



21世纪高校计算机应用技术系列规划教材

谭浩强 主编

# Java语言程序设计

李尊朝 苏军 编著

★本书通过对Java语言的全面介绍,

使读者学会运用面向对象方法,

分析和解决实际问题的能力。

★本书在介绍Java编程原理的前提下,

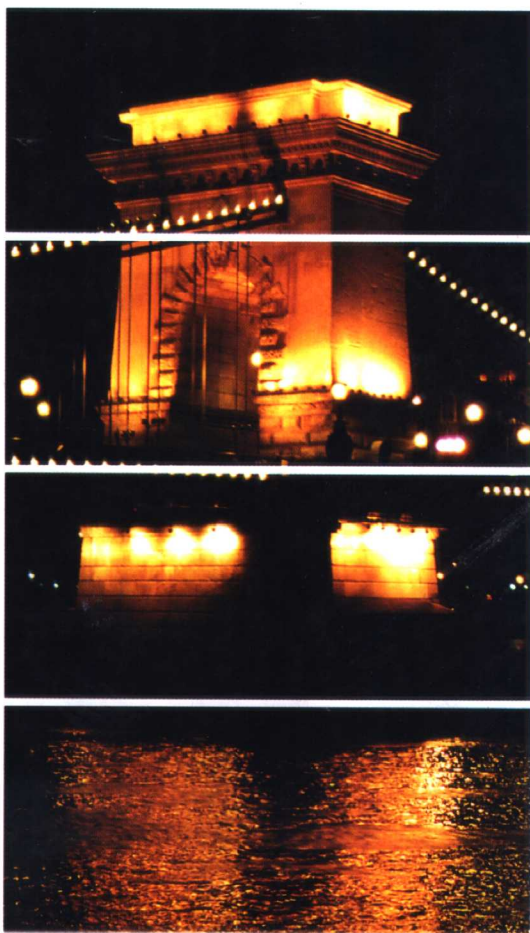
特别注重实用性。

书中包含大量精心设计的编程实例,

方便初学者使用。

★本书可作为高等院校程序设计语言课程教材,

也可供软件开发以及相关领域人员自学使用。



中国铁道出版社  
CHINA RAILWAY PUBLISHING HOUSE



谭浩强 主编

21 世纪高校计算机应用技术系列规划教材

# Java 语言程序设计

李尊朝 苏 军 编著

中国铁道出版社

2004 • 北京

MAJ109/10

## (京)新登字 063 号

### 内 容 简 介

本书通过对 Java 语言的全面介绍,使读者学会运用面向对象方法分析和解决实际问题的能力。在介绍 Java 编程原理和基础知识的前提下,特别注重实用性。书中包含大量精心设计并调试通过的编程实例,方便初学者使用。本书共分 16 章,内容包括 Java 概述、Java 基础、流程控制、方法、数组、面向对象的程序设计基础、类的封装、多态和继承、接口和包、异常处理、输入与输出、图形用户界面 GUI 设计、高级用户界面 GUI 设计、Applet 程序、多线程、数据库编程和网络编程。

本书集作者多年的教学经验编写而成,语言通俗易懂,内容安排合理,深入浅出。本书可作为高等学校各专业的程序设计语言课程教材,也可供从事软件开发以及相关领域的工程技术人员自学使用。

### 图书在版编目(CIP)数据

Java 语言程序设计/李尊朝 苏军编著. —北京:中国铁道出版社, 2004. 1

(21 世纪高校计算机应用技术系列规划教材)

ISBN 7-113-05714-4

I. J… II. ①李… ②苏… III. JAVA 语言-程序设计-高等学校-教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 004605 号

书 名: Java 语言程序设计

作 者: 李尊朝 苏 军

出版发行: 中国铁道出版社(100054, 北京市宣武区右安门西街 8 号)

策划编辑: 严晓舟 魏 春 秦绪好

责任编辑: 苏 茜 黄园园

封面设计: 孙天昭

印 刷: 北京鑫正大印刷有限公司

开 本: 787×1092 1/16 印张: 18.5 字数: 440 千

版 本: 2004 年 2 月第 1 版 2004 年 2 月第 1 次印刷

印 数: 1~5000 册

书 号: ISBN 7-113-05714-4/TP·1116

定 价: 24.00 元

版权所有 侵权必究

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

# 21 世纪高校计算机应用技术系列规划教材

## 编委会名单

主 任： 谭浩强

副主任： 陈维兴 严晓舟

委 员： （以下排名按姓氏字母的先后顺序为序）

|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| 安淑芝 | 安志远 | 侯冬梅 | 李雁翎 | 秦建中 |
| 秦绪好 | 宋 红 | 宋金珂 | 孙中胜 | 魏 春 |
| 魏善沛 | 熊伟建 | 薛淑斌 | 赵乃真 | 訾秀玲 |

# 丛书序言

21 世纪是信息技术高度发展并且得到广泛应用的时代,信息技术深刻地改变着人类的生活、工作和思维方式。每一个人都应当学习信息技术、应用信息技术。人们平常习惯说的计算机教育其内涵实际上已经发展为信息技术教育,内容主要包括计算机和网络的基本知识和应用。

对大多数人来说,学习计算机的目的是为了利用计算机这个现代化工具去处理工作和面临的各种问题,使自己能够跟上时代前进的步伐,同时要在学习的过程中努力培养自己的信息素养,使自己具有信息时代所要求的科学素质,站在信息技术发展和应用的前列,推动我国信息技术的发展。

学习计算机课程,有两种不同的方法,一是从理论入手;一是从实际应用入手。不同的人有不同的学习内容和学习方法。大学生中的多数人将来是各行各业中的计算机应用人才。对他们来说,不仅需要解决知道什么,更重要的是会做什么。因此要以应用为目的,注重培养应用能力,大力加强实践环节,激励创新意识。

根据实际教学的需要,我们组织编写这套“21 世纪高校计算机应用技术系列规划教材”。顾名思义,这套丛书的特点是突出应用技术,面向实际应用。在选材上,根据实际应用的需要决定内容的取舍,坚决舍弃那些现在用不到、将来也用不到的内容。在叙述方法上,采取“提出问题——介绍解决问题的方法——归纳结论和概念”的三部曲,这种从实际到理论、从具体到抽象、从个别到一般的方法,符合人们的认识规律,实践证明已取得了很好的效果。

本丛书采取模块化的结构,根据需要确定一批书目,也就是提供一个课程菜单供各校选用,以后根据信息技术的发展和教学的需要,不断地补充和调整。只要教学有需要,我们就组织编写新的教材,不受任何框框的限制。我们的指导思想是面向实际,面向应用,面向对象。这样比较灵活,能满足不同学校、不同专业的需要。希望各校的老师把你们的要求反映给我们,我们将会尽最大努力满足大家的要求。

本丛书可以作为大学计算机应用技术课程教材以及高职高专、成人高校和面向社会的培训班的教材,也可作为学习计算机的自学教材。

参加本丛书策划和编写工作的专家和老师们有:谭浩强、陈维兴、严晓舟、薛淑斌、秦建中、安淑芝、安志远、赵乃真、吕凤翥、李雁翎、宋红、周永恒、熊伟建、宋金珂、陈元春、冯继生、姚怡、沈洪、沈添、李尊朝、王晓敏、侯冬梅、訾秀玲、魏善沛、孙中胜、王丙义、程爱民、史秀璋、李振银、刘涛、李宁等。此外参加本丛书编辑和其他工作的还有:魏春、秦绪好、张雁芳、戴薇、郭晓溪、马建、姜淑静、杨东晓、于静、穆蓉等。对于他们的智慧、奉献和劳动表示深切的谢意。中国铁道出版社以很高的热情和效率组织了丛书的出版工作。在组织编写出版的过程中,得到全国高等院校计算机基础教育研究会和各高等院校老师的热情鼓励和支持,对此谨表衷心的感谢。

本丛书如有不足之处,请各位专家、老师和广大读者不吝指正。

谭浩强谨识

2003 年 2 月于清华园

# 前 言

面向对象技术是程序设计方法学的一场革命,已成为计算机应用开发领域的主流技术。面向对象技术能够更逼真地模拟现实世界,使软件开发人员按照认识客观事物的思维方法进行软件开发,它所倡导的将数据与操作捆绑的程序风格,正好符合现代大规模软件开发对高可靠性、易维护性和代码重用性的要求和发展潮流。随着网络技术的迅速发展和应用普及,对程序设计语言提出了安全性高、移植性好、操作平台无关等更高要求,Java 语言正是为了满足这些要求和潮流而设计的,所以从 1995 年诞生到现在,在短短的 8 年中就已经席卷全球,扩展到各个应用领域,随着网络科技以令人瞠目结舌的速度向前发展,成为现今最流行的程序设计语言。Java 语言的面向对象、平台无关、安全机制、高可靠性、多线程和内嵌的网络支持,使其成为 21 世纪开发应用程序的首选工具。有人预言,在不久的将来,全球 90% 以上的程序代码将是采用 Java 语言开发。

本书定位于大专院校的学生和从事软件开发以及相关领域的工程技术人员,培养读者利用面向对象的技术分析和解决问题的能力,指导读者在较短的时间内学会利用最先进的 Java 工具开发软件产品,顺应网络时代对人才的新需求。作者根据多年教学和软件开发经验,对本书的内容取舍、组织编排和经典实例都进行了精心设计。本书在难易程度上遵循由浅入深、循序渐进的原则。在写作风格上突出其实用性,将复杂的面向对象理论融于具体的实例之中。书中的大量实例都是经过精心设计,程序代码都做了认真调试,可以直接运行,方便初学者上机操作。

全书由 16 章组成。第 1 章介绍 Java 语言的特点、运行环境和开发步骤。第 2 章介绍 Java 语言的关键字、数据类型、运算符和表达式。第 3 章介绍选择和循环结构程序设计。第 4 章介绍方法的声明及其调用。第 5 章介绍程序开发中常用的数据结构:数组和字符串及其编程实例。第 6、7 和 8 章介绍面向对象技术中的对象、类、封装、继承、多态、接口及包等重要内容。第 9 章介绍 Java 强大的异常处理功能。第 10 章介绍 Java 的输入、输出及文件操作功能。第 11 和 12 章介绍图形用户界面的设计技术,内容包括图形界面开发基础、最新的 Swing 包技术、界面布局管理、键盘和鼠标事件处理、菜单及对话框程序设计等。第 13 章介绍 Applet 编程技术,包括 Applet 基础和编程实例。第 14 章介绍 Java 特有的多线程技术,包括多线程的基本机制、线程的实现方法、等待和同步调度技术等。第 15 章介绍数据库编程技术,内容包括数据库的基本概念、关系数据库的标准语言:结构化查询语言 SQL、Java 数据库连接技术 JDBC、数据库编程的基本技术。第 16 章介绍 Java 强大的网络编程技术。

要掌握面向对象的开发技术,利用 Java 语言开发出高质量的程序,首先要学习 Java 的语法规则,其次要学习 Java 类库,能否熟练地掌握类库的使用方法,在一定程度上决定了程序员的编程能力。计算机学科是实践性很强的学科,优秀的软件开发人员需要经过大量上机

实践的磨练。因此，希望读者在学习本书的过程中，尽可能将书中的大量实例及每章习题上机练习。

本书第1章、第4~5章、第9~10章、第13~15章由李尊朝编写，第2~3章、第6~8章、第11~12章、第16章由苏军编写。李尊朝设计了全书结构，并做了全书的统稿工作。本书在编写的过程中，参阅了大量书籍和网站等参考资料，得到了西安交通大学同仁及中国铁道出版社秦绪好编辑的支持和帮助，在此表示感谢。

尽管书稿几经修改，但由于作者学识有限，书中错误和不妥难免，望读者不吝赐教。

作 者  
2004年1月

# 目 录

|                              |    |
|------------------------------|----|
| 第 1 章 Java 概述.....           | 1  |
| 1-1 程序设计语言.....              | 2  |
| 1-1-1 机器语言 .....             | 2  |
| 1-1-2 汇编语言 .....             | 2  |
| 1-1-3 高级语言 .....             | 2  |
| 1-2 面向对象的程序设计语言——Java.....   | 3  |
| 1-2-1 Java 的发展历史 .....       | 3  |
| 1-2-2 Java 的特点 .....         | 4  |
| 1-2-3 Java 与 C/C++ 的比较 ..... | 6  |
| 1-3 Java 的开发和运行环境.....       | 6  |
| 1-3-1 J2SDK 的安装 .....        | 7  |
| 1-3-2 J2SDK 的设置 .....        | 7  |
| 1-4 开发和运行 Java 程序的步骤.....    | 7  |
| 1-4-1 选择编辑工具 .....           | 7  |
| 1-4-2 编译和运行 Java 程序 .....    | 8  |
| 1-5 UltraEdit.....           | 9  |
| 1-5-1 下载和安装 .....            | 9  |
| 1-5-2 软件的使用 .....            | 10 |
| 1-5-3 UltraEdit 的功能简介 .....  | 10 |
| 习题 .....                     | 12 |
| 第 2 章 Java 基础.....           | 13 |
| 2-1 Java 符号.....             | 14 |
| 2-1-1 标识符和关键字 .....          | 14 |
| 2-1-2 运算符和分隔符 .....          | 14 |
| 2-1-3 注释 .....               | 15 |
| 2-2 数据类型与变量、常量.....          | 15 |
| 2-2-1 数据类型 .....             | 15 |
| 2-2-2 常量 .....               | 16 |
| 2-2-3 变量 .....               | 18 |
| 2-3 语句、表达式和运算符.....          | 21 |
| 2-3-1 语句 .....               | 21 |
| 2-3-2 表达式 .....              | 21 |
| 2-3-3 算术运算符 .....            | 22 |
| 2-3-4 关系运算符 .....            | 24 |





|              |                        |           |
|--------------|------------------------|-----------|
| 2-3-5        | 逻辑运算符 .....            | 25        |
| 2-3-6        | 位运算符 .....             | 26        |
| 2-3-7        | 赋值运算符 .....            | 26        |
| 2-4          | 数据类型的转换 .....          | 27        |
| 2-4-1        | 自动类型转换 .....           | 27        |
| 2-4-2        | 强制类型转换 .....           | 28        |
| 2-4-3        | 表达式的类型转换 .....         | 29        |
| 2-4-4        | 运算符的优先级 .....          | 29        |
| 习题           | .....                  | 30        |
| <b>第 3 章</b> | <b>Java 流程控制 .....</b> | <b>31</b> |
| 3-1          | 选择结构程序设计 .....         | 32        |
| 3-1-1        | if 语句 .....            | 32        |
| 3-1-2        | if...else 语句 .....     | 33        |
| 3-1-3        | if 语句的嵌套 .....         | 34        |
| 3-1-4        | switch 语句 .....        | 36        |
| 3-2          | 循环结构控制 .....           | 37        |
| 3-2-1        | while 语句 .....         | 37        |
| 3-2-2        | do...while 语句 .....    | 39        |
| 3-2-3        | for 语句 .....           | 40        |
| 3-2-4        | 循环嵌套 .....             | 42        |
| 3-3          | 跳转语句 .....             | 44        |
| 3-3-1        | break 语句 .....         | 44        |
| 3-3-2        | continue 语句 .....      | 46        |
| 习题           | .....                  | 47        |
| <b>第 4 章</b> | <b>方法 .....</b>        | <b>49</b> |
| 4-1          | 方法声明 .....             | 50        |
| 4-2          | 方法调用 .....             | 51        |
| 4-3          | 参数传递 .....             | 53        |
| 4-4          | 递归 .....               | 54        |
| 习题           | .....                  | 57        |
| <b>第 5 章</b> | <b>数组 .....</b>        | <b>59</b> |
| 5-1          | 一维数组 .....             | 60        |
| 5-1-1        | 一维数组的声明 .....          | 60        |
| 5-1-2        | 一维数组的初始化 .....         | 60        |
| 5-2          | 多维数组 .....             | 62        |
| 5-2-1        | 二维数组的声明 .....          | 62        |
| 5-2-2        | 二维数组的初始化 .....         | 63        |

|              |                          |           |
|--------------|--------------------------|-----------|
| 5-3          | 数组的基本操作 .....            | 65        |
| 5-3-1        | 数组的引用 .....              | 65        |
| 5-3-2        | 数组的复制 .....              | 65        |
| 5-3-3        | 数组的输出 .....              | 66        |
| 5-4          | 数组应用举例 .....             | 68        |
| 5-5          | 数组参数 .....               | 72        |
| 5-6          | 字符串 .....                | 75        |
| 5-6-1        | 字符数组与字符串 .....           | 75        |
| 5-6-2        | 字符串 .....                | 76        |
| 5-6-3        | 字符串操作 .....              | 77        |
| 5-6-4        | 字符串数组 .....              | 79        |
| 习题           | .....                    | 80        |
| <b>第 6 章</b> | <b>面向对象的程序设计概念 .....</b> | <b>81</b> |
| 6-1          | 概念 .....                 | 82        |
| 6-2          | 类的概念 .....               | 83        |
| 6-3          | 类的定义 .....               | 83        |
| 6-4          | 对象 .....                 | 84        |
| 6-4-1        | 创建对象 .....               | 84        |
| 6-4-2        | 对象的引用 .....              | 85        |
| 6-4-3        | 对象的初始化和构造方法 .....        | 86        |
| 6-5          | 类的成员变量 .....             | 89        |
| 6-6          | 方法的引用和修饰 .....           | 91        |
| 习题           | .....                    | 96        |
| <b>第 7 章</b> | <b>类的封装、多态和继承 .....</b>  | <b>97</b> |
| 7-1          | 类的封装 .....               | 98        |
| 7-1-1        | 对象的可见性和作用域 .....         | 98        |
| 7-1-2        | public .....             | 100       |
| 7-1-3        | private .....            | 100       |
| 7-1-4        | protected .....          | 102       |
| 7-1-5        | friendly .....           | 103       |
| 7-2          | 多态机制 .....               | 103       |
| 7-2-1        | 方法的重载 .....              | 104       |
| 7-2-2        | 构造方法的重载 .....            | 105       |
| 7-2-3        | 方法的覆盖 .....              | 105       |
| 7-3          | 类的继承 .....               | 105       |
| 7-3-1        | 继承的概念 .....              | 105       |
| 7-3-2        | 继承的实现 .....              | 106       |
| 习题           | .....                    | 114       |



|                                |            |
|--------------------------------|------------|
| <b>第 8 章 接口和包</b>              | <b>115</b> |
| 8-1 抽象类和方法                     | 116        |
| 8-1-1 定义抽象类                    | 116        |
| 8-1-2 抽象类的实现                   | 116        |
| 8-2 接口                         | 121        |
| 8-2-1 概念                       | 121        |
| 8-2-2 定义接口                     | 121        |
| 8-2-3 接口的特点和实现                 | 122        |
| 8-3 包                          | 127        |
| 8-3-1 概念                       | 127        |
| 8-3-2 包的定义                     | 127        |
| 8-3-3 存放位置                     | 128        |
| 8-3-4 包的引用                     | 128        |
| 8-3-5 实例：将多个相互独立的类放入同一个包中      | 129        |
| 习题                             | 130        |
| <b>第 9 章 异常处理</b>              | <b>131</b> |
| 9-1 Java 异常处理机制                | 132        |
| 9-2 异常处理方法                     | 133        |
| 9-2-1 try...catch...finally 结构 | 133        |
| 9-2-2 抛出异常                     | 136        |
| 9-2-3 自定义异常类                   | 138        |
| 习题                             | 141        |
| <b>第 10 章 输入与输出</b>            | <b>143</b> |
| 10-1 输入输出类库                    | 144        |
| 10-1-1 流                       | 144        |
| 10-1-2 输入输出流类                  | 144        |
| 10-2 标准输入输出                    | 149        |
| 10-2-1 标准输入                    | 149        |
| 10-2-2 标准输出                    | 152        |
| 10-2-3 标准错误                    | 153        |
| 10-3 文件操作                      | 153        |
| 10-3-1 建立 File 对象              | 153        |
| 10-3-2 File 对象的属性和操作           | 154        |
| 习题                             | 163        |
| <b>第 11 章 图形用户界面</b>           | <b>165</b> |
| 11-1 Java 的图形用户界面 AWT 包        | 166        |
| 11-1-1 屏幕坐标系                   | 166        |

|               |                           |            |
|---------------|---------------------------|------------|
| 11-1-2        | 建立窗口的 Frame 类.....        | 167        |
| 11-1-3        | 使用图形用户界面 AWT 组件.....      | 169        |
| 11-1-4        | AWT 图形界面的深入学习.....        | 172        |
| 11-2          | Java 语言中的事件处理.....        | 176        |
| 11-2-1        | 概念.....                   | 177        |
| 11-2-2        | 可用的事件监听者和它们处理的事件种类.....   | 177        |
| 11-2-3        | 事件及其响应.....               | 178        |
| 11-2-4        | 在 Java 中的事件处理方式.....      | 180        |
| 11-3          | Swing 包.....              | 183        |
| 11-3-1        | 创建 JFrame 窗口.....         | 183        |
| 11-3-2        | Swing 包中常用组件.....         | 186        |
|               | 习题.....                   | 200        |
| <b>第 12 章</b> | <b>高级用户界面 GUI 设计.....</b> | <b>201</b> |
| 12-1          | 布局管理器.....                | 202        |
| 12-1-1        | 概念.....                   | 202        |
| 12-1-2        | BorderLayout.....         | 202        |
| 12-1-3        | FlowLayout.....           | 203        |
| 12-1-4        | CardLayout.....           | 204        |
| 12-1-5        | GridLayout.....           | 205        |
| 12-1-6        | BoxLayout.....            | 207        |
| 12-2          | 键盘和鼠标事件处理.....            | 208        |
| 12-2-1        | 键盘事件.....                 | 208        |
| 12-2-2        | 鼠标事件.....                 | 210        |
| 12-3          | 菜单设计.....                 | 212        |
| 12-4          | 对话框设计.....                | 216        |
| 12-4-1        | JOptionPane 对话框.....      | 217        |
| 12-4-2        | Dialog.....               | 223        |
| 12-5          | 窗口.....                   | 225        |
| 12-5-1        | JPanel 容器.....            | 225        |
| 12-5-2        | JScrollPane 容器.....       | 227        |
|               | 习题.....                   | 230        |
| <b>第 13 章</b> | <b>Applet 程序.....</b>     | <b>231</b> |
| 13-1          | Applet 简介.....            | 232        |
| 13-1-1        | Applet 类.....             | 232        |
| 13-1-2        | Applet 程序的运行过程.....       | 232        |
| 13-1-3        | Applet 程序的建立和运行.....      | 232        |
| 13-2          | Applet 程序举例.....          | 233        |
|               | 习题.....                   | 238        |



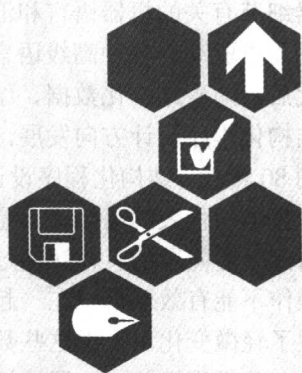
|                                      |     |
|--------------------------------------|-----|
| <b>第 14 章 多线程</b> .....              | 239 |
| 14-1 Java 的多线程机制.....                | 240 |
| 14-1-1 线程的生命周期.....                  | 240 |
| 14-1-2 多线程的实现方法.....                 | 240 |
| 14-2 通过 Thread 类实现多线程.....           | 241 |
| 14-3 通过 Runnable 接口实现多线程.....        | 242 |
| 14-4 线程等待.....                       | 243 |
| 14-5 线程同步.....                       | 246 |
| 习题.....                              | 250 |
| <b>第 15 章 数据库编程</b> .....            | 251 |
| 15-1 数据库简介.....                      | 252 |
| 15-1-1 关系型数据库.....                   | 252 |
| 15-1-2 SQL 简介.....                   | 253 |
| 15-2 使用 JDBC 连接数据库.....              | 254 |
| 15-2-1 JDBC 简介.....                  | 254 |
| 15-2-2 JDBC 驱动程序.....                | 255 |
| 15-3 建立数据源.....                      | 255 |
| 15-3-1 建立数据库.....                    | 256 |
| 15-3-2 建立数据源.....                    | 258 |
| 15-4 Java 数据库编程.....                 | 259 |
| 15-4-1 数据库编程的一般过程.....               | 259 |
| 15-4-2 数据库编程实例.....                  | 261 |
| 习题.....                              | 267 |
| <b>第 16 章 网络编程</b> .....             | 269 |
| 16-1 网络基础.....                       | 270 |
| 16-2 URL.....                        | 271 |
| 16-2-1 URL 类.....                    | 271 |
| 16-2-2 URLConnection 类.....          | 273 |
| 16-3 Socket 通信.....                  | 275 |
| 16-3-1 Socket 概念.....                | 275 |
| 16-3-2 Socket 的通信机制.....             | 275 |
| 16-3-3 Socket 类与 ServerSocket 类..... | 275 |
| 16-3-4 URL 与 Socket 通信的区别.....       | 281 |
| 16-3-5 UDP 通信.....                   | 282 |
| 习题.....                              | 282 |

# 1

## Java 概述

### 本章导读

Java 语言是由 Sun 公司于 1995 年 5 月 23 日正式推出的纯面向对象的程序设计语言，集安全性、简单性、易用性和平台无关性于一身，并且特适合于网络环境下编程使用。





## 1-1 程序设计语言

电子计算机是一种机器，要借助计算机完成人类的某种思维活动，必须将其用计算机能理解和执行的形式语言来描述，这种语言就称为程序设计语言。程序设计语言和人类使用的自然语言之间存在着较大的差距，自从 1946 年第一台电子计算机问世以来，程序设计语言的发展变迁就是为了缩小这一差距。

### 1-1-1 机器语言

计算机由电子器件组成，电子器件最容易表达的是电位的高、低或电流的通、断两种稳定状态，可以用 0 和 1 两种符号来表示这两种状态，这便是最早的程序设计语言——机器语言的基本单位。用机器语言编写的程序由 0 和 1 组成，计算机能理解并直接执行。然而由 0 和 1 组成的“0、1 串”没有丝毫的形象意义，人们难以记忆和理解，所以用机器语言编写的程序效率最低，并且容易出错。

### 1-1-2 汇编语言

为了克服机器语言抽象、难以理解和记忆的缺点，人们用便于理解和记忆的符号来代替 0、1 串，这便是汇编语言。汇编语言使用助记词编写程序，较之机器语言更接近自然语言。但汇编语言涉及大量的机器细节，是与具体机器硬件有关的语言，是一种面向机器的语言。只要更换或升级机器硬件，就得重新编写程序。

### 1-1-3 高级语言

虽然汇编语言较之机器语言便于理解和记忆，但却像机器语言一样，与具体的机器指令系统有关，离不开计算机的硬件特性。用它们编写程序复杂度高，效率低下，可维护性和可移植性差。为了从根本上摆脱语言对机器的依赖，经过多年的潜心研究，与具体机器指令系统无关、表达方式接近自然语言的新一代语言问世了。新一代语言采用具有一定涵义的数据命名和人们易于理解的执行语句，且屏蔽了机器细节，将这种语言称为高级语言，而将与具体机器细节有关的机器语言和汇编语言称为低级语言。高级语言一经推出立刻受到广泛欢迎，受市场需求驱动，各种高级语言相继问世，极大地推动了计算机应用及计算机技术的发展。

此后，随着结构化数据、结构化语句、数据抽象和过程抽象等概念的提出，高级语言逐步向结构化程序设计方向发展，进一步缩小了计算机语言和自然语言的距离。在 20 世纪 70 年代到 80 年代，结构化程序设计语言非常流行，成为当时软件开发的主流技术。以结构化程序设计技术为代表的高级语言是一种面向过程的语言。面向过程的语言可以用计算机能理解的逻辑，表达问题的具体解决过程，然而它将数据和过程分离为独立的实体，使程序中的数据和操作不能有效地组织在一起，很难把具有多种相互联系的复杂事物表述清楚。当数据结构发生了轻微变化，处理这些数据的算法也要做相应的修改。因而用这种程序设计方法编写的软件，其重用性差。为了较好地解决软件的重用性问题，使数据与程序始终保持相容，人

们提出了面向对象的程序设计 OOP (Object Oriented Programming) 方法。

面向对象的程序设计语言能更好地描述客观事物及其相互联系, 追求对现实世界的直接模拟, 具体体现在:

- (1) 客观世界是由具体的事物构成, 每个事物都具有自己的一组静态特征(属性)和一组动态特征(行为)。在面向对象的程序设计语言中, 将客观事物抽象为对象(Object), 用一组数据描述对象的静态特征(属性), 用一组方法刻画对象的动态特征。
- (2) 客观世界中的事物既具有特殊性又具有共同性。人类认识客观世界的基本方法之一就是事物进行分类, 根据事物的共同性将事物分为某些类。面向对象的程序设计语言用类来表示一组具有相同属性和方法的对象。
- (3) 在同一类事物中, 除具有共性外, 每个事物又具有自己的特殊性。面向对象语言用父类与子类的概念来描述。父类中描述事物的共性, 在子类中描述个性。
- (4) 客观世界中的事物是一个独立的整体, 它内部的具体细节外部常常并不关心。面向对象的语言通过封装机制把对象的属性和方法结合为一个整体, 并且屏蔽了对象的内部细节。
- (5) 客观世界中的一个事物可能与其他事物之间存在某种行为上的联系。面向对象语言通过消息连接来表示对象之间的这种动态联系。

综上所述, 面向对象的语言使程序能够比较直接的反映客观世界的本来面目, 并且使软件开发人员能够运用人类认识事物所采用的一般思维方法来进行软件开发。

和其他计算机语言比较, 面向对象的程序设计语言和人类使用的自然语言之间的差距是最小的, 是当今软件开发和应用的主流技术。

## 1-2 面向对象的程序设计语言——Java

目前应用广泛的 Internet 是将世界上成千上万的计算机子网连接成一个超网, 而这些子网是由世界各地各种不同型号、不同规模, 使用不同操作系统, 具有不同应用平台的计算机组成。为了发挥 Internet 的巨大作用, 需要一种能运行在各种计算机上、具有平台无关性和高移植性的语言。Java 语言以其面向对象、平台无关、多线程、安全可靠等特性成为 Internet 时代程序设计语言中的佼佼者。

### 1-2-1 Java 的发展历史

Java 的发展历史, 可以追溯到 1990 年。当时 Sun Microsystem 公司为了发展消费类电子产品而进行了一个名为 Green 的项目计划。这个计划的负责人是 James Gosling。起初他用具有面向对象特征的 C++ 语言编写嵌入式软件, 可以放在面包机或 PDA (Personal Digital Assistant, 个人数字助理) 等小型电子消费设备里, 以使设备变得更为“聪明”, 更具备人工智能。但后来发现 C++ 并不适合这类任务, 因为 C++ 常会使系统失效。尤其在内存管理方面, C++ 采用直接地址访问方式, 需要程序员记录并管理内存资源。这造成程序员编程的极大负担, 并可能产生多个 Bugs。面包机上的程序错误可能使面包机烧坏甚至爆炸。





为了解决此类问题，Gosling 决定开发一种新的语言，并取名为 Oak。它采用了大部分与 C++ 类似的语法，对可能具备危险性的功能加以改进，例如将内存管理改为由语言自己进行管理，以减少程序员的负担及可能发生的错误。Oak 是一种可移植的语言，也是一种平台独立的语言，能够在各种芯片上执行，可以降低设备的研发成本。

到了 1994 年，Oak 的技术已日趋成熟，这时 Internet 也开始蓬勃发展。Oak 研发小组发现 Oak 很适合作为一种 Internet 上的程序设计语言，因此开发了一个能与 Oak 配合使用的浏览器——Hotjava。实践证明，Oak 的确能用于 Internet 上的程序开发。鉴于 Oak 已经被其他产品注册使用，研发小组就以常饮用的咖啡 Java 重新命名该产品。此后，Java 就随着 Internet 的发展而快速发展起来。

## 1-2-2 Java 的特点

Java 是一种简单、安全、容易使用、面向对象、可移植、高性能、多线程的语言。

### 1. 简单性

Java 语言简单高效，基本 Java 系统（编译器和解释器）所占空间不足 250KB。由于 Java 最初是为了对家用电器进行集成控制而设计的，因而具备简单明了的特征。

如上所述，Java 从 C++ 演变而来，保留了 C++ 的许多优点，去除了 C++ 中易产生错误的功能，简化了内存管理，减轻了程序员进行内存管理的负担。

### 2. 面向对象

面向对象技术是现代软件工业的一次革新，提高了软件的模块化程度和重复使用率，缩短了软件开发时间，减低了开发成本。在 Java 之前虽然已经有面向对象的程序设计语言问世，但有些如 C++ 并不是完全的面向对象，而是面向过程和面向对象的混合体。Java 则是完全面向对象的程序设计语言。

### 3. 安全性

Java 是可以用在网络及分布环境下的网络程序设计语言。在网络环境下，语言的安全性变得更为重要。Java 提供了许多安全机制来保证其使用上的安全性。

### 4. 平台独立

平台独立指程序不受操作平台的限制，可以应用在各种平台上。Java 源程序经过编译后生成字节码文件，而字节码与具体的计算机无关。只要计算机安装了能解释执行字节码的 Java 虚拟机 JVM (Java Virtual Machine)，就可以执行字节码文件，从而实现 Java 的平台独立性。

为了理解 Java 语言的平台独立性，在此对 Java 虚拟机做一介绍。大部分高级语言的源程序必须经过编译或解释程序翻译成机器语言，才能在计算机上执行。例如 C、C++ 等属于编译式语言，而 Basic 和 Lisp 属于解释型语言。

然而，Java 程序却比较特殊，它必须先经过编译，再经过解释过程才能执行。通过编译器，Java 语言源程序转换成与平台无关的中间编码，Java 称之为字节码 (Byte Codes)。字节码再经过解释器的解释，转换为机器码，便可在计算机上运行。