

面向对象技术的核心是以更接近于人类思维的方式建立计算机逻辑模型，它利用类和对象的机制将数据与其上的操作封装在一起，并通过统一的接口与外界交互，使反映现实世界实体的各个类在程序中能够独立、自治、继承、多态。这种方法非常有利于提高程序的可维护性和可重用性，而 Java 正是面向对象编程语言的一个典型代表。

### 3. 分布式

分布式包括数据分布和操作分布。数据分布是指数据可以分散在网络的不同主机上；操

作分布是指将一个计算分散在不同主机上处理。

Java 支持 WWW 客户机/服务器计算模式，因此，它支持这两种分布性。对于前者，Java 提供了一个叫做 URL 的对象，利用这个对象，可以打开并访问具有相同 URL 地址上的对象，访问方式与访问本地文件系统相同。对于后者，Java 的 Applet 小程序可以从服务器下载到客户端，即部分计算在客户端进行，提高系统执行效率。

Java 提供了一整套网络类库，开发人员可以利用类库进行网络程序设计。

#### 4. 支持多线程

Java 语言中还提供了专门的支持多线程的 API 库，有效地构建了多线程运行机制。多线程运行即使得程序可以同时运行多个指令流，从而获得更灵活的任务处理能力。

#### 5. 动态性

类是 Java 程序构成的基本单元，而类库是程序中有用的类的集合。程序员可以在类库中加入新方法和实例变量却不影响程序执行。不论是自己编写的类还是从类库中引入，类是运行时动态装载的，这使得 Java 类库在分布环境中动态地支持了 Java 面向对象的程序设计。

#### 6. 简单易学

Java 语言衍生于 C 及 C++，因此 Java 的基本语法与 C 语言几乎一模一样，这样，无论是学过 Java 再学 C，还是已经掌握了 C 再来学 Java，都会感到易于入门。

#### 7. 安全可靠

出于安全性、稳定性的考虑，Java 语言增加了自动内存管理，去除了 C++ 中不容易理解和掌握的部分，如最典型的指针操作等，降低了学习的难度，保证了 Java 程序运行的可靠性。此外，由于 Java 语言在网络应用方面有效的安全措施，也保证了 Java 程序在网络应用方面的安全可靠。

## 1.2 Java 开发运行环境

### 1.2.1 Java 开发工具 SDK

Java 不仅提供了一个丰富的语言和运行环境，而且还提供了一个免费的 Java 软件开发工具箱 SDK（以前称为 Java 开发工具箱 Java Developers Kits，简称 JDK）。目前 Sun 公司发布了 SDK 的 1.4 版本，高级版本可以对低级版本实现向下兼容。可以登录到 Sun 公司的网站（<http://java.sun.com>）免费下载 SDK1.4。

SDK 安装目录的 bin 文件夹中含有编译器（javac.exe）、解释器（java.exe）和一些其他的可执行文件。为了方便地使用编译器和解释器，应当将 bin 所在的目录路径包含在环境变量 path 的设置中。

SDK 包括以下工具：Java 语言编译器 Javac，输出结果为 Java 字节码；Java 字节码解释器 Java；小应用程序浏览工具 Java AppletViewer，用于测试并运行 Java 小应用程序。

### 1.2.2 Java 程序开发步骤

Java 程序的开发通常需要编写源代码、编译生成字节码和运行等几个步骤。

Java 程序开发过程如图 1.1 所示。

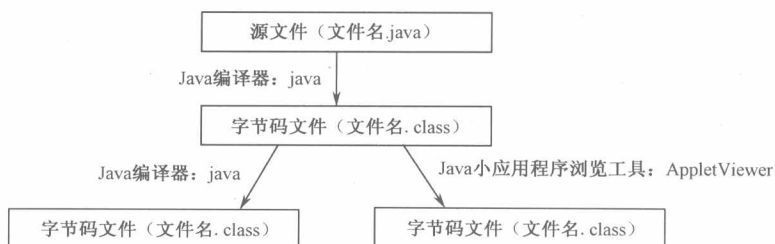


图 1.1 Java 程序开发过程

## 1. 编写源代码

Java 源程序是以.java 为后缀名的简单的文本文件, 可以用各种 Java 集成开发环境中的源代码编辑器来编写, 也可以用其他文本编辑工具, 如 Windows 中的记事本或 DOS 中的 Edit 软件等。

## 2. 编译生成字节码

在 Java 程序中源代码经编译所得的目标代码称为字节码。字节码是二进制文件, 编程人员无法直接读懂, 由 Java 语言的解释器来解释执行。在 SDK 这样的命令行开发工具中需要运行独立的编译程序 javac.exe。

## 3. 运行

Java 程序可以分为 Java Application (应用程序) 和 Java Applet (小应用程序) 两类, 这两类程序的运行方式有很大的差别。Java Application 是由独立的解释器程序来运行的, 需要调用 Java 的解释器软件 java.exe。

Java Applet 是一类非常重要的 Java 程序, 虽然它的源代码编辑与字节码的编译生成过程与 Java Application 相同, 但它却不是一类可以独立运行的程序, 其字节码文件必须嵌入到另一种语言 HTML 的文件中并由负责解释 HTML 文件的 WWW 浏览器充当其解释器, 来解释执行 Java Applet 的字节码程序。

## 1.2.3 初识 Java 应用程序

### 1. 源文件编写

下面是一个简单的 Java Application 程序, 我们通过文本编辑器 (如 Windows 自带的记事本程序 Notepad) 新建一个文件, 输入如下代码:

```

import java.io.*;
public class Hello
{
    public static void main(String rgs[])
    {
        System.out.println("Java, 你好!");
    }
}
  
```

程序中的类: Hello

程序中的方法

输出结果

一个 Java 程序是由若干个类组成的。这个 Java 应用程序简单到只有一个类, 类的名字为 Hello。

上述源文件必须命名为 `Hello.java`（不可写成 `hello.java`，因为 Java 语言区分大小写）。

以上程序中，`class` 是 Java 用来定义类的关键字。`public` 也是关键字，说明 `Hello` 是一个 `public` 类。第一个大括号和最后一个大括号以及它们之间的内容叫做类体。

```
public static void main(String args[] )
{...}
```

是类体中的一个方法，在一个 Java Application 应用程序中，`main` 方法必须被说明为 `public static` 和 `void`；`String args[]` 声明一个字符串类型的数组 `args[]`，它是 `main` 方法的参数。之后的两个大括号之间的内容叫做方法体。

一个 Java Application 应用程序必须有一个类且只有一个类含有这样的 `main` 方法，这个类称为应用程序的主类。`main` 方法是 Java Application 应用程序的标志，也是程序开始执行的入口位置。

注意：在源文件中

1. 如果源文件中有多类，那么只能有一个类是 `public` 类；
2. 如果有一个类是 `public` 类，那么源文件的名称必须与这个类的名称完全相同，扩展名是 `.java`；
3. 如果源文件没有 `public` 类，那么源文件的名称只要和某个类的名称相同，并且扩展名是 `.java` 就可以了。

## 2. 程序编译

我们使用 Java 编译器（`javac.exe`）对源文件 `Hello.java` 进行编译。不妨假设源文件已经保存到路径 `C:\javabook` 中，则在命令提示符环境下，进入当前源文件所在目录，并输入命令：

```
C:\javabook>javac Hello.java
```

编译成功后，在同一目录 `C:\javabook` 下生成字节码文件 `Hello.class`。如果 Java 源程序中包含了多个类，在编译成功后，每个类都会在源文件所在目录中生成自己的字节码文件。这些字节码文件的文件名与该类的名称相同，并且都以 `.class` 作为后缀。

需要注意的是，如果对源文件进行了修改，就要对其重新编译，生成新的字节码文件。

## 3. 程序运行

使用 Java 解释器（`java.exe`）运行这个应用程序：

```
C:\javabook>java Hello
```

程序运行结果：

```
Java, 你好!
```

当 Java 应用程序中有多个类时，必须运行包含了 `main` 方法的那个类，即主类。

## 1.2.4 初识 Java 小应用程序

### 1. 源文件编写

`Student.java` 程序：

```

import java.applet.*;
import java.awt.*;
public class Student extends Applet
{
    public void paint(Graphics g)
    {g.setColor(Color.red);
    g.drawString("我用 Java 写出红色的字",5,30);
    g.setColor(Color.green);
    g.drawString("现在变成绿色的字了，真有意思！",15,50);
    }
}

```

用 import 语句引入所需要的包: java.applet 和 java.awt

g 为画笔对象

定义画笔颜色

定义字符串, 规定了书写位置。

一个 Java Applet 也是由若干个类组成的。Java Applet 程序中不需要有 main 方法，但是 Java Applet 程序的主类必须是类库中已定义好的类 java.applet.Applet 的子类。Java Applet 的主类必须是 public 的。

**注意：**Java 源程序中的语句所涉及到的的小括号及标点符号都是英文状态下输入的括号和标点符号，而字符串中的符号不受汉字符号或英文符号的限制。

## 2. 程序编译

与 Java 应用程序类似，源文件必须命名为 Student.java，不妨假设源文件已经保存到路径 C:\javabook 中，则源文件的编译命令如下：

```
C:\javabook>javac Student.java
```

若编译成功，在 C:\javabook 下会出现字节码文件 Student.class。

## 3. HTML 文件编写

Applet 程序的执行方式与 Java Application 程序不同，Applet 程序的执行需要依赖支持 Java 的 WWW 浏览器充当其解释器，来解释执行 Java Applet 的字节码程序，所以必须编写 HTML（超文本标记语言）文件，将 Applet 字节码文件嵌入到 HTML 文件中。

最简单的 HTML 文件如下（也可以只写第四行）：

```

<html>
<head><title></title></head>
<body>
<applet code =Student.class height=100 width=300></applet>
</body>
</html>

```

我们仍然可以用记事本程序书写以上 HTML 文件，并将其保存为 Student.html 文件名（后缀 html 不能变，但是文件名可以是任意名字）。将此 HTML 文件保存到与 Student.class 同一个目录中，即 C:\javabook。