

<<探秘Java>>

图书基本信息

书名：<<探秘Java>>

13位ISBN编号：9787308040365

10位ISBN编号：7308040364

出版时间：2004-12

出版时间：浙江大学出版社

作者：唐尼

页数：390

译者：张平

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>

前言

从其他人的发明中我们获得了巨大的利益，所以，我们也应该让其他人享用自己的发明成果，并且应该是自由享用和无私慷慨的。

——本杰明·富兰克林此书的缘起 自1999年开始，我一直在科尔比学院教书。

那时就写了此书，今天呈现在读者面前的已是第四版了。

那个时候，当我想用Java程序设计语言，向学生们介绍计算机科学的基础知识时，却怎么也找不到一本称心如意的教材。

为什么呢？

因为这些著作的“块头”都太大了，班上的学生们根本无法阅读这些厚达800多页的、满是技术名词术语的“砖头”大书，即使我强制要求他们这么做，他们也还是力不从心。

当然我没有理由、也不忍心让他们这样做。

同时，这些书中的材料都太专门、太琐碎了——包括了各种各样Java语言和库函数的细节——以致到学期结束后，学生也会早已模糊和遗忘。

另外一个问题是，这些教材在向学生介绍面向对象的程序设计方式时，显得太过唐突和生硬。

有许多对学习过程式语言非常适应、成绩也非常好的学生，一旦学到面向对象的设计方法时，就像是碰到了一堵墙，竟然无法逾越。

我也尝试在程序语言课一开始时，就教学面向对象的设计方法，或者在中间，甚至在最后，但效果却全都一样。

内容概要

Java语言是目前最流行的面向对象的编程语言。

它具有强大的功能，既能面向一般跨平台的工程开发，又能面向因特网应用，为广大编程者和学习者所青睐。

《探秘Java：如何像计算机科学家一样思考》的特点有四：一是用类似英语教学中的情景教学方法，尽可能用简洁明快的方式按部就班地叙述。

二是多用读者日常生活中随手可拈的趣味性实例作为编程材料。

三是把计算机科学家们在编程实践中总结出来的诀窍“捅”给读者。

四是所有的习题均给出答案，并加以简略分析。

《探秘Java：如何像计算机科学家一样思考》从结构上可分两部分。

其中1-13章为Java语言的基础部分，循序渐进地介绍Java语言的各种要素：变量、操作符、条件语句、循环语句、嵌套与递归、引用、数组、结构、类以及对象等。

14-20章为数据结构部分，分门别类地介绍各种常用的数据结构：链表、堆栈、队列、优先队列、树、堆、映射表、哈夫曼树以及文件等。

阅读《探秘Java：如何像计算机科学家一样思考》的读者可以是真正的初学者，可以是在校学习的研究生、本科生或大专生，也可以是企、事业单位的初、中级用户。

最合适的读者是非计算机专业的大学本科生或大专生，以及中、小学生中信息技术的爱好者。

书籍目录

前言此书的缘起此书的宗旨面向对象的程序设计数据结构计算机AP考试有关此书的版权关于此书书名

第1章 按部就班的方式1.1 什么是程序语言1.2 什么是程序1.3 什么是调试1.4 形式化语言和自然语言1.5 第一个程序1.6 术语1.7 练习第2章 变量和类型2.1 再打印一些东西2.2 变量2.3 赋值2.4 打印变量2.5 关键字2.6 操作符2.7 操作符的执行顺序2.8 对字符串的操作2.9 组合句2.10 术语2.11 练习第3章 方法3.1 浮点数3.2 把double类型转换成int类型3.3 数学函数3.4 组合3.5 自己编写新方法3.6 类和方法3.7 具有多个方法的程序3.8 形式参数和实际参数3.9 堆栈状态图3.10 具有多个参数的方法3.11 具有结果的方法3.12 术语3.13 练习第4章 条件和递归4.1 模除运算4.2 条件执行4.3 选择执行4.4 链式条件4.5 嵌套条件4.6 返回语句4.7 类型转换4.8 递归4.9 递归调用方法的堆栈图4.10 惯例和神圣规则4.11 术语4.12 练习第5章 “开花结果”的方法5.1 返回值5.2 “逐渐生长”的程序5.3 组合5.4 重载5.5 布尔函数5.6 逻辑运算符5.7 布尔方法5.8 递归的例子5.9 确信跳跃5.10 另一个递归例子5.11 术语5.12 练习第6章 重复6.1 变量的多次赋值6.2 重复6.3 while语句6.4.表格6.5 两维表6.6 封装和泛化6.7 方法6.8 进一步封装6.9 局部变量6.10 进一步泛化6.11 术语6.12 练习第7章 字符串和其他7.1 调用依附于某个对象的方法7.2 长度7.3 遍历7.4 运行错误7.5 司读说明文档7.6 indexOf方法7.7 循环和计数7.8 加一和减一操作符7.9 字符的算术7.10 不可更改的字符串7.11 字符串之间不能直接比较7.12 术语7.13 练习第8章 趣味对象8.1 有趣的是什么8.2 包8.3 Point对象8.4.实例变量8.5 把对象作为参数8.6 矩形8.7 把对象作为返回值8.8 对象是可变动的8.9 别名8.10 空引用8.11 垃圾回收机制8.12 对象和基类8.13 术语8.14 练习第9章 定做对象9.1 类的定义和对象类型9.2 T1me类9.3 构造器9.4 编写多个构造器9.5 生成新对象9.6 打印对象9.7 操纵对象9.8 无瑕作用9.9 改动作用9.10 填入作用9.11 哪一种更好9.12 发展型风格vs规划型风格9.13 泛化处理9.14 算法.....第10章 数组第11章 对象数组第12章 数组对象第13章 面向对象程序设计第14章 链表第15章 堆栈第16章 队列和优先队列第17章 树第18章 堆第19章 映射表第20章 哈夫曼码附录A 按部就班开发程序附录B 除错附录C 输入与输出附录D 图形附录E 与JAVA环境混个脸熟附录F 太过完美的洗牌附录G 练习分析与解答附录H GUN Free Documentation License

章节摘录

神圣规则：这是我的用语。

这些规则是由逻辑或数学里的内部定律所制约而必须得以遵守的，而且它们也是任何程序设计语言（或者其他形式化语言）所必须具备的。

例如，在指定一个矩形边界的位置和尺寸时，少于四个参数就无法指定。

另一个例子是，加法运算是符合交换律的，这是加法定义的一部分，对此，Java无法对之进行任何改动。

Java规则：这是使用Java语言所必须遵守的语法和逻辑规则，你我也无法违背，不然的话，所编写的程序就无法编译或者无法运行。

但某些时候还是具有一定的灵活性，比如，“+”号对于数字意味着“加”，对于字符串意味着“串接”。

但是，另外一些却是编译器和执行过程内部限制的反映。

例如在编制方法时，你必须指定形式参数的类型，但在调用方法时，你却绝对不能指定实际参数的类型。

风格与惯例：这些规则并非由编译器强制执行，但是遵守它却是编写正确程序的基本要求；遵守它可以容易地调试程序，也可以方便地修改程序；遵守它，其他人也容易阅读程序，等等。

这样的例子包括行与行之间的缩进要求、放置大括号的位置，以及对变量、方法和类的命名约定。

在继续行文时，我总努力地指出各种各样的东西落入了哪种规则，你呢，也必须时不时地思索一下遇到的东西应归于哪种范畴中。

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>