

## <<Java语言程序设计>>

### 图书基本信息

书名：<<Java语言程序设计>>

13位ISBN编号：9787534128240

10位ISBN编号：7534128242

出版时间：2006-1

出版时间：浙江科学技术

作者：胡维华主编

页数：258

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;Java语言程序设计&gt;&gt;

## 前言

Java语言具有面向对象、跨平台、安全性、多线程以及强大的网络支持等显著特点,这使得Java语言成为许多应用系统的理想开发工具。

随着Internet的产生和发展,Java语言的这些优点引起软件开发人员极大的关注,Java语言已经成为最流行的网络编程语言之一。

在欧美国家,许多大学将Java语言作为第一门程序设计课程。

在国内,越来越多的高校选择Java语言作为程序设计课程的入门语言。

本书主要是为高等学校学生学习“Java语言程序设计”课程而编写的教材,教学对象是Java语言的初学者。

在学习本书之前,并不要求具备面向对象的基础知识,也不要求学习或接触过C或其他高级程序设计语言。

本书主要讲述Java程序设计的基础知识和基本方法。

在编写过程中,结合作者多年Java程序设计教学中的经验与体会,针对程序设计语言初学者的特点,力求体现以下特色:适当加大了流程控制语句等Java语言基础知识方面的讲述,在教学安排上注重编程能力的培养,着力训练好程序设计的基本功。

在全面讲述了Java语言基本语法和面向对象程序设计基本概念的基础上,更强调如何利用Java语言解决实际应用问题的能力。

本书的内容安排及阐述遵循由浅入深、循序渐进的原则,书中的实例程序力求简洁明了,能在最基本的Java运行环境下执行。

每章后面精心设计了练习题,读者可以通过做这些习题,巩固并灵活运用所学知识。

本书还结合新技术的发展,通过一组实例介绍网络、多媒体、JDBC数据库等编程技术,使读者在掌握Java基本概念和编程方法的同时,能全面了解Java的特点,掌握Java最新实用技术。

本书编写得到了浙江省高校计算机教学研究会和浙江科学技术出版社的大力支持。

全书由胡维华教授和韩建平副教授担任主编并设计全书结构。

全书共分12章,其中瞿有甜老师编写第1章和第4章,田贤忠老师编写第2章和第3章,韩建平老师编写第5章和第6章,杨蓉老师编写第7章和第8章,杨柏林老师编写第9章和第10章,罗显贵老师编写第11章和第12章。

全书由胡维华和韩建平统稿,徐小良担任主审。

本书可以作为各类大专院校、各类培训与等级考试的教学用书,也可作为对Java语言程序设计感兴趣者的自学用书。

本书在编写过程中,参阅了大量国内外书籍和网站参考资料,在此表示衷心的感谢。

由于作者水平有限,书中难免还存在缺点和错误,期待广大读者批评和指正。

## <<Java语言程序设计>>

### 内容概要

《Java语言程序设计》著作权属于浙江省计算机等级考试办公室，我社独家享有《Java语言程序设计》的图书出版发行专有使用权。

任何单位，组织，个人未经我社和浙江省高校计算机等级考试办公室授权同意，不得擅自翻印，摘录，汇编《Java语言程序设计》书中有关内容或根据《Java语言程序设计》编制相关辅导读物。

未经授权而有上述行为者，是对我社和浙江省高校计算机等级考试办公室合法权益的侵害，我社将通过法律手段追究其法律责任。

## &lt;&lt;Java语言程序设计&gt;&gt;

## 书籍目录

第1章 Java概述1.1 Java语言简介1.1.1 Java的来历1.1.2 Java语言的现状1.1.3 Java语言的特点1.1.4 Java程序的工作原理1.2 Java语言开发环境1.2.1 Java平台的三个版本1.2.2 Java程序执行开发环境的建立1.2.3 Java集成环境开发工具简介1.2.4 Java在线帮助1.3 初识Java程序1.3.1 JavaApplication1.3.2 JavaApplet1.4 Java程序的基本输入与输出1.4.1 JavaApplication字符界面的输入与输出1.4.2 JavaApplet图形界面输入与输出1.4.3 JavaApplication图形界面输入与输出习题第2章 基本数据类型与表达式2.1 基本数据类型2.1.1 数据类型分类2.1.2 基本数据类型2.1.3 数据类型的转换2.2 常量与变量2.2.1 标识符与关键字2.2.2 常量2.2.3 变量2.3 表达式2.3.1 赋值表达式2.3.2 算术表达式2.3.3 关系表达式2.3.4 逻辑表达式2.3.5 条件表达式2.3.6 位运算表达式习题第3章 流程控制语句3.1 Java语句概述3.1.1 Java语句3.1.2 程序的三种基本结构3.2 选择语句3.2.1 if语句3.2.2 switch语句3.3 循环语句3.3.1 while语句3.3.2 do.while语句3.3.3 for语句3.3.4 break和continue语句3.4 循环的嵌套习题第4章 数组和字符串操作4.1 一维数组4.1.1 一维数组的声明4.1.2 一维数组的创建与初始化4.1.3 一维数组的访问4.2 二维数组4.2.1 二维数组的声明4.2.2 二维数组的创建与初始化4.2.3 二维数组的访问4.3 数组应用举例4.4 字符串的操作4.4.1 字符串概述4.4.2 String字符串说明及初始化4.4.3 String字符串的基本操作4.4.4 字符串的查找4.4.5 字符串比较4.4.6 字符串缓冲区类StringBuffer习题第5章 类与对象5.1 类的定义5.1.1 类与对象的基本概念5.1.2 类定义的基本框架5.1.3 成员变量定义5.1.4 方法的定义5.2 对象的创建和使用5.2.1 声明与创建对象5.2.2 构造方法与对象初始化5.2.3 访问对象5.3 方法的进一步讨论5.3.1 方法的调用与返回值5.3.2 方法的参数传递5.3.3 递归方法5.3.4 方法的重载5.4 静态变量与静态方法5.4.1 静态变量5.4.2 静态方法5.4.3 静态初始化器5.5 this关键字5.5.1 使用this解决局部变量与成员变量同名问题5.5.2 在构造方法中调用另一个重载的构造方法5.6 包5.6.1 包的含义5.6.2 创建包5.6.3 使用包中的类习题第6章 继承、封装与多态6.1 类的继承6.1.1 继承的基本概念6.1.2 子类的定义6.1.3 域的继承、隐藏与添加6.1.4 方法的继承、重载与覆盖6.1.5 super关键字的使用6.1.6 子类构造方法与父类构造方法6.1.7 父类对象与子类对象6.2 封装与访问控制6.2.1 封装6.2.2 类成员的访问控制符6.2.3 类的访问控制符6.3 抽象类与接口6.3.1 抽象类和抽象方法6.3.2 接口的声明与实现6.4 final关键字6.4.1 final变量6.4.2 final方法6.4.3 final类6.5 多态性6.5.1 覆盖实现多态性6.5.2 重载实现多态性6.6 内部类和匿名类6.6.1 内部类6.6.2 方法中的内部类6.6.3 匿名类6.7 Java类库与常用类6.7.1 Java类库及结构6.7.2 Object类6.7.3 Math类6.7.4 基本类型的包装类6.7.5 System类习题第7章 异常处理7.1 异常与异常处理类7.1.1 异常的基本概念7.1.2 异常处理类7.2 异常处理7.2.1 捕获与处理异常7.2.2 多异常的捕获与处理第8章 Java输入输出及文件处理第9章 图形用户界面编程第10章 Applet程序设计第11章 多线程第12章 Java网络、数据库和多媒体编程参考文献

## &lt;&lt;Java语言程序设计&gt;&gt;

## 章节摘录

7.解释性 Java解释器（运行系统）能直接运行目标代码指令。链接程序通常比编译程序所需资源少，所以程序员可以在创建源程序上花上更多的时间。因为解释执行的语言一般会比编译执行的语言（如C和C++语言）的执行效率低，所以，这其实也是Java的一个缺点。

8.高性能 如果解释器速度不慢，Java可以在运行时直接将目标代码翻译成机器指令。Sun用直接解释器一秒钟内可调用300 000个过程。翻译目标代码后运行的速度与C和C++的性能没什么区别。

9.多线程 Java提供的多线程功能使得在一个程序里可同时执行多个小任务。线程（有时也称小进程）——是一个大进程里分出来的小的独立的进程。

因为Java的多线程技术实现得较好，所以比C和C++更健壮。

多线程带来的更大的好处是更好的交互性能和实时控制性能。

当然实时控制性能还取决于系统本身（UNIX、Windows、Macintosh等）。

任何用过当前浏览器的人，都感觉为调一副图片而等待是一件很烦恼的事情。

在Java里，你可用一个单独的线程来调一副图片，而你可以访问HTML里的其他信息而不必等它。

多线程主要是用来处理复杂事务或需要并行的事务。

Java虚拟机本身就是一个多线程的程序。

采用多线程机制是提高程序运行效率的一种方法，当然也增加了程序设计的难度。

10.动态 Java的动态特性是其面向对象设计方法的发展。

它允许程序动态地装入运行过程中所需要的类，这是C++语言进行面向对象程序设计所无法实现的。

Java从如下几方面采取措施解决这个问题。

Java编译器不是将对实例变量和成员函数的引用编译为数值引用，而是将符号引用信息在字节码中保存下传递给解释器，再由解释器在完成动态连接类后，将符号引用信息转换为数值偏移量。

这样，一个在存储器生成的对象不在编译过程中决定，而是延迟到运行时由解释器确定。

这样，对类中的变量和方法进行更新时就不至于影响现存的代码。

解释执行字节码时，这种符号信息的查找和转换过程仅在一个新的名字出现时才进行一次，随后代码便可以全速执行。

在运行时确定引用的好处是可以使用已被更新的类，而不必担心会影响原有的代码。

如果程序连接了网络中另一系统中的某一类，该类的所有者也可以自由地对该类进行更新，而不会使任何引用该类的程序崩溃。

Java还简化了使用升级或者更新协议的方法。

如果你的系统运行Java程序时遇到了不知该怎样处理的程序，没关系，Java能自动下载你所需要的功能程序。

总而言之，Java语言是一种易学好用，健壮性高，但执行效率相对较低的计算机语言。

它适合于各种对执行时间要求不是很苛刻的应用程序。

用Java语言编写程序一般比其他计算机语言编写程序花费更少的时间，而且调试所需的时间也会较短。

对于计算机初学者或正打算开始学习一门计算机语言的工程师或教学科研工作者来说，选择Java程序设计是一个很好的方案。

## <<Java语言程序设计>>

### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>