

<<Java开发技术大全>>

图书基本信息

书名：<<Java开发技术大全>>

13位ISBN编号：9787302192138

10位ISBN编号：7302192138

出版时间：2009-2

出版时间：清华大学出版社

作者：刘新

页数：889

字数：1423000

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开发技术大全>>

前言

Java是目前最为流行的程序开发语言。

作为一种完全的面向对象的语言，它吸取了其他语言的各种优点，设计简洁而优美，使用起来方便而高效，是一门典型的门槛低而天花板高的语言。

既像Basic那样易于入门，又像C/C++那样具有强大的实际应用能力。

而且与其他纯粹的语言不同，Java其实是一个完整的技术平台，这也就决定了学习Java和其他语言的一些区别。

市面上介绍Java的书籍很多，既包括了国外的经典名著，也包括了国内各种各样的教学书籍，但真正适合大学计算机本科专业教学以及普通程序员自己学习的书籍却很少。

国外名著由于知识背景的差异、作者的思维方式总是和中国读者有一定的距离，因此刚入门的读者无法领略其中的精妙。

而大多数的一般教学书籍，则将Java当作纯粹的语言来教学，忽视了它作为一个应用平台的强大威力，读者看了之后，难免会误会Java不过是一个精简版的C++。

作者在多年的教学过程中，深感需要编写一本能够让初学者快速入门，又能真正利用Java进行软件开发的指导性书籍。

大概从3年前起，作者就萌生了自己编写教程的想法，而真正付诸实施，则花了整整7个月的时间。

作者在自己平时所用课件的基础上，进行了大量的增改，终于编写出了本书。

本书以教程的形式深入浅出、由易到难地介绍了有关J2SE平台下的常用开发技术，每个知识点和技术都采用了例程讲解为主、理论分析为辅的方式进行介绍。

本书假设读者没有任何的编程经验，举例时也尽量避免复杂的数据结构和算法设计。

每个例子都着重于Java知识点本身，尽量浅显易懂，不涉及其他的知识。

对于初学者易犯的错误，都有明显的提示。

为了让读者养成良好的编程习惯，本书的程序代码均按照软件工程规范来编写。

全书配合了大量的程序示例、实用程序、图例及代码说明，所有例程作者均仔细调试过，确保准确无误。

作者根据多年的Java教学经验，总结出Java初学者最需要的知识以及正确的学习方法，帮助读者以最少的时间获取最大的收益。

本书特别适合于自学，也可作为计算机、软件工程专业的教材，本书的高级部分也适合作为Java程序员的参考用书。

本书的特色 本书不仅包含了Java的基础部分知识，也对它的高级技术和实用技术做了详细的介绍。

本书是根据作者多年的教学经验和软件开发经验总结出来的，将知识范围锁定在了适合初级、中级的部分。

本书以大量的实例进行示范和解说，其特点主要体现在以下几个方面。

- * 本书的编排采用循序渐进的方式，使得初级、中级学者逐步掌握复杂的编程技术。

- * 本书采用了大量的示例，讲述Java中的基本知识点。

并且尽量使这些示例简洁规范，让读者能专心于知识点上，而不被其他事物所干扰。

- * 本书提供了相当多的实例，它们都具有实际的意义，着重解决工作中的实际问题。

- * 对于有特点的实例进行详细的解释和分析，帮助读者理解和模拟实践。

- * 对于学习和编程中经常遇到的问题以及需要注意的关键点予以特别提示。

- * 按递进关系进行章节组织，介绍新知识点时与已学知识点相互比较，易于读者理解。

- * 本书采用技术要点、详细介绍、运行效果等多种方式进行讲解，系统性强、可读性强。

本书的内容安排 本书共分为8篇，先是讲述了Java的基本语法，然后介绍了面向对象的编程思想，再介绍Java中的一些高级技术。

随后，进入到实际编程阶段，先后介绍了GUI程序设计、数据库和网络程序设计，最后提供了一个完整的即时通讯软件设计实例，让读者能够从实例中学习程序设计的真谛。

<<Java开发技术大全>>

第1篇（第1章～第2章）首先全面介绍了Java的运行开发环境。其中详细讲解了JDK的安装和配置，如何使用UltraEdit来编辑一个Java源程序，以及如何编译和运行Java程序。

第2章介绍了Java的基础知识，包括数据类型、运算符与表达式、流程控制等。最后以几个实例来引导读者步入程序设计的大门。

这一篇是整个Java程序设计的基础。

第2篇（第3章～第4章）介绍了如何使用Java来进行面向对象的程序设计。包括对象和类的成员定义与使用、单继承和多重继承、运行时多态、接口、内部类、包等。本篇是Java的精华。

第3篇（第5章～第7章）介绍了Java中的数据处理。首先介绍了Java中的两个特殊类：数组和字符串。

然后介绍了Java中的异常处理机制。

最后介绍了输入和输出，包括标准设备的输入和输出以及文件的处理。

学完本篇，已经可以编写一些实用程序了。

第4篇（第8章～第13章）介绍了Java中的高级技术，这些包括多线程、RTTI、泛型、集合、类型包装以及实用工具类等。

这些内容是编写复杂实用程序的基础。

使用这些高级技术，可以大大降低编程的繁琐程度和难度。

第5篇（第14章～第15章）介绍了普通窗口程序和多媒体程序的编写。

GUI是目前最为流行的程序界面，但这类程序的编制比普通控制台程序要复杂一些。

本篇详细介绍了和GUI有关的事件、布局管理以及各种组件的使用。

并通过大量的实例来介绍如何编写一个实用的桌面程序，以及编程中的常用技巧和应该要注意的问题。

在多媒体程序设计中，则着重介绍了各种文字、图像、声音、视频的处理。

在编程中，应尽量使用Java自己提供的类，以降低编程的难度。

第6篇（第16章～第17章）介绍了数据库程序设计。

数据库编程是Java的一个重要应用方面。

本篇先介绍一般性的数据库理论，主要是SQL语句的使用。

然后详细介绍了如何使用Java中的各种类来处理数据库，并提供了一个实例说明编写数据库程序与普通程序的一些差别。

第7篇（第18章～第19章）介绍了网络程序设计。

首先介绍一般的C/S模式的网络程序设计，主要是利用Socket进行网络通讯。

随后介绍了JSP程序设计，这是Java应用的又一重要领域。

另外，本篇提供了5个实例来说明JSP程序设计中应该注意的一些问题。

第8篇（第20章～22章）是一个完整的即时通讯软件。

它综合使用了Java中的桌面程序设计、图像处理、数据库处理以及网络通讯中的各种技术，是对Java技术的一个全面应用。

通过这个软件，读者可以领略到Java的强大实用能力。

适合阅读本书的读者 * 没有任何编程基础的计算机专业学生。

* 具备一定自学能力的Java编程爱好者。

* 利用Swing开发桌面程序的Java程序员。

* 进行JSP网站开发的人员。

* 使用C/S模式设计网络程序的Java程序员。

* 想了解Java中高级技术的其他人员。

* Java语言的教学人员。

本书作者 本书由刘新主笔编写。

其他参与编写和资料整理的人员有苑令轩、王征、陈冠军、王石、程彩虹、姜海英、邵毅、张路平、

<<Java开发技术大全>>

李臻、武勇、徐宁、刘玉珊、麻雪、赵建领、陈刚、吝晓宁、范永龙、姚志娟、赵盟、傅靖、李佳、徐磊、刘丹、肖冰、陈杰、王行恒、冯浩楠、纪超、段桂东、颜盟盟、黄宝生、张珍珍、石淑珍、陈超、牛晓辉、刘聪、任潇、商斌、张双、于志华、李秀劲、李胜美、蔡文仙、杜阳阳、吴兴亮、陈水望、黄任桢、梅婷婷、皇波、白雪蛟、陈浩然、许程程、巩长宇、黄金亮、姜艳超、李军、李庆、彭志林、王志娟、武娜、尹成业等。
在此一并表示感谢。

<<Java开发技术大全>>

内容概要

Java是目前最为流行的程序设计语言。

本书以J2SE为平台全面介绍了Java的基础知识和常用技术。

《Java开发技术大全》共分为8篇，首先讲述了Java的基础语法，然后介绍了Java中类和对象的实现，这也是Java的核心所在。

随后介绍了Java中的高级技术，详细讨论了其中的多线程、集合、泛型、RTTI等。

这些基本知识介绍完毕之后，就进入到实际编程阶段，先后介绍了GUI程序设计、多媒体程序设计、数据库和网络程序设计，这些知识均以示例程序来讲解。

最后提供了一个完整的即时通讯软件设计实例，让读者能够从实例中学习程序设计的真谛。

本书内容全面，实例丰富，特别适合于自学者。

本书可作为计算机、软件工程专业的教材，也可作为Java程序员参考手册。

<<Java开发技术大全>>

作者简介

刘新，获得计算机软件与理论专业硕士，现为湘潭大学信息工程学院教师，副教授。

从事计算机专业教学11年。

主要讲授《Java程序设计语言》、《Java应用技术》、《C语言程序设计》、《数据结构》、《算法设计与分析》等专业课程。

先后参与过10多个系统的开发工作，其中包括解放军

<<Java开发技术大全>>

书籍目录

第1篇 Java基础知识入门	第1章 Java的开发运行环境	1.1 Java的运行环境与虚拟机	1.2 Java的开发环境
	1.2.1 JDK的安装	1.2.2 如何设置系统环境变量	1.2.3 编译命令的使用
	1.2.4 解释执行命令的使用	1.2.5 UltraEdit的使用	1.3 一个简单的Java应用程序
	1.4 一个简单的Java小程序	1.5 本章小结	第2章 Java语言基础
	2.1 Java语言的特?	2.2 Java程序的构成	2.3 数据类型
	2.3.1 基本数据类型	2.3.2 常量	2.3.3 变量
	2.3.4 整型数据	2.3.5 浮点型数据	2.3.6 字符型数据
	2.3.7 布尔型数据	2.3.8 变量赋初值	2.3.9 变量的作用域
	2.3.10 数据类型转换	2.4 运算符与表达式	2.4.1 算术运算符和算术表达式
	2.4.2 关系运算符和关系表达式	2.4.3 逻辑运算符和逻辑表达式	2.4.4 条件运算符和条件表达式
	2.4.5 位运算符和位运算表达式	2.4.6 赋值运算符和赋值表达式	2.4.7 表达式的求值顺序
	2.5 流程控制语句	2.5.1 三种基本控制结构	2.5.2 表达式语句和空语句
	2.5.3 块语句	2.5.4 if~else分支语句	2.5.5 多路分支switch~case语句
	2.5.6 当型循环while语句	2.5.7 直到型循环do~while语句	2.5.8 当型循环for语句
	2.5.9 循环的嵌套	2.5.10 跳转语句break	2.5.11 跳转语句continue
	2.6 程序文本的风格	2.6.1 注释	2.6.2 程序的格式编排
	2.7 基础语法实战演习	2.7.1 判断闰年	2.7.2 求最大公约数和最小公倍数
	2.7.3 Fibonacci数列	2.7.4 逆向输出数字	2.7.5 求水仙花数
	2.7.6 输出图形	2.7.7 输出九九口诀表	2.8 本章小结
	第2篇 Java面向对象编程	第3章 对象和类	3.1 面向对象的基本概念
	3.1.1 对象	3.1.2 类	3.1.3 消息
	3.1.4 面向对象的4个基本特征	3.2 类与对象	3.2.1 类的基本结构
	3.2.2 类的声明	3.2.3 创建类体	3.2.4 对象的生命周期
	3.2.5 对象的创建	3.2.6 对象的使用	3.2.7 对象的释放和垃圾收集机制
	3.3 成员变量的定义与使用	3.3.1 成员变量的定义	3.3.2 成员变量的访问权限
	3.3.3 实例成员变量和静态成员变量	3.4 方法的定义和实现	3.4.1 方法的声明
	3.4.2 创建方法体与return语句	3.4.3 局部变量和成员变量的区别	3.4.4 方法的访问权限
	3.5 方法的调用	3.5.1 方法调用的形式	3.5.2 方法调用的参数
	3.5.3 隐含参数this	3.6 构造方法	3.6.1 无参数构造方法的定义和使用
	3.6.2 带参数构造方法的定义和使用	3.6.3 this关键字和构造方法的调用	3.7 静态方法
	3.7.1 静态方法的声明和定义	3.7.2 静态方法和实例方法的区别	3.7.3 静态代码块
	3.7.4 再论静态成员变量	3.8 main()方法和命令行参数	3.9 结束方法
	3.10 本地方法	3.11 本章小结	第4章 继承与多态
	4.1 继承的基本原理	4.2 子类对父类的继承	4.3 属性隐藏和方法的覆盖
	4.3.1 属性的隐藏	4.3.2 方法的覆盖	4.4 构造方法的继承
	4.5 super的使用	4.5.1 用super引用父类的成员	4.5.2 使用super调用父类的构造方法
	4.6 继承的内部处理	4.7 多态的基本概念	4.8 重载
	4.8.1 普通方法的重载	4.8.2 构造方法的重载	4.8.3 重载的解析
	4.8.4 重载与覆盖的区别	4.9 运行时多态	4.9.1 实例方法的运行时多态
	4.9.2 成员变量运行时的表现	4.9.3 静态方法运行时的表现	4.10 抽象类与抽象方法
	4.10.1 抽象方法的声明	4.10.2 抽象类的定义	4.10.3 抽象方法与回调函数
	4.11 最终类与最终方法	4.12 接口	4.12.1 接口的定义
	4.12.2 接口的继承	4.12.3 接口的实现	4.13 内部类
	4.13.1 内部类的定义	4.13.2 内部类访问外部类的成员	4.13.3 内部类之间的相互使用
	4.13.4 在外部使用内部类	4.13.5 匿名内部类	4.13.6 内部类的作用
	4.14 包	4.14.1 包的创建	4.14.2 包的使用
	4.14.3 JAR文件的创建和使用	4.14.4 JDK中的常用包	4.15 本章小结
	第3篇 Java数据处理	第5章 数组与字符串	5.1 数组
	5.1.1 一维数组的声明	5.1.2 一维数组的创建	5.1.3 一维数组的使用
	5.1.4 二维数组的声明	5.1.5 二维数组的创建	5.1.6 二维数组的使用
	5.1.7 for~each循环	5.1.8 应用程序示例1——数组排序	5.1.9 应用程序示例2——杨辉三角
	5.2 字符串	5.2.1 String对象的声明	5.2.2 String对象的创建
	5.2.3 String对象的使用	5.2.4 String类型的数组	5.2.5 StringBuffer对象的声明
	5.2.6 StringBuffer对象的创建	5.2.7 StringBuffer对象的使用	5.3 本章小结
	第6章 Java的异常处理	6.1 异常的基本概念和作用	6.2 Java的异常处理机制
	6.3 异		

<<Java开发技术大全>>

常类的层次结构 6.3.1 运行时异常 6.3.2 检查型异常 6.3.3 自定义异常 6.4 捕获和处理异常 6.5 抛出异常 6.6 try-catch-finally语句的嵌套 6.7 用户自定义异常 6.8 使用异常来实现键盘输入 6.9 本章小结 第7章 Java的输入和输出 7.1 文件和输入输出流 7.2 InputStream类和OutputStream类的使用 7.2.1 InputStream中的方法 7.2.2 OutputStream中的方法 7.2.3 文件输入流FileInputStream 7.2.4 文件输出流FileOutputStream 7.2.5 保存用户输入到文件 7.2.6 显示文件内容 7.2.7 文件的复制 7.2.8 顺序输入流 7.2.9 管道输入输出流 7.2.10 过滤输入输出流 7.3 Reader和Writer类使用 7.3.1 InputStreamReader类中的方法 7.3.2 OutputStreamWriter类中的方法 7.3.3 从键盘输入 7.3.4 文件复制 7.4 文件属性处理 7.4.1 File类中的方法 7.4.2 File类使用示例——文件列表 7.5 随机访问文件 7.5.1 RandomAccessFile类中的方法 7.5.2 RandomAccessFile类使用示例——文件加密 7.6 控制台输入和输出 7.6.1 控制台输入类Scanner 7.6.2 格式化输出printf. 7.7 序列化 7.8 本章小结第4篇 Java中的高级技术 第8章 Java的多线程机制 8.1 线程的概念 8.1.1 多线程的特点 8.1.2 线程的状态 8.2 Thread类 8.2.1 Thread类的构造方法 8.2.2 Thread类的常用方法 8.3 多线程程序的编写 8.3.1 利用Thread的子类创建线程 8.3.2 实现Runnable接口创建线程 8.3.3 使用isAlive()和join()等待子线程结束 8.3.4 设置线程优先级 8.4 线程的通信与协调 8.4.1 线程的互斥 8.4.2 线程的同步 8.4.3 暂停、恢复和停止线程 8.4.4 生产者-消费者问题实例 8.5 本章小结 第9章 运行时类型识别 第10章 泛型 第11章 集合 第12章 类型包装器、自动装箱和元数据(注释) 第13章 常用工具类第5篇 桌面程序开发 第14章 GUI程序设计 第15章 多媒体程序设计第6篇 数据库程序设计 第16章 数据库的预备知识 第17章 Java对数据库编程第7篇 Java网络程序开发 第18章 C/S网络程序设计 第19章 JSP程序设计第8篇 即时通讯系统开发实例 第20章 系统分析和设计 第21章 服务器端功能模块的实现 第22章 客户端功能模块的实现

章节摘录

第1篇 Java基础知识入门 第1章 Java的开发运行环境 学好Java最重要的一个步骤就是上机编程,熟悉Java,的开发运行环境是成为Java程序员的第一步。本章将详细介绍如何安装并配置好Sun公司提供的JDK1.5 for Windows,如何编制一个简单的Java程序,如何编译一个Java源程序,如何运行编译好的class文件以及如何避免初学者常犯的错误。通过本章的学习,将轻松地迈入Java的殿堂。

1.1 Java的运行环境与虚拟机 任何一个可执行文件,都必须在某个平台上才能运行。例如,Windows下的exe文件,必须在Windows环境下、X86硬件平台上才能运行。这些exe文件,通常是使用C / C++、Pascal或vB等语言编程,然后通过编译、链接而形成的。在这些可执行文件中,包含了运行它的硬件平台的相关信息,所以如果要把它移植到其他平台上,必须要重新编译,甚至要修改源文件。

Java和这些语言不同,它的最大特点就是平台无关性。Java文件经编译后,生成的是一个后缀名为class的文件。这是一种字节码文件,它不像普通可执行文件那样包含硬件信息,而是完全与硬件平台无关,也就是无法直接由操作系统调用运行。所以Java的应用程序,需要一个更为复杂的平台才能运行。这个运行平台,包括计算机操作系统、适配器、Java虚拟机、Java基本软件和Java应用程序接口,它们负责将Java的字节码翻译成硬件可以接受的指令。整个运行系统的结构如图1.1所示。

Java运行系统执行Java应用程序的过程包括三个步骤:首先是代码的装入,然后是代码的验证,最后是代码的执行。

代码的装入由类装配器完成,它主要负责装入程序运行时所需要的全部代码,包括程序运行中调用到的其他类。

当类装配器装入一个类后,该类被放在自己的命名空间中,除了通过符号引用其他类之外,该类不能影响其他类所在的空间。

当装入了运行程序所需要的所有类后,运行系统就能确定整个可执行程序的内存在布局。

然后,被装入的代码将由字节码验证器进行安全检查,以确保字节代码不存在违反访问权限、不规范的数据类型和非法调用等问题。

通过校验之后,代码就可以运行了。

Java的字节码有两种运行方式: 解释执行方式:通过“解释器”,将字节码翻译成机器码,然后由“运行支持库”将机器码送往硬件执行。

整个执行过程是一边翻译一边执行,称为解释执行。

Java系统一般采用这种方式。

即时编译方式:通过“代码生成器”,先一次性地将所有字节码翻译成适用于特定计算机系统的机器码,然后送往硬件执行。

对程序运行速度要求较高时,一般采用这种方式。

从图1.1可以看出,Java的字节码并没有直接运行在硬件平台上,而是在一个虚拟的软件平台上运行。

这个虚拟的软件平台,被称为Java虚拟机(Java Virtual Machine, JVM)。

为了让编译产生的字节码可以更好地解释与执行,通常把JVM分成6个功能模块:JVM解释器、指令系统、寄存器、栈、存储区和碎片回收区。

JVM解释器:JVM解释器负责将字节码转换成为CPU,能执行的机器指令。

指令系统:指令系统同硬件计算机很相似。

一条指令分成操作码和操作数两部分。

操作码为8位二进制数,操作数可以根据需要而定。

操作码是为了说明一条指令的功能,所以JVM可以有高达256种不同的操作指令。

<<Java开发技术大全>>

寄存器：JVM有自己的虚拟寄存器，这样就可以快速地和JVM的解释器进行数据交换。

为了实现必需的功能，YVM设置了4个常用的32位寄存器：pc（程序计数器）、optop（操作数栈顶指针）、frame（当前执行环境指针）和vars（指向当前执行环境中第一个局部变量的指针）。

栈：JVM：栈是指令执行时数据和信息存储的场所和控制中心，它提供给JVM解释器运算时所需要的信息。

存储区：JVM存储区用于存储编译后的字节码等信息。

碎片回收区：JVM碎片回收，是指将那些使用后的Java.类的具体实例从内存中进行回收。

因此，可以避免开发人员自己编程控制内存的麻烦。

随着JVM的不断升级，其碎片回收技术和算法也更加合理。

比较经典的算法有引用计数、复制、标记-清除和标记.整理。

在JVM 1.4.1版以后，产生了一种代收集技术。

简单地说，就是利用对象在程序中生存的时间划分成代，以这个代为标准进行碎片回收。

说明：JVM的运用，真正让Java实现了“一次编译，处处运行”，它是整个运行系统的核心。

<<Java开发技术大全>>

编辑推荐

《Java开发技术大全》是根据作者十多年的教学经验和软件开发经验总结而来，全方位解读Java语言基础知识、高级技术和应用开发。

由多年Java教学经验和开发经验的人员全力打造，每个技术点都配有典型实例进行讲解，非常容易掌握，对学习中容易犯的错误给出了明确的提示，让你少走弯路，程序代码按软件工程的规范编写，培养良好的编程习惯，所有实例都经过了仔细调试，确保可以正常运行，提供450个示例、6个小案例、1个大案例，可作为案头必备的查询手册。

一线开发人员全力打造，分享技术盛宴！

重点内容及特色： 《Java开发技术大全》全面介绍了、Java语言的基础知识和应用开发技术，基本涵盖了J2SE的各方面。

对基础知识的介绍深入、Java语言的核心，对实用技术的介绍则紧密联系实际开发。

《Java开发技术大全》对Swing包、实用工具类、集合、泛型等内容的介绍全面而深入，已经出版的同类图书很少有如此透彻介绍这些内容的。

《Java开发技术大全》讲解深入浅出，由易到难，循序渐进，书中的每个知识点和技术都采用了例程讲解为主，理论分析为辅的方式进行介绍。

《Java开发技术大全》举例时尽量避免复杂的数据结构和算法设计，每个例子都着重于Java知识点本身，做到浅显易懂。

对于初学者容易犯的错误，都有明显的提示。

和国外的一些经典图书相比，《Java开发技术大全》门槛低，讲述符合中国人的思维，更容易理解。

另外，《Java开发技术大全》如此大量的实例在国外的经典图书中是很难见到的。

读者对象：想要全面学习Java语言的人员，具备一定自学能力的Java编程爱好者，Java培训班的培训学员和老师，想了解Java中高级开发技术的人员，需要一本案头必备查询手册的程序员。

<<Java开发技术大全>>

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>