

新起点 电脑教程

中国计算机职业教育联盟推广使用教材

Java 2

程序设计基础教程

与 上 机 指 导

计算机职业教育联盟 主编
蔡勇 姜磊 马宁 倪文志 编著

- 概念清晰，语言通俗。易于理解Java语言的基础知识
- 精选范例，循序渐进。便于准确掌握Java应用程序的编程方法
- 教学与练习相结合。适合各种电脑培训班作为教材使用



清华大学出版社

新起点电脑教程

Java 2 程序设计基础教程 与上机指导

计算机职业教育联盟

主编

蔡 勇 姜 磊 马 宁 倪文志

编著

清华大学出版社

北 京

内 容 简 介

Java 经过多年发展,现在已经真正成长为严格的、主流的开发语言。本书结合大量实例,由浅入深地对 Java 2 及面向对象编程的基础知识、Java 的数据结构和 Java 2 的高级内容——异常处理、I/O 操作、GUI 编程、Java Applet、网络编程、线程、数据库操作等内容进行了系统的介绍,并结合大量示例演示了应用程序的编程方法与技巧,目的是使读者能迅速、准确地了解和掌握这门编程语言。每章最后都配有上机指导和一组练习题,帮助读者巩固本章所学的内容。

本书选例典型,针对性强,基本概念清楚,语言简明通俗,内容生动翔实,行文循序渐进,特别适合各类培训班和高职高专的相关专业作为教材使用,也适合于自学 Java 的读者。

版权所有,翻印必究。举报电话:010-62782989 13901104297 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用清华大学核研院专有核径迹膜防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

Java 2 程序设计基础教程与上机指导/计算机职业教育联盟主编,蔡勇,姜磊,马宁,倪文志编著.

—北京:清华大学出版社,2005.1

(新起点电脑教程)

ISBN 7-302-09883-2

I. J… II. ①计… ②蔡… ③姜… ④马… ⑤倪… III. JAVA 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 114376 号

出 版 者:清华大学出版社 地 址:北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编:100084

社 总 机:010-62770175 客 户 服 务:010-62776969

组稿编辑:林章波

文稿编辑:杨作梅

封面设计:陈刘源

印 刷 者:北京密云胶印厂

装 订 者:北京市密云县京文制本装订厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印 张:23.25 字 数:548 千字

版 次:2005 年 1 月第 1 版 2005 年 1 月第 1 次印刷

书 号:ISBN 7-302-09883-2/TP·6805

印 数:1~5000

定 价:30.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或 (010)62795704

序 言

一、中国职业培训行业面临的契机和挑战

中国高等职业教育和职业培训服务近年来得到了快速发展,为经济社会发展作出了巨大的贡献。中国正式成为世界贸易组织(WTO)成员后,整个中国教育界尤其是职业教育界都将面临一个新的契机和挑战。我国的职业教育主要包括学校职业教育和职业培训两种模式。学校职业教育基本是正规学历教育,已经不能适应目前经济的高速发展、职业多变和终身教育的需要。随着入世后就业结构调整和技术技能折旧速度的加快,劳动力跨行业流动更加频繁,职业培训进一步成为涉及面最广、受益面最大的教育,从而越来越成为职业教育的重要组成部分,越来越受到人们的广泛欢迎。

加入 WTO 对中国职业培训将产生积极的影响,至少表现在以下几个方面:

- 加快引进国外优质职业教育资源,有利于推动职业教育办学体制和运作机制改革,提高职业教育的整体水平;
- 有利于吸收国外多种职业教育模式,加快改进职业教育的办学模式、课程体系和教学培训方法,形成具有多元化特色的职业教育体系;
- 引进国外职业资格证书及其培训模式,加快完善我国的职业资格证书制度,有利于培训内容及培训资格的国际化;
- 经济增长和产业结构调整将扩大职业培训和继续教育的市场需求,有利于职业培训的持续发展;
- 加快教育的市场化步伐,有利于推进相关的法制建设进程,改变人们的传统教育观念,提高职业培训的社会地位。

同时,境外职业教育和培训机构的进入必将对职业教育领域造成巨大的冲击。境外职业教育与培训机构十分看好中国的职业培训和继续教育市场。此前由于教育服务业开放方面的原因,多为依托跨国公司在开展经济活动的同时建立培训机构,或通过职业资格证书培训来拓展职业教育市场,入世后放宽境外职业教育培训机构合作办学的条件,并允许其获得多数拥有权,这将大大加快其入境提供职业教育服务的步伐。他们凭借丰富的职业教育和培训经验,及其国际通用的职业资格证书,使中国职业教育服务领域形成新的竞争格局,一批办学条件差、专业设置陈旧及教学水平落后的职业教育和培训机构将被淘汰出局。

二、社会发展对计算机培训行业提出的新要求

IT 行业的发展和激烈的人才竞争对 IT 从业人员的综合素质提出了越来越高的要求,总结起来可以归纳为五种“基本能力”与三项“基本素质”。五种基本能力包括合理利用与支配各类资源的能力;处理人际关系的能力;获取并利用信息的能力;综合与系统分析能力;运用各种技术的能力。三项基本素质包括基本技能、思维方式与个人品质。在市场经

济体制和新技术飞速发展的条件下, 计算机人才只掌握一门特定的技能已经远远不够。必须全面强调学生的基础知识、基本能力和基本素质, 而且专业面要宽, 以适应技术进步与市场的不断变化。例如电脑平面设计师, 像 2000 年以前那样只掌握 1~2 种常用的平面设计软件已经远远不能胜任工作, 还必须具备良好的手绘功底和创意能力, 这就对整个计算机培训行业提出了更高的要求。被广大培训机构广泛采用的只重知识传授, 不重能力培养的教学模式已不能适应社会对计算机培训行业提出的要求。计算机培训必须进行从知识传授到能力培养的转变, 全面提高学生的职业能力, 即其在就业后的工作及配合中综合运用知识与技术的能力: 能够做到迅速收集、分析、归纳意见与信息, 与他人交流思想与信息, 制定计划与组织活动, 与他人合作共事, 解决实际问题, 学习和使用最新的技术, 成为适合社会进步的人才。

三、计算机培训教学的新模式和新探索

社会对学生的职业能力要求催化出新型的课程结构和教学模式。新型教学模式必须是以工作为基础的模仿学习, 它是将学生置于一种模拟环境中, 呈现给学生的是具有挑战性、真实性和复杂性的问题, 使学生在身体和经济不受到损失的前提下, 得到较真实的锻炼。新型课程结构必须按照职业能力的要求创建并组织实施新的教学模式。教学以专项能力的培养展开, 以综合能力的形成告终, 能力的培养既是教学目标, 又是评估的依据和标准。在能力标准的制定、考核与评估中, 均要有企业或行业参与。

新型课程结构在具体的实施中又派生出两大结构, 即模块式结构与阶梯式结构。模块式结构即按照程序模块化的构想和原则设计课程。其特点是每个模块均为独立的教学单元, 有特定的目标和评估标准, 教学时间相对较短。模块之间组合灵活, 基础性模块可与其他功能模块组成任务更复杂的课程, 不同模块的增删可调整课程结构的重心, 以满足个人和职业需求的变化。上述这些特性适应了新型职业能力的教学要求, 部分计算机培训机构已经开始尝试采用这种课程结构。

阶梯式结构最典型的就是所谓的双元培训, 它把整个培训课程分为两个阶段, 首先集中学习文化课和职业基础课, 第二阶段转入职业领域的专业训练和专向深化。这种以较广泛的科学文化和职业理论为基础, 逐步深化职业知识与技能的课程结构, 在欧洲国家的职业教育体系中采用较多, 在中国的计算机培训机构中基本上没有采用。

以上两种课程结构虽模式不同, 但目标一致, 即意在使学生以较宽厚的基础和实力去迎接未来多变且多元的就业市场。

长城计算机学校充分吸取了国外职业教育机构的经验, 并对大量企业进行了科学广泛的调研分析, 然后结合模块式和阶梯式两种结构的优点, 从 2001 年起, 设计推出了一系列面向就业的计算机培训课程, 并且在教学实践过程中不断探索先进的教学方法, 如发现式教学法、启发式教学法、体验式教学法、行为导向教学法、任务驱动教学法等, 在大量的实践基础上不断完善课程。经过在各地分校和合作学校的长期推广使用, 得到了巨大的成功。

清华大学出版社从建社伊始, 就致力于高新技术, 特别是计算机与信息技术类图书的出版。20 多年来, 出版社坚持和发展自己的优势与特色, 围绕“开展全方位出版, 形成总体优势”的指导方针制定选题规划, 重点抓教材与专著的出版, 形成品牌与规模。为了适

应信息技术飞速发展和广泛应用的特点,出版社及时出版了一大批计算机应用技术的图书并已形成品牌。为了适应各层次,各类专业人员的需要,出版社组织了门类齐全,层次丰富的多种系列丛书,同时配合电子音像出版物,做到全方位、立体化出版。在国内计算机图书的出版中,清华版图书占有明显的优势。清华版计算机图书以其内在的品质,已经在读者心目中树起精品品牌,享有很高的声誉。

为了共同推进国内计算机培训行业的入世步伐,长城计算机学校与清华大学出版社亲密合作,共同开发了《新起点电脑教程》,在总结了百所分校的教学经验的基础上,把教学模式和教学成果推向市场,造福社会,并借以与广大教育、培训业专家交流经验,共同提高。本套丛书贯穿了清华大学出版社一贯的严谨、科学的图书风格,融会了长城计算机学校实用、专业的教学理念,是对计算机教学的一种深入、广泛的探索和总结,也得到了信息产业部领导和专家的首肯。我们相信,此套丛书的出版,必将对整个计算机培训市场的发展和规范起到巨大的推进作用!

四、丛书特点

《新起点电脑教程》丛书定位于计算机初、中级用户。不管是培训班学员还是大中专院校的师生,都可以通过本丛书快速进入计算机科学的大门,学到实用的计算机技能;对于自学者,本套教程也有很大的参考价值,大量实用技巧可供自学者在工作中随时参阅。

本套丛书主要具有如下几个方面的特点:

1. 系统性和丰富性

本丛书的所有课程都是围绕着职业素质训练展开的。我们根据计算机相关职业把计算机教程划分为四大类别:

- (1) **应用类**: 主要面向广大计算机家庭用户、企事业单位的文员、秘书和行政助理、打字排版人员等电脑应用人员;
- (2) **设计类**: 主要面向平面设计师、网页设计师、三维动画设计师等电脑设计专业人员;
- (3) **网络类**: 主要面向网络管理员、系统集成工程师、安全工程师等网络类专业人员;
- (4) **编程类**: 主要面向软件开发工程师、软件测试工程师等编程类专业人员。

以上四类内容基本涵盖了计算机应用的主要领域。本丛书的选题既考虑了每门课程本身的完整性,又兼顾了课程间的联系与衔接。每一本书可能都自成一体,完全满足相应课程的教学要求,使得培训学校或读者可以根据需要灵活的进行选择 and 组合,满足个性化学习的需要。不管读者是什么样的情况,都能在本丛书中找到自己需要的教程。

2. 先进性和探索性

一般的计算机教材往往只注重课程本身的体系结构和语言的科学准确,术语连篇,让读者感觉高深莫测,而这种做法实际上忽略了人的认知规律:人的学习应该是由具体到抽象、由已知到未知的过程。

本丛书的编写充分融入了教育心理学的最新成果,在内容的安排和教授方法上进行了大胆的尝试。在介绍各个知识点时,大量使用了教学中常用的比喻、类比和举例等方法,在内容的先后顺序上由浅入深、层次分明,使得知识的讲解变得深入浅出,易于理解。

按照成人教育的规律,本丛书每章前配有学习的目标以及重点难点,让读者有明确的目标,了解学习过程中应该重点掌握和比较难以理解以及容易混淆的知识点。在正文的实例演示的操作中,还在必要和适当地方加上“注意”、“技巧”等内容,以让初学者少走弯路,快速掌握到知识的精髓。

从教育心理学的角度来看,遗忘率最高的是知识,技能其次,能力最低。技能、实践必须从亲自动手中得来。老师讲的,学生往往只是一种平面的、苍白的理解。只有自己亲自动手实践的才是有色彩的、立体的、生动的理解和感受。真正的技能源自动手与实践!所以本丛书每章正文后面都配有上机指导,列举了大量实例,并给出操作步骤和实验结果,供读者上机练习时使用。另在每一章的最后安排了习题,包括填空题、判断题、选择题、简答题、操作题等各种形式,供课后练习和巩固知识之用。

另外,丛书的每一本教材都有配套的幻灯片,培训教师可以从网站上下载,满足教学的需要。每本书大约 15~20 章左右,符合短期培训的课时安排规律;每一章 15~20 页上下,符合授课时的教学内容的量的需要,所以本套丛书完全可以满足各种教学情况。

3. 实用性和专业性

传统教学中的最大的缺点就是理论强于实践,实践和应用脱节。而职业教育的特点要求必须加强对学生自学能力和实践动手能力的培养,在教学中鼓励学员探索和分析归纳创新习惯的养成。并且,将学生学习的技术技能同他将来的工作岗位紧密的结合起来。实现了学生学习与职业间的“无缝连结”。

本丛书的内容是在仔细分析初学者学习电脑的困惑和目前电脑图书市场现状的基础上确定的,围绕实际应用和就业需要选择内容。丛书中选取的大量实例都是在企业的实际工作案例的基础上改编来的,让读者在学习每个知识点时能“避虚就实”,在练习实例的同时,就掌握了相应行业的职业规范,学到真正有用的东西。读者只要按书中的实例和上机指导的方法去做成、做会、做熟,就能举一反三,学以致用,从而能真正完成“专业”的作品和项目,就业时能够迅速进入工作岗位。

本套丛书从策划选题到编写校对,全部由多年从事计算机教学的专业老师和图书出版业专家协同完成,并经过业内资深专家审核。我们希望这里的每一本教程都能成为您在学习计算机的道路上一个新的起点,也成为您人生道路上的一个新的起点!

前 言

Java 是一种面向对象的具有众多功能的高级编程语言,掌握了 Java 语言的程序员可以容易地开发复杂而又方便维护的程序。从 1995 年由 Sun Microsystems 推出 Java 至今,已有数以万计的程序员认识到 Java 语言的强大魅力而投入到 Java 的学习和开发中,Java 语言凭着其独立于平台,同 Web 的强交互性,支持线程等强大功能取得了显著的成就,随着 Java 语言的不断发展和成熟,成为未来主流开发语言的趋势越来越明显。

本书编写的目的旨在提高 Java 语言的教学质量,改善客观上存在的 Java 语言难讲述、难学习、难掌握的现状,着眼于培养学生独立编程能力和驾驭程序设计语言的能力,从实用和独立操作出发,循序渐进地讲解 Java 语言方面的知识,各章均配有附有答案的习题,努力为课堂教学过度到独立编程提供模拟环境。

本书共分 14 章,第 1 章介绍进行 Java 编程的入门知识,包括面向对象程序设计思想,Java 特有的虚拟机机制,Java 编程环境的搭建以及一些编程工具的使用;第 2 章介绍 Java 编程基础,讲述了关键字、变量、运算符、修饰符、流控制和数组;第 3 章介绍 Java 类、继承、创建使用对象、多态等内容;第 4 章介绍接口的定义和使用,以及接口的多重继承和嵌套;第 5 章介绍异常的定义和处理;第 6 章介绍 Java 语言对字符串的处理;第 7 章介绍一些常用 Java 集合,包括集合接口和集合实现;第 8 章介绍 Java 图形界面以及 GUI 组件的使用;第 9 章介绍 Java 的 Web 应用工具 applet,包括 applet 的基本特征,applet 的实现,以及 Applet[y1]通信;第 10 章介绍 Java 的 I/O(输入/输出)系统,包括对文件的基本操作,Java 中流的概念及其应用,以及随机访问文件;第 11 章介绍多线程的内容,包括线程的概念,线程的创建,线程间的同步与通信,以及线程的生命周期和状态控制;第 12 章介绍 Java 的网络编程技术,包括网络基本知识,Java 中网络资源的表示以及基于 TCP 和 UDP 的网络通信技术;第 13 章介绍 Java 访问数据库的机制 JDBC 以及访问数据库的方法;第 14 章简单介绍 Java 的编程规划和 Java 2 类库。

本书由蔡勇、姜磊、马宁、倪文志、刘绪崇、陈天霞、曾详明、谌跃飞、周玲、杭志、刘泽栋、陆昌红、杨国锴、张颖华、刘杰、彭高、温建国、易军等编写。由于编写时间仓促和作者水平有限,书中错误、缺点在所难免,欢迎广大读者提出宝贵的建议和意见。

编者

2004 年 5 月 20 日

目 录

第 1 章 Java 开发基础.....	1	2.3.2 运算符的优先级.....	28
1.1 面向对象程序设计思想.....	1	2.4 修饰符.....	29
1.1.1 面向对象的基本特征.....	1	2.5 程序的流程控制语句.....	31
1.1.2 面向对象程序设计步骤.....	3	2.5.1 顺序结构.....	31
1.2 Java 的虚拟机机制.....	4	2.5.2 分支语句.....	31
1.3 搭建 Java 开发环境.....	5	2.5.3 循环语句.....	34
1.3.1 安装 J2SDK.....	5	2.5.4 跳转语句.....	36
1.3.2 设置 J2SE.....	6	2.5.5 灵活运用 return 语句.....	38
1.3.3 J2SDK 帮助文档的利用.....	7	2.6 数组.....	39
1.4 应用程序流程.....	8	2.6.1 数组的定义及初始化.....	39
1.4.1 建档应用程序 Javadoc.....	8	2.6.2 使用数组时应该注意的 问题.....	41
1.4.2 反编译工具的使用.....	10	2.6.3 多维数组.....	42
1.5 上机指导.....	11	2.7 上机指导.....	43
1.5.1 环境测试.....	11	2.7.1 一个经典算法.....	43
1.5.2 程序调试.....	16	2.7.2 值求和问题.....	43
1.6 习题.....	18	2.7.3 冒泡排序问题.....	44
第 2 章 Java 编程基础.....	20	2.8 习题.....	44
2.1 Java 基本语法规式.....	20	第 3 章 类.....	46
2.1.1 关键字.....	20	3.1 包.....	46
2.1.2 Java 中的标识符.....	21	3.1.1 包及包中类的导入.....	46
2.1.3 Java 语言中的基本数据 类型.....	21	3.1.2 自定义程序库.....	48
2.1.4 Java 中的命名规则.....	22	3.2 类.....	49
2.1.5 Java 中的一些注意事项.....	22	3.2.1 类的定义.....	49
2.2 变量及变量的作用域.....	23	3.2.2 成员变量.....	51
2.2.1 变量的定义.....	23	3.2.3 成员方法.....	52
2.2.2 Java 的变量类型.....	23	3.2.4 类的组合.....	53
2.2.3 基本数据类型之间的转换.....	23	3.2.5 类的继承.....	55
2.2.4 变量的作用域.....	24	3.3 对象.....	58
2.3 运算符与表达式.....	25	3.3.1 创建对象.....	58
2.3.1 运算符的分类.....	25	3.3.2 使用对象.....	59
		3.3.3 访问对象.....	60

3.3.4 this 语句.....	62	6.1.1 String 类字符串的构造.....	97
3.4 多态的灵活运用.....	64	6.1.2 获取字符串信息.....	99
3.5 上机指导.....	65	6.1.3 字符串比较.....	100
3.5.1 一个简单的加密类的 创建与使用.....	65	6.1.4 字符串提取.....	103
3.6 习题.....	66	6.1.5 字符串的修改.....	105
第 4 章 接口	69	6.1.6 字符串的连接.....	107
4.1 接口的定义.....	69	6.1.7 字符串转换和 toString.....	108
4.1.1 接口的声明.....	69	6.2 StringBuffer 类.....	109
4.1.2 接口体.....	69	6.2.1 创建 StringBuffer 类对象.....	109
4.1.3 抽象类.....	70	6.2.2 StringBuffer 类对象的操作.....	109
4.2 接口的实现和使用.....	71	6.3 上机指导.....	114
4.2.1 关键字 implements 的使用.....	71	6.3.1 搜索字符串.....	114
4.2.2 接口的类型.....	71	6.3.2 用 StringBuffer 处理字符串.....	115
4.2.3 实现接口时要注意的问题.....	73	6.4 习题.....	116
4.3 接口的多重继承.....	75	第 7 章 Java 集合	118
4.3.1 接口的多重继承.....	75	7.1 Java 集合框架简介.....	118
4.3.2 把 interfaces 内的数据成员 初始化.....	78	7.2 Java 集合接口.....	119
4.4 接口的嵌套.....	79	7.2.1 Collection 接口.....	119
4.5 上机指导.....	81	7.2.2 List 接口.....	121
4.6 习题.....	83	7.2.3 Set 接口.....	122
第 5 章 异常处理	85	7.2.4 Map 接口.....	122
5.1 异常的概述.....	85	7.3 集合的实现.....	123
5.1.1 异常及其分类.....	85	7.3.1 ArrayList.....	123
5.1.2 异常的局限性.....	87	7.3.2 LinkedList.....	126
5.2 异常的处理.....	88	7.3.3 HashSet.....	128
5.2.1 异常的处理机制.....	88	7.3.4 TreeSet.....	129
5.2.2 捕捉异常.....	89	7.3.5 HashMap.....	130
5.2.3 引发异常.....	89	7.3.6 TreeMap.....	131
5.3 异常的使用.....	92	7.4 通过迭代方法访问类集.....	133
5.3.1 try/catch/finally 的使用.....	92	7.5 上机指导.....	135
5.3.2 自定义异常.....	92	7.6 习题.....	142
5.4 上机指导.....	93	第 8 章 Java 图形用户界面	144
5.5 习题.....	95	8.1 Java 图形用户界面概述.....	144
第 6 章 字符串的使用	97	8.1.1 Java GUI 的组成.....	144
6.1 String 类.....	97	8.1.2 创建 GUI 应用程序.....	145
		8.2 GUI 程序设计常用技术.....	148

8.2.1 属性和方法	148	10.4 字符流	208
8.2.2 事件处理机制	150	10.4.1 CharArrayReader	210
8.3 GUI 常用组件的使用	151	10.4.2 CharArrayWriter	211
8.3.1 标签	152	10.4.3 InputStreamReader	213
8.3.2 按钮	152	10.4.4 OutputStreamWriter	214
8.3.3 下拉框	153	10.4.5 FileReader	216
8.3.4 文本框	154	10.4.6 FileWriter	217
8.4 布局管理的使用	155	10.4.7 StringReader	217
8.5 上机指导	157	10.4.8 StringWriter	218
8.6 习题	161	10.4.9 BufferedReader	219
第 9 章 applet	163	10.4.10 LineNumberReader	221
9.1 applet 概述	163	10.4.11 BufferedWriter	223
9.2 applet 实现	164	10.4.12 PrintWriter	224
9.2.1 applet 的生命周期	164	10.5 基本输入输出	226
9.2.2 简单 applet 实例	164	10.5.1 标准输入的读取	227
9.2.3 applet 标记	165	10.5.2 标准输出的写入	227
9.2.4 applet 部署	167	10.5.3 重定向标准输入输出	228
9.2.5 结合图形组件	169	10.6 随机访问文件	230
9.3 applet 通信	170	10.7 上机指导	233
9.3.1 同网页内 applet 间的通信	170	10.8 习题	243
9.3.2 applet 与浏览器的通信	172	第 11 章 多线程	245
9.4 上机指导	173	11.1 线程的概念	245
9.5 习题	177	11.2 线程的创建	246
第 10 章 I/O 系统	179	11.2.1 继承自类 Thread	246
10.1 文件的基本操作	179	11.2.2 实现接口 Runnable	248
10.1.1 获取文件名清单	179	11.2.3 两种创建线程方式的对比	249
10.1.2 获取文件属性信息	181	11.2.4 守护线程 Daemon	249
10.1.3 建立和删除文件	183	11.3 线程的同步	252
10.2 流概述	184	11.3.1 资源冲突	252
10.3 字节流	185	11.3.2 同步机制	253
10.3.1 ByteArrayInputStream	188	11.3.3 同步效率	256
10.3.2 ByteArrayOutputStream	190	11.4 线程间的通信	256
10.3.3 FileInputStream	192	11.5 线程的控制	261
10.3.4 FileOutputStream	194	11.5.1 线程的状态	261
10.3.5 FilterInputStream	196	11.5.2 线程的优先级	261
10.3.6 FilterOutputStream	200	11.5.3 线程的生命周期	262
10.3.7 SequenceInputStream	206	11.5.4 控制线程的状态	263

11.6 使用多线程应注意的问题.....	268	13.2 建立与数据库系统的连接.....	311
11.6.1 多线程的效率.....	268	13.2.1 建立与 Oracle 的连接.....	313
11.6.2 饿死.....	268	13.2.2 建立与 Microsoft SQL Server 的连接.....	315
11.6.3 死锁.....	269	13.3 访问数据库.....	316
11.6.4 不建议使用的方法.....	270	13.3.1 Statement.....	316
11.7 上机指导.....	270	13.3.2 ResultSet.....	319
11.8 习题.....	274	13.4 上机指导.....	324
第 12 章 网络编程	276	13.5 习题.....	335
12.1 网络基础知识.....	276	第 14 章 Java 编程规范与开发技巧	337
12.1.1 IP 地址与端口号.....	276	14.1 Java 编程规范.....	337
12.1.2 服务器与客户端.....	277	14.1.1 版权和版本的声明.....	337
12.1.3 TCP 与 UDP.....	278	14.1.2 方法文档标记.....	338
12.2 Java 网络编程基础.....	278	14.1.3 命名规范.....	338
12.2.1 URL 的表示.....	279	14.1.4 代码编写格式.....	339
12.2.2 IP 地址的表示.....	280	14.1.5 Java 源文件范例.....	339
12.2.3 IP 地址加端口号的表示.....	283	14.2 Java 中链表的实现方法.....	340
12.3 基于 TCP 的网络通信.....	284	14.3 Java 2 类库.....	342
12.3.1 TCP 连接.....	285	14.3.1 Java 基础类库.....	342
12.3.2 TCP 通信.....	287	14.3.2 其他功能.....	344
12.4 基于 UDP 的网络通信.....	292	14.3.3 商业功能.....	346
12.4.1 UDP 数据报文包.....	293	14.4 上机指导.....	347
12.4.2 UDP 通信.....	295	14.5 习题.....	350
12.5 上机指导.....	299	附录 参考答案	352
12.6 习题.....	304		
第 13 章 数据库应用	307		
13.1 JDBC 简介.....	307		

第 1 章 Java 开发基础

教学提示: 本章以实用为目的,通过简要讲述时下流行的面向对象程序设计思想和 Java 特有的虚拟机机制,使读者初步了解 Java 程序开发,并指导读者完成第一个 Java 程序。通过本章学习,读者将对 Java 程序开发有一个比较全面的了解。

教学目标: 领会面向对象编程思想;了解 Java 的虚拟机机制;能独立搭建 Java 开发环境。通过上机实践,能编写及运行简单程序。

1.1 面向对象程序设计思想

Java 是一种真正面向对象的具有众多功能的高级编程语言,了解并掌握面向对象程序设计思想是学习 Java 编程必备的内容。本节初步介绍面向对象思想和面向对象程序设计步骤。

1.1.1 面向对象的基本特征

程序设计思想是程序设计的灵魂,掌握好的设计思想,才能写出高效的程序。从计算机诞生到现在,程序设计语言的发展从最开始的机器语言、汇编语言到过程式语言、结构化高级语言,最后到支持面向对象技术的面向对象程序设计,使程序员可以脱离机器层次,在更抽象的层次上表达程序设计意图。

面向对象程序设计思想最重要的概念是对象,一个对象既包括数据,又包括操作这些数据的方法,对象中的数据并不能被对象的使用者直接访问,只能通过由对象提供的方法来访问和操作。这好比现实世界中的对象,比如一辆汽车,当需要增加车速时,是通过踩油门这个方法来实现的。

学习面向对象的程序设计首先要了解面向对象的基本概念。

1. 面向对象

面向对象是一种认识客观世界的观点,是从结构组织角度模拟客观世界的一种方法,人们在认识和理解现实世界的过程中,普遍运用以下三个构造法则:

- (1) 区分对象及其属性,如区分车和车的大小。
- (2) 区分整体对象及其组成部分,如区分车和车轮。
- (3) 不同对象类的形成及区分,如所有车的类和所有船的类。

2. 对象(Object)

对象是对一组信息及其操作的描述。如:一个窗口是一个对象,它包含了窗口的信息(如大小、颜色、位置等)及其操作(如打开、关闭等)。

3. 属性(Property)

即对象所包含的信息。如：窗口的高度。属性可以在设计对象时确定，也可以在程序运行时读取和修改。

4. 方法(Method)

即对象所具有的各种操作。如：窗口关闭。这种操作的过程对外是封闭的，即用户只能看到这一方法实施后的结果。这相当于事先已经设计好的各种过程，只需要调用就可以了，用户不必去关心这一过程是如何编写的，事实上，这个过程已经封装在对象中，用户也看不到。对象的这一特性，即是对象的封装性。

5. 事件(Event)

即对象在执行某一操作后激发并执行的一个或多个过程。这些过程对用户是透明的，用户可以为这个过程编写自己的程序代码，以完成特定的操作。如：窗口对象在执行打开过程时，就会激活一个 **Active** 事件(过程)，用户可以自己编写这一过程的代码，以便在打开这个窗口时完成一些自己所要求的任务，如打开一个数据库，对某个变量进行初始化等。

6. 类(Class)与实例(Instance)

类是具有共同属性、共同方法、共同事件的对象的集合。而一个具体的对象则是其对应类的一个实例。如果对某一个类的定义进行修改，如增加一些属性或修改一些方法，就得到一个新的类，而原先的类就是新类的父类。如：我们在窗口类的定义上，另外定义它还包含两个按钮，一个标题是“确定”，一个标题是“取消”，并且在分别按下这两个按钮后，执行 **OK** 与 **Cancel** 两个事件。这样就定义了一个新类，我们可以把它叫做“选择窗口”类，并保存它，这样就可以直接由“选择窗口”生成窗口实例，而不用每次都由“窗口”类产生，然后再添加按钮。

7. 继承(Inheritance)

任何一个子类都具有其父类所有的属性、方法和事件，这一特性叫做类的继承。如果父类的特性发生变化，其子类也相应改变。

继承机制的优点在于：

(1) 避免了由于系统内类对象封闭而造成数据和操作冗余的现象。每个子类都可以继承其父类的特性，包括状态与行为。同时子类可以有与父类不同的地方，即子类可根据自身特点新增或局部修改父类的行为而加以使用，甚至可以覆盖父类中的定义。利用继承，只要在原有类的基础上修改、增补、删减少量的数据和方法，就可以得到子类，然后生成大量不同的实例。

(2) 接口的一致性(Consistency of interface)，父类衍生子类的其他操作接口也传递给其子类。

(3) 可重用性，通过继承明显使软件开发速度加快，实现较高程度的共享，这是继承最重要的优势。

8. 消息(message)

面向对象的世界是通过对象与对象间彼此的相互合作来推动的, 对象间的这种相互合作需要一个机构协助进行, 这样的机构可以称为“消息传递”。消息传递过程中, 由发送消息的对象的使动操作产生输出结果, 消息传送至接受消息的对象, 引发接受消息的对象一系列的操作。

总之, 面向对象的整体概念可具体表示如下:

面向对象 = 数据抽象 + 数据抽象类型 + 继承性

目前有 Ada、C++、Eiffel、Java 等众多面向对象语言, 这些语言都有各自的特点, 但它们都具有面向对象的三个基本特征:

- 类: 对象系统的基本构筑块, 是一种数据结构, 它使用一组数据和方法来描述一组对象中公共的地方, 也就是说对象是类的一个具体实例。
- 继承: 继承提供了创建新类的一种方法, 一个新类可以通过对已有类进行修改或扩充来满足新类的要求。新类拥有已有类的所有域和方法, 而且还可以添加新的行为或修改现有的行为。
- 多态性: 多态指的是可以使用多种不同方式使用的对象, 多态性隐含着表明对象可以属于两个以上的分类, 提高了类代码的重用性和可维护性。

类、继承和多态性是面向对象语言所必需的基本特性。一个没有继承和多态性, 而只有类定义的语言, 只能称为基于类的语言。

1.1.2 面向对象程序设计步骤

了解面向对象程序开发的一般步骤对编写面向对象程序, 特别是初学者有很大帮助。在开始编程前首先要确定需要那些类和对象来支持程序。一般可以分以下几个步骤来进行程序设计。

1. 分析阶段

这一阶段主要采用面向对象技术进行需求分析。面向对象分析运用以下主要原则:

- (1) 构造和分解相结合的原则。构造是指由基本对象组装成复杂或活动对象的过程; 分解是对大粒度对象进行细化, 从而完成系统模型细化的过程。
- (2) 抽象和具体结合的原则。抽象是指强调事务本质属性而忽略非本质细节; 具体则是对必要的细节加以刻画的过程。面向对象方法中, 抽象包括数据抽象和过程抽象。数据抽象把一组数据及有关的操作封装起来, 过程抽象则定义了对象间的相互作用。
- (3) 封装的原则。封装是指对象的各种独立外部特性与内部实现相分离, 从而减少了程序间的相互依赖, 有助于提高程序的可重用性。
- (4) 继承的原则。继承是指可以直接获取父类已有的性质和特征。这样, 在系统开发中只须一次性说明各对象的共有属性和服务, 对子类的对象只须定义其特有的属性和方法。继承的目的也是为了提高程序的可重用性。

根据上述分析的主要法则得到的模型是具有一定层次关系的问题空间模型, 这个模型是相对有弹性, 且易修改、易扩充的。技术识别出问题域中的对象实体, 标识出对象间的关系, 然后通过对象的分析, 确定对象属性及方法, 利用属性变化规律完成对象及其关

系的有关描述, 并利用方法演变规律描述对象或其关系的处理。

2. 设计阶段

这一阶段主要利用面向对象技术进行概念设计。值得注意的是, 面向对象的设计与面向对象的分析使用了相同的方法, 这就使得从分析到设计的转变非常自然, 甚至难以区分。可以说, 从“面向对象分析”到“面向对象设计”是一个积累型的扩充模型的过程。这种扩充使得设计变得很简单, 它是从增加属性、服务开始的一种增量递进式的扩充。这一过程与结构化开发方法从数据流程图到结构图所发生的剧变截然不同。

一般而言, 在设计阶段就是将分析阶段的各层模型化的“问题空间”逐层扩展, 得到下个模型化的特定的“实现空间”。有时还要在设计阶段考虑硬件体系结构、软件体系结构, 并采用各种手段(如规范化)控制因扩充而引起的数据冗余。

3. 编码阶段

这一阶段主要是将面向对象设计中得到的模型利用程序设计实现。具体操作包括: 选择程序设计语言编程、调试、试运行等。前面两阶段得到的对象及其关系最终都是由程序语言、数据库等技术来实现的。

1.2 Java 的虚拟机机制

Java 的一个重要的特征平台无关性就是通过 Java 虚拟机屏蔽了与具体平台相关信息来实现的。一般高级语言程序如果需要搬到另一平台上使用, 至少必须重新编译才能保证能正常运行, 而 Java 语言只需编译成在 Java 虚拟机上运行的目标代码, 就可以直接在多个平台上运行。

Java 虚拟机(简称 JVM)其实是在实际的计算机上通过软件模拟来实现的虚拟机器。Java 虚拟机有自己想像中的硬件, 共包括 5 个部分: 指令集、寄存器、栈、堆和本地方法区。指令集用来执行基本的 CPU 运算; 寄存器用来存放地址信息, 比如执行指令的地址; 栈包括局部变量区域、运行环境区域和操作数区; 堆是 Java 运行时数据区, 程序初始化时所有对象都置于堆上; 本地方法区保存方法代码和符号表等。

Java 应用程序的开发周期包括编译、下载、解释和执行几个部分。Java 编译程序将 Java 源程序翻译为 JVM 可执行代码——字节码。这一编译过程同 C/C++ 的编译有些不同。当 C 编译器编译生成一个对象的代码时, 该代码是为在某一特定硬件平台运行而产生的。因此, 在编译过程中, 编译程序通过查表将所有对符号的引用转换为特定的内存偏移量, 以保证程序运行。Java 编译器却不将对变量和方法的引用编译为数值引用, 也不确定程序执行过程中的内存布局, 而是将这些符号引用信息保留在字节码中, 由解释器在运行过程中创立内存布局, 然后再通过查表来确定一个方法所在的地址。这样就有效保证了 Java 的可移植性和安全性。

运行 JVM 字节码的工作是由解释器来完成的。解释执行过程分三步进行: 代码的装入、代码的校验和代码的执行。装入代码的工作由“类装载器”(class loader)完成。类装载器负责装入运行一个程序需要的所有代码, 也包括程序代码中的类所继承的类和被其调用的类。

当类装载器装入一个类时,该类被放在自己的名字空间中。除了通过符号引用自己名字空间以外的类,类之间没有其他办法可以影响其他类。在本台计算机上的所有类都在同一地址空间内,而所有从外部引进的类,都有一个自己独立的名字空间。使得本地类通过共享相同的名字空间获得较高的运行效率,同时又保证它们与从外部引进的类不会相互影响。当装入了运行程序需要的所有类后,解释器便可确定整个可执行程序内存布局。解释器为符号引用同特定的地址空间建立对应关系及查询表。通过在这一阶段确定代码的内存布局,Java 很好地解决了由超类改变而使子类崩溃的问题,同时也防止了代码对地址的非法访问。

随后,被装入的代码由字节码校验器进行检查。校验器可发现操作数栈溢出,非法数据类型转化等多种错误。通过校验后,代码便开始执行了。

Java 字节码的执行有两种方式:

1. 即时编译方式

解释器先将字节码编译成机器码,然后再执行该机器码。

2. 解释执行方式

解释器通过每次解释并执行一小段代码来完成 Java 字节码程序的所有操作。

通常采用的是第二种方法。由于 JVM 规格描述具有足够的灵活性,这便将字节码翻译为机器代码的工作具有较高的效率。对于对运行速度要求较高的应用程序,解释器可将 Java 字节码即时编译为机器码,从而很好地保证了 Java 代码的可移植性和高性能。

1.3 搭建 Java 开发环境

1.3.1 安装 J2SDK

访问 <http://Java.sun.com>, 在出现的主页中显示 Java 2 平台的 3 个不同版本: Standard Edition (J2SE)、Enterprise Edition (J2EE)和 Micro Edition (J2ME)。本节使用的是 J2SE。目前的 JDK 最高版本是 1.5.0 Beta 1, 我们以 1.4.2 版本为例。

单击 J2SE 1.4.2 版本的链接,在出现的下载页面(<http://Java.sun.com/j2se/1.4.2/download.html>)中选择 Download J2SE SDK 一行,在随后出现的协议页面中单击 Accept 按钮,出现最后的下载窗口,单击 Windows Offline Installation, Multi-language (j2sdk-1_4_2_04-windows-i586-p.exe, 49.36 MB), 则提示保存目录并开始下载软件,该软件大小为 49.36MB。

J2SDK 安装过程如下:

1. 开始安装

双击下载的 J2SE 安装程序,在出现的初始安装界面中单击 Next 按钮,出现许可协议对话框,单击 Yes 按钮接受协议。