

<<Java程序设计与实验>>

图书基本信息

书名：<<Java程序设计与实验>>

13位ISBN编号：9787811239058

10位ISBN编号：7811239051

出版时间：2010-1

出版单位：清华大学出版社有限公司

作者：万年红 编

页数：295

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>

<<Java程序设计与实验>>

前言

当今的社会是信息的社会，以网络技术为核心的信息技术的迅猛发展，为计算机网络编程提供了前所未有的发展机遇。

Java是当前最为流行的纯面向对象的网络编程语言之一，它所具有的面向对象性、简单性、分布性、健壮性、安全可靠、平台无关性、可移植性、解释性、多线程、动态性和高性能使得它特别适合开发网络应用程序。

目前，用Java开发的手机程序、网络游戏比比皆是，在移动电话、家用电器等领域中无一没有Java技术的存在。

Java完全具备了面向对象的抽象性、封装性、继承性、多态性等特征，被Bill Gates确定为有史以来最伟大的面向对象程序设计语言。

本书主要内容如下。

第1章是概述部分，主要介绍程序设计与程序设计语言，面向对象基本概念及与面向过程的区别，面向对象技术基本要素、特征，Java的发展历史及特点，Java类的实现、对象的产生及构造方法，Java程序运行环境的搭建安装、配置与使用，简单Application和Applet程序的编译与运行等内容。

第2章是Java程序设计基础部分，主要介绍Java程序结构及编写规范，注释、关键字、标识符，数据类型及其转换、变量和常量，运算符及其优先级、表达式、数组，程序的流程控制语句等内容。

第3章是方法与变量部分，主要介绍方法的声明与调用格式，参数传递方式，嵌套与递归，对象与变量的生命周期，变量初始化及赋值，static：方法和变量，main（）的语法等内容。

第4章介绍包和封装性。

主要包括包的概念及分类，用户自定义包的声明及建立途径，import语句的使用，类的封装、访问权限等内容。

第5章介绍继承性，主要包括继承性的概念，实现继承的语法格式，多级继承、单继承和多重继承，构造方法的调用次序，对象的类型转换等内容。

第6章介绍多态性，主要包括多态性概念，方法重载与方法覆盖，关键字this、super和final的使用情况。

第7章介绍抽象类、接口和内部类，主要包括抽象类和抽象方法，接口的概念，接口声明与使用，内部类和匿名内部类等内容。

<<Java程序设计与实验>>

内容概要

本书以J2SE平台为基础，主要介绍面向对象程序设计思想与方法，主要包括Java语言基本概述，Java类的实现、对象的产生，Java程序运行环境的搭建安装、配置与使用，Java程序的基本语法，方法与变量，Java包的定义与使用，抽象性、封装性、继承性、多态性，抽象类、抽象方法、接口及内部类的声明及使用，常用的Java API实用类，图形用户界面设计方法和步骤、常用控件、事件类、监听器、适配器及事件处理模型，异常与异常处理机制，多线程，Applet与WEB开发应用、Applet的多媒体程序开发，Java数据库开发，输入和输出流处理等内容。

本书非常适合作为高职高专计算机及相关专业学生的教材，同时也可作为软件开发人员的参考书。

<<Java程序设计与实验>>

书籍目录

第1章 概述 1.1 程序设计 1.2 程序设计语言及其类型 1.3 面向过程程序设计与面向对象程序设计 1.3.1 面向过程程序设计 1.3.2 面向对象程序设计 1.4 类的实现 1.4.1 Java类的定义 1.4.2 对象的产生与使用 1.4.3 构造方法 1.5 面向对象的基本特性 1.5.1 封装性 1.5.2 继承性 1.5.3 多态性 1.6 Java语言概述 1.6.1 Java语言发展历史 1.6.2 Java语言特点 1.6.3 Java语言与C、C++的区别 1.7 Java开发环境的搭建 1.7.1 Java技术的架构 1.7.2 Java程序与C程序的编译、执行过程比较 1.7.3 JDK安装与配置 1.7.4 Java类库 1.7.5 Java程序类型 1.8 Java集成开发环境 1.9 简单Java程序编写、编译与运行 1.9.1 Application程序编译与运行 1.9.2 Applet程序编译与运行 1.9.3 Application和Applet运行过程的比较 小结 习题第2章 Java程序设计基础 2.1 Java程序结构及编写规范 2.1.1 Java程序基本规范 2.1.2 关键字 2.1.3 标识符 2.2 数据类型、常量与变量 2.2.1 数据类型 2.2.2 常量 2.2.3 变量 2.3 数据类型转换 2.3.1 自动类型转换 2.3.2 强制类型转换 2.4 Java中的运算符与表达式 2.4.1 运算符 2.4.2 表达式及运算符优先级、结合性 2.4.3 语句和语句块 2.5 程序的流程控制 2.5.1 顺序结构 2.5.2 分支语句 2.5.3 循环语句 2.5.4 跳转语句 2.6 数组 2.6.1 数组概念 2.6.2 数组的声明与使用 2.6.3 数组的分类 2.6.4 一些与数组操作有关的方法 小结 习题第3章 方法与变量 3.1 方法 3.1.1 方法的声明 3.1.2 方法的调用 3.1.3 方法嵌套与递归 3.1.4 方法参数传递方式 3.2 变量的作用域和生命周期 3.2.1 变量的作用域和生命周期 3.2.2 对象的作用域和生命周期 3.3 变量初始化 3.3.1 成员变量初始化 3.3.2 局部变量初始化 3.4 static方法和变量.....第4章 包和封装性第5章 继承性第6章 多态性第7章 抽象类、接口和内部类第8章 Java API实用类第9章 图形用户界面设计第10章 异常处理第11章 多线程第12章 Applet与Web应用开发第13章 JDBC技术和数据库应用开发第14章 输入输出流处理附录参考文献

<<Java程序设计与实验>>

章节摘录

2.goto语句 Java不像c和c++中那样有goto语句，它的转向功能是通过异常处理语句、break语句和continue语句来实现的，这些语句使得程序更具可读性和结构性。

3.指针 C和C++中通过指针可以进行任意的类型转换，但也常常带来安全问题，指针所进行的内存地址操作常常会造成不可预知的错误，从而降低了系统的安全性。

Java语言无指针类型，它利用引用来替代指针，因为系统在处理对象时要进行类型相容性检查，使用户不能直接操作任何指针，使得没有指针的程序无法操作不属于它的内存，因此提高了程序的安全性。

但是Java中不允许对引用进行加减，因为Java语言不存在进行地址操作的运算符。

4.内存管理与自动回收垃圾 C、C++中，用户可通过库函数malloc()、free()、delete()和运算符New来分配和释放内存，但是可能会造成系统的瘫痪。

而在Java中，通过运算符new为对象分配内存地址，并增加了对象和变量的强制类型检查。

当对象在其作用域内执行完毕后，内存也就不存在了，由此产生的垃圾被自动地回收，不需要用户来释放内存空间，提高了系统资源的使用效率。

5.数据类型的支持 C和C++语言的数据类型所占用的空间对于不同的平台而言会发生变化，但在Java语言中，基本数据类型所占用的空间被固定了，无论在何种平台下，总是分配固定长度的位数，保证了。

Java语言的平台独立性。

6.头文件 C和C++中维护全局变量、库函数这些头文件是很难的。

而Java类中的相关方法和变量及其类型、访问权限都抽象在一个类中，Java不支持头文件。

在系统进行访问控制时，可以隐藏与保护一些重要数据。

7.结构和联合 学过C和C++的读者知道，C和C++的结构和联合中所有成员都被声明为公共的，这就会出现安全问题。

Java不包含结构和联合，而是把所有的数据和操作都封装在一个类中。

<<Java程序设计与实验>>

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>