



普通高等教育“十一五”国家级规划教材

谭浩强 主编

高职高专计算机教学改革 **新体系** 规划教材

Java 2 高级程序设计

邵丽萍 张后扬 张驰 编著



清华大学出版社



普通高等教育“十一五”国家级规划教材

谭浩强 主编

高职高专计算机教学改革**新体系**规划教材

Java 2 高级程序设计

邵丽萍 张后扬 张驰 编著

清华大学出版社
北京

内 容 简 介

Java 是近年来十分流行的程序设计语言,又是一门通用的网络编程语言,在 Internet 上有着广泛的应用。本书是针对《Java 2 程序设计基础》(ISBN 978-7-302-17609-1)教材编写的配套教材,内容更深入,主要包括:输入/输出(I/O)流类、图形用户界面设计、Java 的多线程机制、使用多线程技术制作动画、多媒体处理、数据库访问技术、Java 程序综合应用实例、Java 服务网页——JSP 以及使用 JSP 访问数据库的技术等内容。

本书言简意赅、实例具体、图文并茂,从提出问题开始,针对提出的问题通过具体实例给出具体解题方案,在此基础上归纳指出知识点,应用性和操作性强。不仅适合学习过《Java 2 程序设计基础》教材的读者使用,也适合有一定 Java 程序语言基础的读者自学。本书可作为高等院校、高职高专或计算机培训班的教材使用。

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。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java 2 高级程序设计/邵丽萍,张后杨,张弛编著. —北京:清华大学出版社,2011.3
(高职高专计算机教学改革新体系规划教材)

ISBN 978-7-302-24690-9

I. ①J… II. ①邵… ②张… ③张… III. ①JAVA 语言—程序设计—高等学校:技术学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 013743 号

责任编辑:张 景 刘翰鹏

责任校对:袁 芳

责任印制:孟凡玉

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185×260 印 张:18 字 数:409 千字

版 次:2011 年 3 月第 1 版 印 次:2011 年 3 月第 1 次印刷

印 数:1~3000

定 价:29.00 元

产品编号:033660-01

丛书编委会

主 任 谭浩强

副主任 丁桂芝 李凤霞 焦金生

委 员	孔令德	王天华	王兴玲	王学卿
	刘 星	安淑芝	安志远	宋金珂
	宋文官	沈 洪	束传政	邵丽萍
	尚晓航	张 玲	张翰韬	林小茶
	赵丰年	高文胜	秦建中	崔武子
	谢 琛	薛淑斌	熊发涯	



近年来,我国高等职业教育迅猛发展,目前,高等职业院校已占全国高等学校半数以上,高职学生数已超过全国大学生的半数。高职教育已占了我国高等教育的“半壁江山”。发展高职教育,培养大量技术型和技能型人才,是国民经济发展的迫切需要,是高等教育大众化的要求,是促进社会就业的有效措施,也是国际上教育发展的趋势。

高等职业教育是我国高等教育的重要组成部分,高职教育的质量直接影响了全国高等教育的质量。办好高职教育,提高高职教育的质量已成为我国教育事业中的一件大事,已引起了全社会的关注。

为了更好地发展高职教育,首先应当建立起对高职教育的正确理念。

高职教育是不同于普通高等教育的一种教育类型。它的培养目标、教学理念、课程体系、教学内容和教学方法都与传统的本科教育有很大的不同。高职教育不是通才教育,而是按照职业的需要,进行有针对性培养的教育,是以就业为导向,以职业岗位要求为依据的教育。高职教育是直接面向市场、服务产业、促进就业的教育,是高等教育体系中与经济社会发展联系最密切的部分。

在高职教育中要牢固树立“人才职业化”的思想,要最大限度地满足职业的要求。衡量高职学生质量的标准,不是看学了多少理论知识,而是看会做什么,能否满足职业岗位要求。本科教育是以知识为本位,而高职教育是以能力为本位的。

强调以能力为本位,并不是不要学习理论知识,能力是以知识为支撑的,问题是学什么理论知识和怎样学习理论知识。有两种学习理论知识的模式:一种是“建筑”模式,即“金字塔”模式,先系统学习理论知识,打下宽厚的理论基础,以后再结合专业应用;另一种是“生物”模式,如同植物的根部、树干和树冠是同步生长的一样,随着应用的开展,结合应用学习必要的理论知识。对于高职教育来说,不应该采用“金字塔”模式,而应当采用“生物”模式。

可以比较一下以知识为本位的学科教育和以能力为本位的高职教育在教学各个方面的不同。知识本位着重学习一般科学技术知识;注重的是系统的理论知识,讲求的是理论的系统性和严密性;学习要求是“了解、理解、掌握”;构建课程体系时采用“建筑”模式;教学方法采用“提出概念—解释概念—举例说明”的传统三部曲;注重培养抽象思维能力。而能力本位着重学习工作过程知识;注重的是实际的工作能力,讲求的是应用的熟练性;学习

要求是“能干什么,达到什么熟练程度”;构建课程体系时采用“生物”模式;教学方法采用“提出问题—解决问题—归纳分析”的新三部曲;常使用形象思维方法。

近年来,国内教育界对高职教育从理论到实践开展了深入的研究,引进了发达国家职业教育的理念和行之有效的做法,许多高职院校从多年的实践中总结了成功的经验,有力地推动了我国的高职教育。再经过一段时期的研究与探索,会逐步形成具有中国特色的完善的高职教育体系。

全国高校计算机基础教育研究会于2007年7月发布了《中国高职院校计算机教育课程体系2007》(简称《CVC 2007》),系统阐述了高职教育的指导思想,深入分析了我国高职教育的现状和存在的问题,明确提出了构建高职计算机课程体系的方法,具体提供了各类专业进行计算机教育的课程体系参考方案,并深刻指出为了更好地开展高职计算机教育应当解决好的一些问题。《CVC 2007》是一个指导我国高职计算机教育的重要的指导性文件,建议从事高职计算机教育的教师认真学习。

《CVC 2007》提出了高职计算机教育的基本理念是:面向职业需要、强化实践环节、变革培养方式、采用多种模式、启发自主学习、培养创新精神、树立团队意识。这是完全正确的。

教材是培养目标和教学思想的具体体现。要实现高职的教学目标,必须有一批符合高职特点的教材。高职教材与传统的本科教育的教材有很大的不同,传统的教材是先理论后实际,先抽象后具体,先一般后个别,而高职教材则应是从实际到理论,从具体到抽象,从个别到一般。教材应当体现职业岗位的要求,紧密结合生产实际,着眼于培养应用计算机的实际能力。要引导学生多实践,通过“做”而不是通过“听”来学习。

评价高职教材的标准不是愈深愈好、愈全愈好,而是看它是否符合高职特点,是否有利于实现高职的培养目标。好的教材应当是“定位准确,内容先进,取舍合理,体系得当,风格优良”。

教材建设应当提倡百花齐放,推陈出新。我国高职院校为数众多,情况各异。地域不同、基础不同、条件不同、师资不同、要求不同,显然不能一刀切,用一个大纲、一种教材包打天下。应该针对不同的情况,组织编写出不同的教材,供各校选用。能有效提高教学质量的就是好教材。同时应当看到,高职计算机教育发展很快,新的经验层出不穷,需要加强交流,推陈出新。

从20世纪90年代开始,我们开始注意研究高职教育,并在1999年组织编写了一套“高职高专计算机教育系列教材”,由清华大学出版社出版,这是在国内最早出版的高职教材之一。在国内产生很大的影响,被许多高职院校采用为教材,有力地推动了蓬勃兴起的高职教育,后来该丛书扩展为“高等院校计算机应用技术规划教材”,除了高职院校采用之外,还被许多应用型本科院校使用。几年来已经累计发行近300万册,被教育部确定为“普通高等教育‘十一五’国家级规划教材”。

根据高职教育发展的新形势,我们于2005年开始策划,在原有基础上重新组织编写一套全新的高职教材——“高职高专计算机教学改革新体系规划教材”,经过两年的研讨和编写,于2007年正式由清华大学出版社出版。这套教材遵循高职教育的特点,不是根据学科的原则确定课程体系,而是根据实际应用的需要组织课程;书名不是按照学科的

角度来确定的,而是体现应用的特点;写法上不是从理论入手,而是从实际问题入手,提出问题、解决问题、归纳分析、循序渐进、深入浅出、易于学习、有利于培养应用能力。丛书的作者大都是多年从事高职院校计算机教育的教师,他们对高职教育有较深入的研究,对高职计算机教育有丰富的经验,所写的教材针对性强,适用性广,符合当前大多数高职院校的实际需要。这套教材经教育部审查,已列入“普通高等教育‘十一五’国家级规划教材”。

本套教材统一规划,分工编写,陆续出版,逐步完善。随着高职教育的发展将会不断更新,与时俱进。恳切希望广大师生在使用中发现本丛书不足之处,并不吝指正,以便我们及时修改完善,更好地满足高职教学的需要。

全国高校计算机基础教育研究会 会长 谭浩强
“高职高专计算机教学改革新体系规划教材”主编

前言

Java 是目前推广速度最快的程序设计语言,它采用面向对象编程技术,功能强大而又简单易学,深受广大程序设计人员的偏爱。Java 伴随着互联网问世,随着互联网的发展而成熟。Java 是精心设计的语言,它具有简单性、面向对象性、平台无关性、安全性和健壮性等诸多特点,内置了多线程和网络支持能力,可以说它是网络世界的通用语言。为了迎接信息时代的挑战,学习和掌握 Java 语言无疑会带来更多的机遇。

本书具有简单易学、理论和实例相结合的特点,可以使读者很容易地接受 Java 语言的概念和设计方法,很快地编写出合格的面向对象程序来解决一些简单的实际问题。一些抽象的很难理解的内容,如类、对象、继承、多态、异常、多线程等,在本书中都通过通俗易懂的方式进行了简化。使用本书学习,读者将会发现 Java 语言不难掌握。书中所有的程序都可上机运行,便于读者通过实际上机运行来体会 Java 的原理、Java 的功能与作用。

作为一本教材,本书在《Java 2 程序设计基础》教材的基础上对如何介绍 Java 高级程序设计的内容做了精心的规划,在各个章节内穿插使用了 Java 常用类库和方法,以复习巩固基础内容,在章节安排上尽可能做到由浅到深、由简到繁、循序渐进。在内容的编排上体现了新的计算机教学思想和方法,以“提出问题→解决问题的方法→归纳必要的结论和概念”的方式介绍 Java 编程思路,通过大量的实例和插图,使读者尽可能快地熟悉基本概念和掌握基本编程方法。

本书主要特色:

(1) 通俗易懂、图文并茂

本书都是通过具体的例子来介绍有关 Java 语言的概念、方法和技术,每章都有大量完整的例子,用来说明使用 Java 语言编程的基本步骤和基本方法,并有图片配合说明,通俗易懂。读者完全可以按书中介绍的方法完成每个例子,通过实例理解 Java 语言的基本思想和编程技巧。

(2) 内容完整、结构清晰

本书从介绍输入/输出(I/O)流类开始,重点介绍了如何创建图形用户界面,详细介绍了 Java 的多线程机制,系统介绍了数据库访问技术,在此基础上给出 Java 综合应用实例以巩固提升所学内容。在最后两章给出了 Java 在 Web 网络方面新的应用,介绍了 Java 服务网页——JSP 的技术,并给出了具体应用实例,使 Java 与数据库技术的联系更广泛、更简单、更实用。



(3) 循序渐进、逐步深入

本书对整个内容作了精心设计和安排,首先介绍 Java 语言图形用户界面的编程模式,再介绍 Java 服务网页——JSP 的技术,循序渐进,先易后难,逐步深入,通过综合应用实例巩固前面介绍的内容。

(4) 内容具体、便于操作

在使用本书学习时,可结合具体的实例,上机实践,按照书中介绍的例子,在短时间内使用 Java 语言设计图形用户界面系统或使用 JSP 创建动态网页与网站。

本书主要内容:

第 1 章是输入/输出(I/O)流类,介绍 Java 语言处理输入/输出(I/O)的方式,如何通过流处理与存储不同类型的数据,如何通过流将数据从一个地方送到另外一个地方。

第 2 章和第 3 章是图形用户界面,介绍如何利用 Java 的容器与组件进行图形用户界面设计,编写方便用户使用的窗口与界面,开发常见的软件系统。还介绍了使用 Swing 组件创建表格、树形菜单、选项卡面板等高级组件的方法。

第 4 章是 Java 的多线程机制,介绍如何通过 Java 的多线程机制更好地利用 CPU 的资源,使应用程序在相同的一段时间内同时做多件事情。并介绍了使用多线程技术创建美妙动画的具体方法。

第 5 章是数据库访问,帮助读者了解如何使用 Java 程序对数据库进行操作,并概要介绍 SQL 语句以及使用 Access 数据库保存数据的方法。

第 6 章是 Java 的一些综合应用实例,这些具体的应用实例综合体现了前面各章的编程思想和基本技术,对读者学习起到承上启下的作用,通过练习可掌握一些解决实际问题的 Java 编程技巧。

第 7 章是 Java 服务网页——JSP,介绍了 JSP 的基本语法、JSP 的几个重要内置对象的属性和方法,介绍了如何使用 JSP 开发 Web 应用开发程序,使这些程序能够与各种 Web 服务器、浏览器和开发工具共同工作。

第 8 章是通过 JSP 访问数据库,介绍了 JSP 如何与 Access 数据库进行连接,如何在客户端对服务器端的数据库内容进行搜索、查询、编辑、删除等操作。并给出了两个综合应用程序——密码表维护应用程序与客户留言系统应用程序。

本书有教师配套使用的电子课件,还有书中实例的源代码,需要的读者可登录清华大学出版社网站(<http://www.tup.com.cn>)免费下载。

本书由邵丽萍编写第 1、2、3 章,张后扬编写第 4、5 章,张驰编写第 6 章,郭彦涛编写第 7 章,杨丽编写第 8 章和附录。全书由邵丽萍统稿。

作 者

2009 年 12 月



第1章 输入/输出(I/O)流类	1
1.1 流的基本概念	2
1.1.1 什么是流	2
1.1.2 输入流与输出流	3
1.1.3 缓冲流	3
1.2 Java 的标准输入与输出	4
1.2.1 System 类的 in 与 out 变量	4
1.2.2 标准输入与输出实例	4
1.3 输入/输出流应用	6
1.3.1 面向字符的文件输入/输出流	6
1.3.2 随机存取文件流 RandomAccessFile	8
1.3.3 数据输入/输出流	10
1.3.4 对象输入/输出流	11
1.4 总结提高	14
1.5 思考与练习	16
1.5.1 思考题	16
1.5.2 上机练习	17
第2章 图形用户界面(上)	18
2.1 图形用户界面概述	19
2.2 容器组件	21
2.2.1 创建窗口	21
2.2.2 创建对话框	23
2.2.3 创建面板	24
2.3 组件	26
2.3.1 创建标签	27
2.3.2 创建按钮	28
2.3.3 创建文本框	30
2.3.4 创建文本区	32
2.3.5 创建单选按钮	35
2.3.6 创建复选框	37
2.3.7 创建下拉列表	39

2.4	总结提高	41
2.5	思考与练习	41
2.5.1	思考题	41
2.5.2	上机练习	41
第3章	图形用户界面(下)	43
3.1	菜单栏	44
3.1.1	创建菜单栏	44
3.1.2	创建多级菜单	47
3.1.3	创建快捷菜单	48
3.1.4	文件选择器	51
3.2	工具栏	53
3.3	树形菜单	55
3.4	选项卡面板	59
3.5	进度条	61
3.6	表格	65
3.7	总结提高	66
3.8	思考与练习	67
3.8.1	思考题	67
3.8.2	上机练习	67
第4章	Java 的多线程机制	69
4.1	多线程的概念	70
4.2	创建线程对象	71
4.2.1	通过继承 Thread 类创建线程对象	72
4.2.2	通过 Runnable 接口创建线程对象	74
4.2.3	线程的优先级	76
4.3	Java 的线程同步机制	78
4.3.1	生产消费模型	78
4.3.2	共用公司银行账户模型	81
4.4	用线程实现动画效果	83
4.4.1	动画的基本概念	83
4.4.2	跳伞的动画效果	84
4.4.3	球的动画效果	85
4.5	总结提高	88
4.5.1	线程类的方法	88
4.5.2	控制线程的状态	89
4.5.3	两种创建线程对象方法的比较	89
4.5.4	使用多线程应注意的问题	90
4.6	思考与练习	90

4.6.1 思考题	90
4.6.2 上机练习	91
第5章 数据库访问	92
5.1 结构化查询语言 SQL	93
5.1.1 SELECT 查询语句	93
5.1.2 SQL 的运算符与常用函数	95
5.1.3 INSERT INTO 添加语句	97
5.1.4 UPDATE 更新语句	98
5.1.5 DELETE 删除语句	99
5.1.6 CREATE TABLE 创建表语句	99
5.1.7 DROP TABLE 删除表语句	100
5.2 Access 数据库	100
5.2.1 创建数据库文件	101
5.2.2 通过 ODBC 创建连接数据库的数据源 DSN	107
5.3 使用 JDBC 访问数据库	111
5.3.1 通过 JDBC 创建 Java 程序输出数据库的数据	111
5.3.2 通过 JDBC 创建 Java 程序添加并删除数据库中的数据	114
5.3.3 通过 JDBC 创建 Java 程序更新数据库中的数据	116
5.4 通过窗口界面访问数据库	117
5.4.1 创建连接数据源的类	118
5.4.2 创建数据处理的类	120
5.4.3 创建用户访问数据库的窗口界面类	124
5.5 总结提高	129
5.6 思考与练习	131
5.6.1 思考题	131
5.6.2 上机练习	131
第6章 Java 程序设计综合应用实例	133
6.1 数值变换运算	134
6.2 幻灯机效果	135
6.3 利用滑块改变背景颜色	137
6.4 Applet 与 Application 合并运行的程序	139
6.5 对象的克隆	141
6.6 正弦曲线	142
6.7 四叶玫瑰曲线	144
6.8 在画布上手工画图	146
6.9 电闪雷鸣的动画	148
6.10 文字动画	150
6.11 控制移动的文字	152

6.12	图形钟	154
6.13	水中倒影	157
6.14	网上购物结算窗口	159
6.15	交通信号灯的模糊控制	162
6.16	简单学生信息管理系统	165
第 7 章	Java 服务网页——JSP	167
7.1	HTML 与 JSP	168
7.1.1	HTML	168
7.1.2	JSP	169
7.2	构建 JSP 的运行环境	169
7.2.1	下载与安装 SDK	169
7.2.2	下载与安装 Tomcat	173
7.2.3	创建 JSP 用户目录	179
7.2.4	编写与运行 JSP 程序	179
7.3	JSP 的基本语句	181
7.3.1	JSP 指令语句	181
7.3.2	JSP 注释语句	182
7.3.3	JSP 声明语句	183
7.3.4	JSP 表达式语句	183
7.3.5	Java 代码片段	184
7.4	JSP 常用的内置对象	185
7.4.1	获取输入信息的 request 对象	185
7.4.2	发送响应信息的 response 对象	188
7.4.3	输出结果信息的 out 对象	190
7.4.4	保存用户信息的 session 对象	192
7.4.5	保存公共信息的 application 对象	195
7.5	JSP 常用的动作标记	198
7.5.1	jsp:include 动作标记	198
7.5.2	jsp:forward 动作标记	200
7.5.3	jsp:plugin 动作标记	201
7.5.4	jsp:JavaBean 动作标记	203
7.5.5	jsp:setProperty 与 jsp:getProperty 动作标记	207
7.6	创建虚拟目录	207
7.6.1	根目录、物理目录与虚拟目录	207
7.6.2	通过 Tomcat 管理器创建虚拟目录	208
7.6.3	通过 server.xml 文件创建虚拟目录	210
7.7	总结提高	212
7.8	思考与练习	213

7.8.1 思考题.....	213
7.8.2 上机练习.....	213
第8章 通过 JSP 访问数据库	215
8.1 在服务器端通过 JSP 访问数据库	216
8.1.1 通过 JSP 页面显示数据库中的数据	216
8.1.2 通过 JSP 在数据库中添加或删除数据	218
8.2 在客户端通过 JSP 访问数据库	222
8.2.1 创建两个重复使用的公用文件.....	222
8.2.2 在客户端向数据库添加数据.....	223
8.2.3 在客户端输入查询条件并显示查询结果.....	225
8.3 JSP 综合应用实例	226
8.3.1 密码表维护应用程序.....	226
8.3.2 创建用户留言系统应用程序.....	232
8.4 总结提高	249
8.4.1 连接 Access 数据库的不同方式	249
8.4.2 在 JSP 文档中插入其他文件的方式	250
8.5 思考与练习	251
8.5.1 思考题.....	251
8.5.2 上机练习.....	251
附录 A HTML 常用标记	253
附录 B JavaScript 常用内置对象	257
附录 C JavaScript 常用内置函数	268
参考文献	271

输入/输出 (I/O) 流类

流是 Java 语言中处理输入/输出(I/O)的方式,通过流可以更加方便快捷地处理与存储不同类型的数据。由于文件是计算机用来保存大量数据域信息的地方,文件保存的数据对应不同的数据类型,因此,Java 提供了不同的流类来读取文件的内容或是向文件写入数据。通过流,可以将数据从一个地方送到另外一个地方。

学习目标

通过本章的学习,能够掌握:

- ✓ 流的作用与使用流的基本步骤
- ✓ 标准输入与输出的方法
- ✓ 文件流的使用方法
- ✓ 缓冲流的使用方法
- ✓ 数据流的使用方法
- ✓ 对象流的使用方法

1.1 流的基本概念

本节任务是了解流的概念、特点与用途。

1.1.1 什么是流

1. 问题的提出

数据是指一组有顺序的、有起点和终点的字节集合,数据可以通过不同的格式而存在,如字符串、图像、声音或对象等。使用数据时,提供数据的地方称为数据的发送者,使用数据的地方称为数据的接收者,它们可以是一个程序、一个文件、磁盘、内存、另一个程序或是网络。

在使用程序进行某种计算时,可能需要使用保存在其他位置的文件中的一组数据,如何才能得到这个文件中的数据呢?在使用程序进行某种计算时,也可能需要把计算的结果保存在其他位置的某个文件中,怎么实现这种异地数据的传递任务呢?

Java 使用流解决数据发送者(数据源)与数据接收者(目的地)之间数据的传递任务。

2. 流(Stream)

流是一个很形象的概念,当程序需要读取数据的时候,可以打开一个通向数据源的流,这个数据源可以是文件、内存或是网络连接。类似的,当程序需要写入数据的时候,可以打开一个通向目的地的流。这时可以想象数据好像在流(Stream)中“流”动一样,如图 1.1 所示。

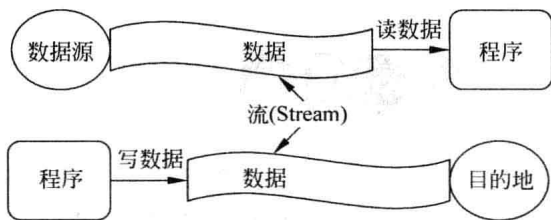


图 1.1 作为通道的流

通过图 1.1 可以看出,流是传递数据的载体,是数据所经历的路径。通过流,程序可以把数据从一个地方带到另一个地方,流可以视为程序在数据发送者和数据接收者之间建立的数据通道。

流的处理过程分为:打开流、读取或写入流、关闭流。如同水龙头一样,需要水时,打开水龙头流水,水流够了,再关闭水龙头。

流的设计使 Java 程序在处理不同 I/O 设备时非常方便。Java 程序不直接操纵 I/O 设备,而是在程序和设备之间加入了一个介质流。采用流的目的就是使程序的输入输出

操作独立于具体设备,程序一旦建立了流,就可以不用理会起点或终点是何种设备,而只关心使用的流。

1.1.2 输入流与输出流

1. 输入流与输出流的分工

建立流,实际上就是建立了一个数据传输通道,将数据起点和终点连接起来。例如,可以在程序和文件之间建立一个流。如果要从文件中读数据,则文件是起点,程序是终点;如果要将数据写入文件,则刚好相反。

可见,流可分为输入流和输出流。输入流指将数据从数据源传递给程序,输出流指将数据从程序送到数据接收者,如内存或文件。输入流只能读,不能写;而输出流只能写,不能读。输入流可从键盘或文件获得数据,输出流可向显示器屏幕、打印机或文件中传输数据。

2. java.io 包中的输入流与输出流类

Java 在 java.io 包中专门声明了用于读写操作的面向字节的输入流类 `InputStream` (抽象类)与输出流类 `OutputStream` (抽象类),还有面向字符的输入流类 `Reader` (抽象类)与输出流类 `Writer` (抽象类)。

java.io 包中有许多输入流类与输出流类,所有面向字节的输入流类都是输入类 `InputStream` 的子类,所有面向字节的输出流类都是输出类 `OutputStream` 的子类。例如,`InputStream` 包含文件输入流类 `FileInputStream`,`OutputStream` 包含对象输出流类 `ObjectOutputStream`。

所有面向字符的输入流类都是输入类 `Reader` 的子类,所有面向字符的输出流类都是输出类 `Writer` 的子类。例如,`Reader` 包含文件输入流类 `FileReader`,`Writer` 包含对象文件输出流类 `FileWriter`。

面向字节的流在处理二进制文件时使用。面向字符的流在处理用 ASCII 字符集或 Unicode 表示的文本,如在处理纯文本文件、HTML 文件和 Java 源代码文件时使用。

只要在程序开头加上语句: `import java.io.*`,即可使用其中输入流类与输出流类的各种 I/O 方法。

1.1.3 缓冲流

对流的每次操作都是以字节为单位进行,即可以向输入流或输出流中读取或写入一个字节,显然这样的数据传输效率很低。为了提高数据传输效率,Java 还设计了缓冲流 (buffered stream),它可以为一个流配备一个缓冲区 (buffer),一个缓冲区就是专门用于存储数据的一块内存。例如:

当向一个缓冲流写入数据时,系统将数据发送到缓冲区,而不是直接发送到外部设备,缓冲区自动记录数据;当缓冲区满时,系统将数据全部发送到相应设备。