

新世纪计算机及相关专业系列教材

Java 程序设计及应用开发教程

虞益诚 主编

缪行外 薛万奉 李晓玲 范伟继 副主编

科学出版社

北京

内 容 简 介

Java 语言是目前最为流行和最具魅力的面向对象程序设计语言。本书系统地介绍了 Java 语言的基本概念与方法、设计技术与应用、开发过程与实施、课程设计与实验。本教程兼顾基础理论、操作技能与应用开发,融理性与感性知识为一体,是作者多年从事该分支教学与研究的心得。为便于学习,每章后均附有小结与习题,供读者领悟与自我测试之用。教材中还有融汇教材主体内容的应用开发、课程设计和上机实验环节。本书提供了与教材配套的教学课件、经过精心设计与调试的编程例题以及实例的源代码,旨在使教师们能从繁杂的课件与代码编写中聊以释负。本书尤其适合于教学与应用开发。

本书既可作为高等院校学习 Java 程序设计语言的教程,也可作为从事该分支研究的研究生与广大工程技术人员的重要参考书。

图书在版编目(CIP)数据

Java 程序设计及应用开发教程/虞益诚主编. —北京:科学出版社, 2007

(新世纪计算机及相关专业系列教材)

ISBN 978-7-03-019234-9

I. J… II. 虞… III. JAVA 语言-程序设计-高等学校-教材
IV. TP312

中国版本图书馆 CIP 数据核字 (2007) 第 092508 号

责任编辑:陈晓萍/责任校对:耿 耘

责任印制:吕春珉/封面设计:王 浩

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2007年7月第 一 版 开本:787×1092 1/16

2007年7月第一次印刷 印张:29 1/2

印数:1—3 000 字数:696 000

定价:39.00 元

(如有印装质量问题,我社负责调换〈环伟〉)

销售部电话 010-62136131 编辑部电话 010-62138978-8003

新世纪计算机及相关专业系列教材

编委会

顾问编委

施伯乐教授	复旦大学
何积丰教授	华东师范大学

主 任

白英彩教授	上海交通大学
-------	--------

秘 书 长

张昆藏教授	青岛大学
-------	------

编 委

刘 璟教授	南开大学
宋方敏教授	南京大学
何炎祥教授	武汉大学
余雪丽教授	太原理工大学
阮家栋教授	上海工程技术大学
顾训穰教授	上海大学
徐汀荣教授	苏州大学
曾 明教授	西安交通大学
曹元大教授	北京理工大学
曹文君教授	复旦大学
陶树平教授	同济大学
缪淮扣教授	上海大学
谢康林教授	上海交通大学

序 一

电子计算机（又称电脑）自 20 世纪中叶诞生以来，经历了一个只能由极少数人参与研制、应用的神秘阶段，长达 30 多年，直到 80 年代才逐渐进入普遍应用与普适教育阶段。这时的计算机应用已渗透到国民经济的各部门，进而又用于人们的工作、生活和娱乐等方面，这是计算机与通信技术、自动化技术和微电子技术密切结合的结果，并以网络形态迅速普及发展开来。与此同时，掀起了以信息技术（IT）为基础的信息经济时代。大致说来，IT 经历了三次浪潮，即以 1981 年个人计算机（PC）的推出乃至广泛应用阶段为第一次浪潮；20 世纪 90 年代初期，Internet 被推向社会进入商用阶段，从而掀起了第二次 IT 浪潮；本世纪之初又掀起了以发展存储为中心的存储区域网（SAN）和网络附加存储系统（NAS），用以存储和保护海量昂贵数据为目标的第三次 IT 浪潮。这三次浪潮期间也恰逢我国改革开放时期，计算机是 IT 的重要子集，全国的计算机教育也得到蓬勃的发展，止于 1995 年，全国已有 137 所高校设有计算机系或专业，而到 2005 年，我国已发展到有 505 所高校具有计算机本科以上的计算机科学技术专业，其中部分高校还招收相应的硕士、博士研究生，说明最近的 10 年我国 IT 技术处于加速发展时期。针对这种情况，我国以往在计算机教育方面多沿用美国的 IEEE/ACM 组织的 CC1991/CC2001 计算机学科的课程计划，恐难以适应目前的发展状况。

21 世纪以来，更多的专家、学者认为，计算机科学技术的教学任务目标，以划分为培养“研究型”、“工程型”和“应用型”三个方面的人才为宜。这三方面不是上、中、下的层次关系，而是处于同一层面。只是各类型人才在培养数量上不同而已。“研究型”人才可能占极少数，“应用型”人才是大多数，而培养“应用型”人才的任务还可以根据各高校的自身情况和各行业的背景而细化。这种定位很重要，既涉及各高校的教学设施、教学队伍和生源情况，又涉及当下的人才市场需求情况。我们在科学出版社支持下成立的“新世纪计算机及相关专业系列教材”编委会，其宗旨就是以满足新形势下的计算机教育为己任。

我们认为，针对“研究型”人才的计算机教育，其课程设置应以程序设计基础、离散结构、数据结构、算法分析、计算机组成基础、计算机体系结构、操作系统、数据库原理、编译原理、软件工程、计算机网络等为核心课程。而培养“应用型”人才的计算机专业的课程设置，只选上述大部分的核心课程即可，代之以更靠近行业应用和更侧重实训性的课程，例如嵌入式系统、计算机网络通信、微机原理与接口、数字逻辑、人工智能、计算机图形学和信息系统工程等，也可以为面向某行业应用背景，开设相应的组合式的课程，其中包括一些对口的选修课和实训性课程。培养“工程型”人才则介于上述两者之间，在此不再赘述。

由于信息技术发展迅猛，其教育内容也日益增多，为使学生打好基础，并培养其实践能力以适应市场需要，精心设计课程设置计划是各院校的首要任务之一。本编委会以组织业内的知名教授、专家和学者精心编审出一批相应的教材为己任。

本套教材的出版工作于 2000 年正式启动，由上海交通大学左孝凌教授负责协调、组织工作。几年来，一批优秀教材陆续出版，在读者中反映良好。2004 年 8 月，左教授因病去世。编委会于 2005 年 1 月聘上海交通大学白英彩教授接任编委会主任，并增补青岛大学张昆藏教授为编委会秘书长，协助继续完善这套教材。我们诚恳希望各兄弟院校的教师能够参与、协助、承担和分享编委会的任务。

编委会
(白英彩执笔)
2006 年 5 月

序 二

20年来, 计算机学科的发展日新月异, 促使现代科学在各个领域突飞猛进。目前, 计算机科学技术已应用在实时控制、信息处理、通信传输、企事业管理等领域, 成为人们工作、学习、生活必不可少的工具。计算机技术的发展瞬息万变, 具有以下三方面特点:

(一) 传统的工、理、文、医、商、农在计算机的应用方面都有着各自专业的需要, 例如, 经济、艺术、法律、管理、医学等各种学科都需要依赖于计算机技术的应用。除了各自领域的专业实践外, 应用计算机已是各个专业提高效率、发挥潜能、促进发展的必不可少的手段。因此现在很难用传统的工、理、文、医、商、农等去界定学科的分类。

(二) 计算机网络改变了计算机通信的时空距离。计算机应用的发展是与计算机网络的发展紧密相连的。从最初的局域网 (LAN) 到广域网 (WAN), 以至用一种新的方法将 LAN 和 WAN 互联起来, 即成为网际网 (Internet)。这种网际网的实验原型 Internet 通常缩写为 Internet。计算机网络将计算机互联起来, 从而使计算机之间可以交换信息, 而且这种信息交换可以在几分钟内就影响到世界各地。计算机网络的发展, 带动了计算机学科在很多领域的拓展。

(三) 现代计算机学科向综合性发展。计算技术发展伊始, 每种学科均以软硬件分类, 泾渭分明。但自网络发展以来, Internet 软件中的两部分变得特别重要和特别具有开创性, 即网际协议 (Internet Protocol, 简称 IP 协议) 和传输控制协议 (Transmission Control Protocol, 简称 TCP 协议)。这些协议是必不可少的软件系统。但是在网络系统中, 网络的互联必须依靠路由器、服务器、接口插座、调制解调器等硬件设施, 所以计算机网络很难归结为软件或硬件的单一体系。

随着计算机技术的发展, 计算机与通信、视频、声音等密不可分; 随着多媒体的发展和应用, 计算机科学已经愈来愈成为与数字传输、视频、声、光、电等综合的学科。

尽管计算机技术的发展如此神速、新异, 但像一切新学科的发展一样, 计算机教育水平仍滞后于计算机技术的发展。为了适应计算机教学改革的需要, 我们国内部分重点院校的教授、学者, 在科学出版社的积极鼓励和支持下, 成立了新世纪计算机及相关专业系列教材编委会。自 2000 年 10 月以来, 我们群策群力, 多次探讨了当前教育与技术进展之间的差距, 并且仔细研讨了美国 ACM/IEEE-CS 公布的 *Computing Curricula 2001* 的优点与不足, 结合我国计算机教育的实际情况, 提出了编著一套适用于计算机本科专业的励精图治的教材计划。这套教材的选题、定位乃至作者的遴选都得到了国内很多著名教授和学者的认同, 并且有很多选题争取到了由一些著名教授亲自参与编写。这套教材立意着重基础, 反映导向, 注重实践。

因此我们在基础课目方面, 首先列选了数据库原理、操作系统、编译程序原理、智能基础等基础教程。这些基础课教材都由一些国内著名学者执笔, 论述内容既注意打

好扎实基础，又注意要反映最新导向，高屋建瓴，使读者迅速接近最新领域。

同时，为了反映导向，我们抓住网络课程作为计算机专业学生的应用基础，编写了一本实用性极强的《计算机网络教程》。这本教材的编著思想是以基础-理论-应用为主线，通信是基础，协议是核心，互联是重点，应用是目标。

其次，为了拓展学生的网络应用本领，我们还安排了电子商务、多媒体应用以及 Web 数据库技术三门应用课程。电子商务和多媒体应用是计算机应用中最为热门的课程，也是拓展性极广的计算机应用领域，应用前景极为广阔。

Web 数据库技术是一种随着互联网技术发展起来的应用技术。它涉及网络、HTTP 协议、Script 语言、动态网页开发平台、远程数据访问技术等各种网络应用技术。目前国内外还无适合教材，因此，编写适用的 Web 数据库技术教材，可以说是填补了应用领域的一个空白。

在研究美国公布的“计算 2001-CS 教程”中，我们仔细探讨了数据结构这一课程的变化。在“计算 1991 教程”中，数据结构内容明确放在算法与数据结构之中，而“2001-CS 教程”却无数据结构的课程名称，代之以程序设计基础（Programming Fundamentals）。文件中提到了基本数据结构和抽象数据类型以及面向对象的程序设计等内容。从这里可以看出，数据结构是以程序设计基础作为研究对象的。另外该教程把算法与复杂性作为一个单独课程列出，这一方面说明算法是一种问题求解的策略；另一方面也说明基本算法及复杂性的讨论对于程序设计是多么重要。

为此在这套丛书中我们安排了一个软件课程系列，即开设从语言、数据结构、算法到软件工程的课程。首先我们从面向对象的 C++ 语言入手，进一步讲解语言学概论。主要内容是分析语法结构，掌握语言构成规律，读懂语言文本。任何计算机语言均可触类旁通，这种从结构规律来学会应用的方法，就是以不变应万变，因为从根本上说，尽管计算机语言千变万化，但万变不离其宗。在掌握语言的基础上，我们组编了数据结构，或者说是研究程序设计基础。然后是学习基本算法，也就是为了程序设计需要而进行问题求解，即进行常用算法讨论。为了使开发软件遵循工程管理方法，软件工程将是计算机专业学生学习规范软件开发的必不可少的训练课程。

我们筹组这套教材时，希望每本教材都有创意，能引起共鸣，能被关注、采纳、推广。但是我们也注意到，由于各个学校情况不同，各人观点不同，理解角度也有所不同，所以对教材的选用和编著，不易一致认同。不过我们希望这套教材能够反映当前学校动向，在促进学以致用等方面有所促进、有所推动，更希望兄弟院校的教师、学者能够积极使用，参与讨论，以使本套教材能够不断修改，日臻完善。

最后，我要感谢科学出版社的领导对本套教材的列选、报审、出版所给予的鼓励和支持。

左孝凌

前 言

Java 语言以其面向对象性、可重用性、平台无关性、多线程、安全可靠等特性成为网络程序设计语言中的佼佼者脱颖而出。它是一种功能强大的程序设计语言，是当今发展最为迅速、应用最为广泛的跨平台软件，强大的网络功能使其升华成最为流行和最具魅力的面向对象程序设计语言。Java 是一种完全面向对象的程序设计语言，它的高可移植性、健壮性使得“一次编写，到处运行”梦想成真，进而使得 Java 语言备受众多程序设计人员的青睐与问津，在 Internet 上更获得了广阔的用武之地。

Java 语言具体的开发与应用在经历了多年的演进与洗礼后已有了长足的发展，而作为一门迅速崛起的高等院校语言类课程，却仍处在不断地建设和完善中，亟需真正能表述其内涵、丰富其外延、展示其方法、体现其应用、发挥其技能的教程来不断充实和提升之。本着普通高等院校教学大纲和同时提升学生（读者）开发技能的理念，作者根据多年从事该分支教学与研究的心得，编写完成了本教程。本教程注重理论的严谨性和完整性、技能的实用性与创新性，力求使学生（读者）在掌握 Java 编程语言的同时，获得程序开发的基本思想、运作方法与实施技能，以培养学生（读者）独立开发较为复杂系统的能力。

本教程归结为三大篇（基础知识篇、程序设计篇、应用开发篇），共 16 章及附录部分。

基础知识篇涵盖了第 1 章～第 5 章，扼要地介绍了 Java 运作机制与开发过程、Java 编辑工具 JCreator Pro 3.5、Java 开发运行环境、数据类型、表达式与运算符、流程控制、数组与字符串等内容，然后对 Java 面向对象程序设计的基本概念：类、对象、接口、继承和多态等进行了深入浅出地讲解。

程序设计篇涵盖了第 6 章～第 11 章，翔实地叙述了异常处理机制和方法、try-catch-finally 语句与 throw/throws 异常抛出、输入/输出流与文件处理、AWT 图形化界面组件、事件处理机制、布局设计、菜单组件与 Swing 组件、多媒体、多线程和 Applet 应用程序等。

应用开发篇涵盖了第 12 章～第 16 章及附录实验环节，具体地介绍了 JDBC 和基于 JDBC 的数据库访问、URL 网络编程与 Socket 网络通信、Servlet 程序设计、JSP 程序设计、Java 应用开发与课程设计实例，以及本教程具体的实验环节。

本教程兼顾基础理论、操作技能与具体应用，融理性与感性知识为一体，思绪明晰、论述严谨、循序渐进、案例丰富、实验翔实，涉及的内容既有一定的深度，也兼顾覆盖面与前沿性，是作者多年从事该分支教学与实践研究的心得。书中配备了大量的操作示例，并辅以已通过运行的屏幕画面，可使读者有身临其境之感，易于阅读和理解。为便于学习、梳理思绪，每章后均附有小结和习题，供读者领悟与自我测试之用，附录中也附有部分习题答案与融汇教材主体内容的上机实验环节。

本书提供了与教材配套的教学课件、经过精心设计与调试的编程例题以及实例的源

代码，旨在使读者（尤其是教师们）能从繁杂的课件与代码编写中聊以释负，具体可从科学出版社网站（www.sciencep.com）的下载区中下载。

本书的编写人员由富有长期从事该领域教学研究与应用开发的博士、专家、教授等组成，并在酌定架构时得到了上海交通大学博导白英彩教授、上海工程技术大学阮家栋教授的关注与帮助，阮家栋教授尚参与了主要文体的审阅，在此一并表示由衷的谢意！

本书由虞益诚总体架构。第1章、第3章、第4章、第8章、第9章、第14章、第16章由虞益诚编写；第2章由范伟绩编写；第5章、第7章、第10章由李晓玲编写；第6章由孟健编写；第11章由薛万奉编写；第12章、第15章由缪行外编写；第13章由范伟继、池涛与薛万奉编写；附录由虞益诚与陶然编写。全书由虞益诚统稿校阅。

在本书的编写过程中，作者阅览、借鉴了大量参考资料，在此谨向诸多学者、同仁表示由衷的敬意与感谢。本书中所使用到的某些人名、电话号码、通信地址等均为解析所用的虚托，如有雷同，实属巧合，烦请见谅。

由于作者水平有限及本书编期仓促、涵盖面广，书中难免有纰漏和考虑不周之处，敬请广大同仁与读者不吝赐教，拨冗指正。E-mail: yuyich@sit.edu.cn。

虞益诚

2007年3月

目 录

基础知识篇

1 Java 概述	1
1.1 Java 发展与特点	1
1.1.1 Java 起源与发展	2
1.1.2 Java 的技术现状	3
1.1.3 Java 语言的特点	3
1.2 Java 程序的分类	6
1.2.1 Java Application	6
1.2.2 Java Applet	6
1.2.3 Java Servlet	7
1.2.4 JavaBean	7
1.2.5 JSP	8
1.3 Java 运作机制	8
1.3.1 Java 虚拟机	8
1.3.2 JVM 执行过程	9
1.4 Java 运行平台	9
1.4.1 JDK 1.5 系统安装	10
1.4.2 JDK 环境变量设置	12
1.5 Java 程序开发	14
1.5.1 Java 程序开发步骤	14
1.5.2 Java Application 的开发	14
1.5.3 Java Applet 开发	17
1.6 Java 编辑工具简介	18
1.6.1 JCreator	18
1.6.2 JBuilder	19
1.6.3 NetBeans	19
1.6.4 Eclipse	19
1.6.5 Visual Age for Java	19
1.6.6 Java WorkShop	19
1.6.7 Visual Café for Java	20
1.6.8 FreeJava	20
1.7 JCreator 3.5 编辑工具	20
1.7.1 JCreator 简述	20
1.7.2 JCreator 安装	21
1.7.3 JCreator 设置	21

1.7.4 JCreator 使用	23
1.8 Java 与 C/C++ 的比较	27
1.9 本章小结	28
思考与练习	29
2 Java 语言基础	31
2.1 符号	31
2.1.1 标识符	31
2.1.2 关键字	31
2.1.3 分隔符	32
2.1.4 注释语句	33
2.1.5 编码规范	34
2.2 基本数据类型	34
2.2.1 整数类型	35
2.2.2 浮点型	36
2.2.3 字符型	37
2.2.4 布尔型	38
2.3 常量与变量	38
2.3.1 常量	38
2.3.2 变量	39
2.4 表达式	41
2.4.1 语句	41
2.4.2 表达式	41
2.5 运算符	42
2.5.1 算术运算符	42
2.5.2 比较运算符	45
2.5.3 逻辑运算符	46
2.5.4 位运算符	47
2.5.5 条件运算符	49
2.5.6 赋值运算符	49
2.5.7 其他运算符	50
2.5.8 运算符优先级	50
2.5.9 运算符的应用	51
2.6 类型转换	52
2.6.1 自动类型转换	52
2.6.2 强制类型转换	53
2.7 本章小结	54
思考与练习	55
3 流程控制	57
3.1 语句控制结构	57
3.1.1 语句类型	57
3.1.2 语句控制结构	58

3.2	分支语句	59
3.2.1	if 条件语句	59
3.2.2	switch 多分支语句	63
3.3	循环语句	65
3.3.1	while 语句	65
3.3.2	do-while 语句	66
3.3.3	for 语句	67
3.3.4	循环嵌套	71
3.4	跳转语句	73
3.4.1	break 语句	73
3.4.2	continue 语句	74
3.4.3	return 语句	75
3.5	数组	75
3.5.1	一维数组	75
3.5.2	多维数组	78
3.5.3	数组应用	81
3.6	字符串	83
3.6.1	创建字符串变量	84
3.6.2	字符串操作	85
3.7	本章小结	88
	思考与练习	88
4	面向对象程序设计基础	91
4.1	面向对象基础	91
4.1.1	面向对象概述	91
4.1.2	面向对象软件的开发过程	92
4.1.3	面向对象程序设计方法的优点	93
4.2	类	95
4.2.1	类的概念	95
4.2.2	类的定义	97
4.3	方法	99
4.3.1	方法声明	99
4.3.2	方法调用	102
4.3.3	参数传递	104
4.3.4	递归	105
4.4	对象与构造方法	106
4.4.1	对象的创建	106
4.4.2	对象的引用	107
4.4.3	构造方法	110
4.5	成员变量	111
4.5.1	成员变量的定义	111
4.5.2	成员变量修饰符	112

4.6	复合数据类型数组	117
4.7	本章小结	118
	思考与练习	118
5	继承与多态	121
5.1	类的封装	121
5.1.1	类的封装	121
5.1.2	访问权限	122
5.2	类的继承	125
5.2.1	继承机制	125
5.2.2	创建子类	125
5.2.3	this 与 super	128
5.3	多态机制	131
5.3.1	方法重载	131
5.3.2	构造方法重载	133
5.3.3	方法覆盖	134
5.3.4	终止覆盖	136
5.4	接口	137
5.4.1	接口及特点	138
5.4.2	接口的实现	139
5.5	包	140
5.5.1	包的定义	141
5.5.2	包的引用	142
5.5.3	常用包介绍	143
5.6	内部类*	144
5.7	本章小结	147
	思考与练习	147

程序设计篇

6	异常处理	150
6.1	异常处理机制	150
6.1.1	异常层次结构	150
6.1.2	异常处理机制	152
6.1.3	异常的分类	152
6.1.4	内置异常类	152
6.2	异常处理方法	153
6.2.1	异常的捕获与处理	153
6.2.2	异常的抛出	155
6.2.3	自定义异常	157
6.3	本章小结	159
	思考与练习	160

7 输入/输出流	161
7.1 流概述	161
7.1.1 流	161
7.1.2 输入/输出流	161
7.2 基本输入/输出流	163
7.2.1 InputStream 类	164
7.2.2 OutputStream 类	167
7.2.3 Reader 类	170
7.2.4 Writer 类	173
7.3 标准输入和输出	175
7.3.1 标准输入	175
7.3.2 标准输出	177
7.3.3 标准出错	177
7.4 文件处理	178
7.4.1 File 类方法	178
7.4.2 FileInputStream 类	180
7.4.3 FileOutputStream 类	181
7.4.4 文件过滤器	181
7.4.5 随机文件操作	182
7.5 本章小结	184
思考与练习	185
8 图形用户界面编程	187
8.1 组件概述	187
8.1.1 组件	187
8.1.2 容器	188
8.1.3 AWT 组件.....	189
8.1.4 Swing 组件	190
8.1.5 Swing 与 AWT 的区别.....	191
8.2 AWT 图形化界面基础	191
8.2.1 Frame 框架	192
8.2.2 Panel 面板.....	193
8.2.3 Button 按钮	194
8.2.4 Label 标签.....	195
8.2.5 文本框与文本域	196
8.2.6 复选框与单选按钮	198
8.2.7 Choice 下拉列表	200
8.2.8 List 列表	200
8.2.9 Dialog 对话框	202
8.2.10 Canvas 画布.....	203
8.3 事件处理机制	205
8.3.1 事件处理基础	205
8.3.2 键盘事件	207

8.3.3	文本事件	211
8.3.4	鼠标事件	212
8.3.5	窗口事件	215
8.3.6	焦点事件	216
8.3.7	事件适配器	217
8.4	本章小结	218
	思考与练习	219
9	高级 GUI 设计与多媒体技术	222
9.1	布局设计	222
9.1.1	布局概述	222
9.1.2	FlowLayout 布局	222
9.1.3	BorderLayout 布局	224
9.1.4	CardLayout 布局	225
9.1.5	GridLayout 布局	227
9.1.6	GridBagLayout 布局	228
9.1.7	BoxLayout 布局	230
9.1.8	组合使用布局管理器	232
9.2	菜单组件	233
9.2.1	菜单栏	233
9.2.2	菜单	235
9.2.3	菜单项	235
9.2.4	弹出式菜单	235
9.2.5	菜单创建步骤	236
9.3	Swing 组件	238
9.3.1	组件常用方法	240
9.3.2	JFrame 窗体	244
9.3.3	标签与文本组件	244
9.3.4	按钮组件	247
9.3.5	复选框与单选按钮	249
9.3.6	下拉列表组件	251
9.3.7	表格组件	254
9.3.8	进度条组件	258
9.3.9	树组件	259
9.3.10	中间容器	262
9.3.11	多文档界面	265
9.3.12	JOptionPane 和 JDialog 对话框	266
9.4	多媒体程序设计	271
9.4.1	Java 屏幕坐标系	271
9.4.2	Graphics 主要方法	272
9.4.3	图形处理	272
9.4.4	图像处理	276
9.4.5	动画处理	280

9.4.6 声音处理	281
9.5 本章小结	283
思考与练习	283
10 Java 线程机制	287
10.1 Java 中的多线程机制	287
10.1.1 线程的基本概念	287
10.1.2 线程的生命周期	288
10.1.3 多线程的实现方法	289
10.1.4 线程调度与优先级	290
10.2 多线程类的创建与实现	291
10.2.1 Thread 类	291
10.2.2 Runnable 接口	293
10.3 线程的同步与死锁	295
10.3.1 线程的等待	295
10.3.2 线程的同步	298
10.3.3 线程的死锁	301
10.3.4 线程的调度	301
10.4 本章小结	305
思考与练习	305
11 Applet 程序	308
11.1 Applet 概念	308
11.1.1 Applet 与 Application	308
11.1.2 Applet 类	308
11.1.3 HTML 中的 APPLET 格式	309
11.1.4 Applet 的创建与运行	310
11.2 Applet 具体应用	312
11.2.1 Applet 图形绘制	312
11.2.2 Applet 图像处理	314
11.2.3 Applet 声音技术	316
11.2.4 Applet 动画制作	318
11.2.5 Applet 类各方法的监控	321
11.2.6 JavaApplet 与 JavaScript 间的通信	322
11.3 本章小结	324
思考与练习	325

应用开发篇

12 Java 数据库连接	326
12.1 数据库概要	326
12.1.1 数据库纵览	326
12.1.2 SQL 语言简介	327
12.1.3 ODBC 接口机制	329

12.1.4 建立 ODBC 数据源	330
12.2 JDBC 基础	331
12.2.1 JDBC 驱动类型	331
12.2.2 JDBC 层次结构	332
12.3 基于 JDBC 访问数据库	333
12.3.1 JDBC 数据库访问过程	333
12.3.2 JDBC 数据库编程过程	334
12.4 JDBC 应用实例	337
12.4.1 添加数据	337
12.4.2 查询数据	338
12.4.3 修改数据	339
12.4.4 删除数据	340
12.4.5 创建表	341
12.4.6 删除表	341
12.5 本章小结	342
思考与练习	342
13 Java 网络编程基础	344
13.1 网络编程基础	344
13.2 URL 编程	346
13.2.1 URL 构成	346
13.2.2 URL 类	347
13.2.3 URLConnection 类	349
13.3 InetAddress 类	351
13.4 Socket 通信	352
13.4.1 Socket 概念	353
13.4.2 Socket 通信步骤	353
13.4.3 ServerSocket 类与 Socket 类	354
13.5 本章小结	359
思考与练习	359
14 Servlet 程序设计	361
14.1 Servlet 基础	361
14.1.1 Java Servlet	361
14.1.2 Servlet 特点	361
14.1.3 Servlet 功能	362
14.1.4 Servlet 与 CGI	362
14.2 Servlet 运行	363
14.2.1 Servlet 执行过程	363
14.2.2 Servlet 运行环境	364
14.2.3 Servlet 生命周期法	367
14.3 Servlet API	369
14.3.1 Servlet 接口	369

14.3.2	GenericServlet 与 HttpServlet 类	369
14.3.3	ServletRequest 与 HttpServletRequest 接口	370
14.3.4	ServletResponse 与 HttpServletResponse 接口	371
14.4	Servlet 程序设计实例	371
14.5	本章小结	374
	思考与练习	375
15	JSP 程序设计	376
15.1	JSP 基础	376
15.1.1	JSP 与 Java	376
15.1.2	JSP 语法基础	377
15.2	JSP 程序开发	382
15.2.1	JSP 与 JavaBean	383
15.2.2	JSP 与 Servlet	384
15.2.3	JSP 的表单使用	386
15.2.4	JSP 与 EJB	387
15.2.5	RMI 与 CORBA	388
15.3	JSP 应用程序举例	390
15.4	本章小结	392
	思考与练习	393
16	Java 应用开发与课程设计实例	394
16.1	系统开发概述	394
16.2	系统开发概述需求分析	394
16.2.1	系统概述	394
16.2.2	系统平台需求	395
16.2.3	系统功能需求	395
16.3	总体设计	395
16.3.1	系统总体构思	395
16.3.2	系统模块设计	395
16.3.3	系统数据流程	396
16.3.4	数据库的设计	397
16.4	详细设计	398
16.4.1	主控平台模块	398
16.4.2	初始化模块	402
16.4.3	信息管理模块	404
16.4.4	成绩管理模块	408
16.4.5	信息查询模块	411
16.4.6	其他功能模块	412
16.5	本章小结	414
	思考与练习	414
附录 A	《Java 程序设计及应用开发教程》实验	416
	实验一 Java 开发环境设置与简单程序编写	416

实验二	Java 语言基础	417
实验三	Java 流程控制	418
实验四	定义类和使用类对象	419
实验五	继承与多态	420
实验六	Java 的异常处理	421
实验七	输入/输出和文件处理	422
实验八	AWT 组件与事件处理机制	423
实验九	布局设计、菜单及 Swing 组件	426
实验十	多媒体与 Applet 程序设计	429
实验十一	Java 线程机制	430
实验十二	Java 数据库程序设计	431
实验十三	Java 网络程序设计	432
实验十四	Servlet 与 JSP 及应用开发程序设计	433
附录 B	思考与练习参考答案 (部分)	436
参考文献	450

基础知识篇



Java 概述

【本章提要】Java 语言以其具有安全性、跨平台、面向对象、简单便捷等显著特点而著称，已成为 IT 业界的领先技术。本章主要讲述了 Java 的起源与发展、技术现状、特点与分类、Java 运作机制、JDK 的获取与安装、JDK 环境变量设置、Java 程序简单开发等内容，同时介绍了 Java 编辑工具，尤其是 JCreator Pro 3.5 编辑工具，最后讲解了 Java 与 C/C++ 的比较。

1.1 Java 发展与特点

互联网（Internet）的出现、万维网（Web）的付诸应用、网格（Grid）技术的悄然兴起，使得计算机运作模式步入了网络计算时代，早期计算机同构运作态势发生了巨大的变更，网络计算模式可以应对异构的计算机系统与异构的网络平台，以发挥网络的性能，据此实施系统与应用开发的编程语言也得到同步的发展。

计算机程序设计语言经历了机器语言、汇编语言、高级语言（即算法语言、过程语言）与面向对象（非过程化语言或结构化程序设计语言）程序设计及智能型语言设计等多个阶段。尤其是面向对象程序设计语言，可直观地反映客观世界的真实表象，使软件开发人员能够运用人类认识事物所采用的一般思维方式来进行软件开发，它与自然语言间的差距最小，是当今软件开发与应用的主流技术。为了发挥 Internet 的巨大作用，需要一种能运行在网络中各种计算机上、具有高移植性和与所用平台无关的跨平台的编程语言，Java 语言以其面向对象、与平台无关、多线程、安全可靠等特性成为互联网时代程序设计语言中的佼佼者脱颖而出，得到了广大开发人员的青睐。

目前，基于 Java 的各项技术已经成为 IT 业界的领先技术。尽管 Sun 公司设计师们的初衷只是想将跨不同硬件平台上的通用控制语言——Java 应用在机顶盒、微波炉这样的智能型家电设备上，但他们这种跨平台的理念，无意插柳柳成荫的举措，唤来了 Java 今天的成功，达到了如下的预期目标：

- ☞ 创建一种面向对象的程序设计语言语境。
- ☞ 提供一个程序代码独立于平台的解释执行程序的运行环境。
- ☞ 吸收 C 和 C++ 的优点，使程序员容易掌握；剔除 C 和 C++ 中影响程序健壮

性的部分，使程序更安全，例如指针、内存申请和释放。

☞ 实现多线程，使得程序能够同时执行多个任务。

☞ 提供代码校验机制以确保系统安全性，且提供程序代码动态下载机制。

1.1.1 Java 起源与发展

1991 年，Sun 公司涉足消费电子产品市场，尝试异构平台语言机制。同年 4 月，Sun 成立了 Green 项目组，旨在开发一种面向家用电器市场的具有平台无关性特征的软件产品，C 语言无法胜任，因而拟定以 C++ 为基础开发新的程序语言，在为该程序语言取名时人们看到窗外的橡树，随即命名为 Oak。

1992 年 10 月 Green 小组组员在印尼爪哇岛度假喝咖啡时灵机一动，意欲将这种技术转移到 Web 上，并把 Oak 改名为 Java。

1993~1994 年间，Web 在 Internet 上开始流行，Java 由此而悄然兴起。

1995 年 5 月，Sun 公司正式发表 Java 与 Hot Java 产品。1995 年 5 月 23 日，Java 语言诞生。同年 10 月，网景（Netscape）公司与 Sun 公司合作，在 Netscape Navigator 中支持 Java。该年 12 月，微软（Microsoft）Internet Explorer 浏览器加入支持 Java 的行列，同时出现了 Java Beta 测试版。

1996 年 2 月，Java Beta 测试版结束，Java 1.0 版正式诞生。同年 4 月，10 个最主要的操作系统供应商申明将在其产品中嵌入 Java 技术，至该年 9 月，约 8.3 万个网页应用了 Java 技术来制作。

1997 年 2 月，Java 急速发展至 1.1 版，届时分为 Personal Java 和 Embedded Java 两大派。1997 年 4 月 2 日，JavaOne 会议召开，参与者逾万人，创当时全球同类会议规模之纪录。同年 9 月，Java Developer Connection 社区成员逾 10 万人。

1998 年 12 月间，Java 2 企业平台 J2EE 发布。

1999 年 6 月，Sun 公司发布 Java 的三个版本：标准版（Java 2 Standard Edition, J2SE）、企业版（Java 2 Enterprise Edition, J2EE）和微型版（Java 2 Micro Edition, J2ME）。同年 7 月升级成 1.2 版。

2000 年 9 月升级至 1.3 版。

2001 年 7 月升级到 1.4 版，并形成 J2EE、J2SE 与 J2ME 体系。同年 9 月，J2EE 1.3 发布。

2002 年 2 月间，J2SE 1.4 发布，自此 Java 的计算能力有了大幅提升。

2004 年 9 月 30 日 18:00，J2SE 1.5 发布，这是 Java 语言发展史上的又一里程碑性事件，为了表示其重要性，J2SE 1.5 更名为 J2SE 5.0，故而两者同时出现在该分支领域中。

2005 年 6 月，Sun 公司通过会议形式公开了 Java SE 6。此时，Java 的各种版本着手更名，用以取消其名中的数字“2”：J2EE 更名为 Java EE，J2SE 更名为 Java SE，J2ME 更名为 Java ME。

概而言之，1991 年迄今（尤其是 1995 年以来），美国 Sun 公司推出的 Java 语言，以其具有安全性、跨平台、面向对象、简单便捷、适用于网络平台等显著特点而著称。Java 语言的这些特点恰好符合互联网发展的要求，于是 Java 和 Internet 迅速融合并互相推动

快速发展。Microsoft、IBM、Oracle、Netscape、Apple 等大公司纷纷与 Sun 公司签订合同，授权使用 Java 平台技术。Microsoft 公司总裁比尔·盖茨先生对 Java 语言也有很高的评价与赞誉，称之是长时间以来最卓越的程序设计语言。

1.1.2 Java 的技术现状

目前，Sun 公司把 Java 平台划分成 J2EE、J2SE、J2ME 共三个平台，针对不同的市场目标和设备进行定位。

(1) J2SE

J2SE 主要目的是为台式机和 workstation 提供一个开发和运行的平台。我们在学习 Java 的过程中，主要是采用 J2SE 来进行开发。

(2) J2ME

J2ME 主要是面向消费电子产品，为消费类电子产品提供一个 Java 的运行平台，使得 Java 程序能够在手机、机顶盒、PDA 等产品上运行。

(3) J2EE

J2EE 主要目的是为企业计算提供一个应用服务器的运行和开发平台。J2EE 本身是一个开放的标准，任何软件厂商都可以推出自己的符合 J2EE 标准的产品，使用户可以有多种选择。IBM、Oracle、BEA、HP 等近 30 家公司业已推出自己的产品，其中尤以 IBM 公司的 WebSphere 产品和 BEA 公司的 weblogic 产品最为著名。J2EE 将逐步发展成为可以与 Microsoft 的 .NET 技术相抗衡的网络计算平台。J2EE、J2SE 与 J2ME 三个 Java 平台的关系如图 1-1 所示。

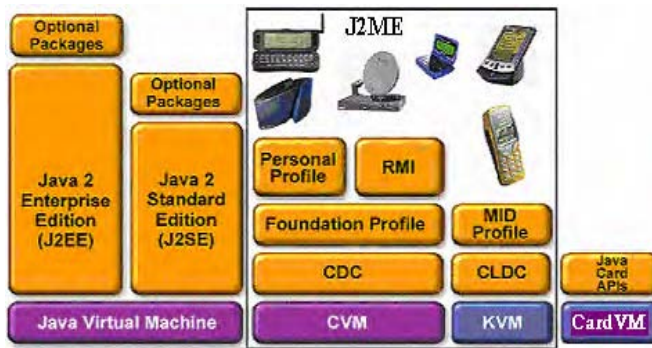


图 1-1 Java 平台示意图

1.1.3 Java 语言的特点

1. 面向对象性

面向对象是现代编程语言的重要特性之一，也是 Java 语言的特色所在。在现实世界中任何实体都是可以表示为对象的，而对象间通过消息来相互传递、作用。对于传统的面向过程编程语言而言，若其中心是过程、驱动是算法的话，那么对于面向对象的编程

语言而言，则对象是它们的中心，消息是它们的驱动。前者可用公式表示为：程序=算法+数据；后者可表示为程序=对象+消息，两者的基点显而易见。

Java 语言和所有新一代的程序设计语言一样，也采用了面向对象技术，并据此而升华、完善，使得面向对象技术更趋简单。所有的 Java 程序和 Applet 程序均是对象，通过封装性、继承性、多态性实施开发。封装性实现了模块化和信息隐藏，继承性实现了代码的复用，用户可以建立并运作自己的类库、构造方法重载等。

2. 简单便捷性

Java 语言的简单首先体现在精简的系统上，力图用最小的系统实现足够多的功能；对硬件的要求不高，在小型的计算机上可便捷地运行。其次，Java 基于面向对象技术，通过提供最基本的方法来完成指定的任务，只需了解一些基本的概念，就可以编写出适合于各种情况的应用程序。

Java 语言源于 C++ 语言（Java 语言采用了 C 语言中的大部分语法，熟悉 C 语言的程序员会发现，Java 语言在语法上与 C 语言极其相似），但剔除了 C++ 中复杂罕用的功能，略去了运算符重载、多继承的复杂概念，采用了单一继承、多线程、引用等机制，使程序设计更加简单，充分体现了系统的简单便捷性。

3. 平台无关性

网络上充满了各种不同类型的计算机和操作系统，为使 Java 程序能在网络的任何地方运行，Java 编译器编译生成了与体系结构无关的字节码结构文件格式，然后再由虚拟机转换成某种处理器的专用代码，因而任何种类的计算机，只要在其处理器和操作系统上有 Java 运行的环境，字节码文件就可以在该计算机上运行，从而实现了用 Java 语言编写的应用程序不用修改就可在不同的软硬件平台上运行，如 UNIX、Linux、Mac 或 Windows，使得“一次编写，到处运行（Write Only, Run Anywhere）”梦想成真，也体现了系统的平台无关性。图 1-2 表述了 Java 语言的平台无关性，该性能也是 Java 快速发展与普及的动因所在。

图 1-2 Java 平台无关性示意图

4. 语言健壮性

Java 语言在编译及运行程序时，都要进行严格的检查。作为一种强制类型语言，Java 在编译和连接时都进行大量的类型检查，以防止不匹配问题的发生。若引用一个非法类型或执行一个非法类型操作，Java 将在解释时指出该错误。在 Java 程序中取消了指针计算，大大减少了该类错误发生的可能性；而且 Java 的数组并非用指针实现，这样就可以在检查中避免数组越界的发生。垃圾自动回收机制也增加了 Java 的健壮性。

5. 运作安全性

作为网络语言，Java 必须提供足够的安全保障，并且要防止病毒的侵袭。Java 在运行应用程序时，严格检查其访问数据的权限，如不允许网络上的应用程序修改本地的数据。下载到用户计算机中的字节代码在其被执行前要经过一个核实工具，一旦字节代码被核实，便由 Java 解释器来执行，该解释器通过阻止对内存的直接访问来进一步提高 Java 的安全性。同时，Java 健壮可靠性、实现内存管理自动化的虚拟机（JVM）、提供加密技术与支持多种安全网络协议等特性也增强了 Java 的运作安全性。

6. 系统移植性

Java 语言的无关性，成为该类应用程序可方便地移植到网络上的其他计算机上运行的良好基础，使得 Java 语言应用程序在已配备了 Java 语言解释及运行环境的任一计算机上运行是可行的，即使是在不同的操作系统平台上。与此同时，Java 语言类库中的与不同平台的接口也得到了实现，这样类库具有的可移植性更深化了系统的开发的可移植性。Java 运行时系统可以移植到不同的处理器，Java 的编译器是由 Java 语言实现的，解释器是由 Java 语言和标准 C 语言实现的，这就使得 Java 语言系统自身就已经实现了可移植性，因此 Java 系统可以较为方便地进行移植工作。

7. 动态扩展性

语言具有动态性，即可以在本地或网上动态地加载各种类库，在执行过程中可随意地增加新方法、实例变量等，这一特点使之非常适合于在网络运行，同时也非常有利于软件的开发，此种运行中的程序只在需要时才会加载相应模块的机制，可加快程序的运行效率与开销。Java 语言是基于 C++ 的一种编程语言，它们之间有许多相似处，便于扩展。当然，Java 的动态扩展性还体现在：Java 语言可以把内部的方法映射成软件库所定义的功能，可实现动态地与虚拟机链接。

8. 程序高性能

虽然 Java 语言程序是解释执行的，但程序仍然具有非常高的性能，通过周详设计的 Java 字节码技术，可快速地将编辑的 Java 语言转换成高性能的机器码来执行，相应的自动寄存器分配与编译器对字节码的一些优化可使之生成高质量的代码。随着 Java 虚拟机的改进和“即时编译”（Just in Time）。技术的出现，使得 Java 的执行速度有了更大的提高，Java 语言系统的高性能使其即使在相当低档的 CPU 上仍能顺利运行。

9. 多线程机制

多线程机制使应用程序可以同时进行不同的操作，处理不同的事件。线程是一种特殊的进程，多个线程不仅可以共用一个内存区域，而且可以共享同一组系统资源。线程间进行通信和切换时的系统开销要比进程机制小得多。Java 语言本身提供了一个 `Thread` 类和一组内置的方法，它负责生成线程、执行线程或者查看线程的执行状态。

在 Java 的多线程机制中，不同的线程处理不同的任务，互不干涉，不会由于某一任务处于等待状态而影响了其他任务的执行，使得各多线程并发运作、独立执行，提高了系统的运行效率，这样就容易地实现网络上的实时交互操作。

1.2 Java 程序的分类

基于实现环境的差异，Java 语言主要可划分为 Java Application、Java Applet、Java Servlet、JavaBean 与 JSP 五种程序类型。

1.2.1 Java Application

Java Application 是一个完整的程序，它需要独立的解释器来解释执行。Java Application 和其他 Windows 应用程序一样，可以居于窗体界面运行，也可以在命令行运行。Java Application 是本教材研究的主体程序，可以用来开发命令行程序、窗体程序或服务端程序，许多著名的软件产品都是用它来开发的，例如大型数据库 Oracle（如图 1-3 所示）。



图 1-3 Java Application 开发的 Oracle 安装程序界面

1.2.2 Java Applet

Java Applet 也称为小应用程序，适合在网络中传输，安全可靠，功能强大，特别适合用于开发 Web 的交互式界面。它是一种嵌入到 Web 页面的网络应用程序，由 Web 浏览器内部的 Java 解释器来解释运行，它的主要功能是在浏览器端实现某些特殊效果。网络用户访问服务器的 Applet 时，这些浏览器先下载 Applet 程序，然后在浏览器端运行所

下载的 Applet 程序。图 1-4 所示的是一个 Applet 应用实例。



图 1-4 Java Applet 动画设计实例

1.2.3 Java Servlet

Servlet 是一种运行于 Web 服务器端的 Java 程序，也叫服务器端小程序。它既可以与客户端的 Applet 进行交互，也可以直接与客户端的 HTML 页面交互，在许多大型的应用中可以作为现有的 Internet 技术和 Java 技术的中间桥梁或枢纽。

1.2.4 JavaBean

JavaBean 是一种可重用、独立于平台的 Java 程序组件。通常，可以将一些比较复杂的、需要重复使用的功能封装成一个 Bean 组件。

1. JavaBean 的功能

通过 JavaBean 可以扩充 Java 程序的功能，可以快速地生成新的应用程序。一个 Bean 能被设计成在用户工作站上独立运行，也可以与其他一组分布式组件协同工作。其功能没有限制，可以完成一个简单的功能（如检查一个文件的语法拼写等），也可以完成复杂的功能（如预测一个股票或期货品种的业绩等）。

2. JavaBean 的特点

JavaBean 组件具有如下特点。

- ☞ 可以使用工具栏控制一个 Bean 的相关属性、事件和方法。
- ☞ 一个 Bean 具有 Java “一次编程，随处可用”的特性。
- ☞ Bean 的配置保存在永久存储区域中，使用可以按需选择性地恢复。相关辅助软件可以帮助使用者配置 Bean。
- ☞ Bean 的注册可来自其他对象的事件，且能够再度产生事件送往其他对象。

3. JavaBean 的开发工具

人们通常使用 JavaBean 的 Bean 开发工具包 (Bean Developer Kit, BDK) 来付诸应用, 可以配置 Bean 的应用程序开发工具 (Application Builder Tool) 生成一个可工作的应用程序。该工具软件可以从 Java Soft 的相关站点 (<http://java.sun.com>) 下载, 这是一个可以用来创建、配置与连接一组 Bean 的简单工具, 其中还包括一些带有源代码的 Bean 示例。下载以后即可安装、使用, 并可创建具体实例连接使用。

1.2.5 JSP

JSP (Java Server Page) 是一种用于生成动态网页的技术, 类似 ASP, 它基于 Servlet 技术, 在传统的网页 HTML 文件 (*.htm) 中加入 Java 程序片段 (Scriptlet) 和 JSP 标记 (tag), 就构成了 JSP 网页 (*.jsp)。JSP 程序同样可运行于 Web 服务器端, 实现程序与页面格式控制的分离, 网络上传送给客户端的仅是得到的结果。

Web 服务器在遇到访问 JSP 网页的请求时, 首先执行其中的程序片段, 然后将执行结果以 HTML 格式返回给客户。程序片段可以操作数据库、重新定向网页以及发送 E-mail 等, 这就是建立动态网站所需要的功能。JSP 被广泛应用于新闻网、电子商务网等系统用户界面的开发。图 1-5 所示的是一个网上银行系统的用户界面实例。



图 1-5 使用 JSP 开发的用户界面示意图

1.3 Java 运作机制

1.3.1 Java 虚拟机

Java 程序是基于 Java 虚拟机 (Java Virtual Machine, JVM) 而运行的。JVM 是软件模拟的虚拟计算机, 可以在任何处理器上 (无论是在计算机中还是在其他电子设备中)

安全、兼容地执行 Java 应用程序。Java 虚拟机可建立于不同的硬件系统、不同的操作系统平台，其在整个系统平台体系结构中的层次位置如图 1-6 所示。JVM 定义了指令集、寄存器集、类文件结构栈、垃圾收集堆、内存区域等，提供了跨平台能力的基础框架。Java 虚拟机的“机器码”保存在字节码文件 (*.class) 中，Java 程序的跨平台主要是指字节码文件可以在任何具有 Java 虚拟机的计算机或者电子设备上运行。

图 1-6 Java 虚拟机层次架构

1.3.2 JVM 执行过程

Java 中 JVM 执行过程体现了 Java 程序的运作过程。程序设计员通过编辑软件编写出 Java 源程序，Java 源程序需要通过编译器编译成为字节码文件 (*.class)，Java 虚拟机中的 Java 解释器负责将字节码文件解释成为特定的机器码来运行。Java 程序的编译和执行过程如图 1-7 所示。其间经过加载代码、校验代码、代码执行三个步骤。Java Applet（小应用程序）将字节码嵌入超文本文件，在浏览器中运行。

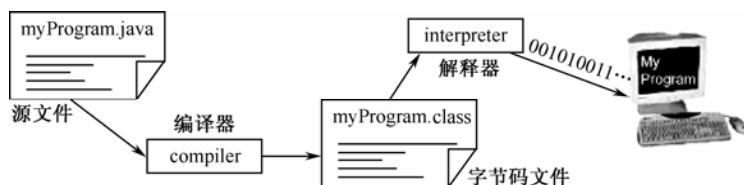


图 1-7 Java 程序编译执行过程

JVM 执行过程有如下三个典型特点。

- ☞ 多线程：Java 虚拟机支持多个线程同时运行，这些线程可独立地执行 Java 代码，处理公共数据区和私有堆栈中的数据。
- ☞ 动态连接：Java 虚拟机具有的动态连接使得 Java 程序适合在网上运行。
- ☞ 异常处理：Java 虚拟机提供了可靠的异常处理。

1.4 Java 运行平台

Java 不仅是编程语言，还是一个开发平台，Java 系统给程序员提供了许多工具，包括编译器、解释器、文档生成器和文件打包工具等。同时 Java 还是一个程序发布平台，有两种主要的发布环境：首先是 Java 运行时环境（Java Runtime Environment, JRE），它包含了完整的类文件包，其次是许多主要的浏览器都提供的 Java 解释器和运行时的环境。

如前所述，目前 Sun 公司把 Java 平台划分成 J2EE、J2SE、J2ME 三个平台，针对不同的市场目标和设备加以定位。

Java Development Kit (Java 开发工具集) 简称 JDK，是 Sun 公司为全世界的 Java 程序员提供的一个免费的 Java 程序开发包和文档。它运行在 Windows 下的 MS-DOS 窗口状态中，用户需要在 DOS 提示符下输入命令。除此方法外，还可以在 JCreator、JBuilder、Java WorkShop、Visual J++ 6.0、Visual Age for Java、Free Java 等集成环境中输入、编辑并运行源程序。本教材主要介绍 JCreator 编辑工具。

JDK 由一个标准类库和一组用来建立程序与测试的 Java 实用程序组成。其核心 Java API 是一些预定义类库，开发人员需要用这些类来访问 Java 语言。JDK 还包含了 Java 运行环境以及可以供用户调用的 API (应用程序接口)。Java API 包括一些重要的语言结构以及基本图形、网络和 I/O 等。

JDK 包括了 Java 用于生成字节码编译器 (命令) Javac.exe，Java 程序用于执行命令的解释器 Java.exe (Java 解释器是面向 Java 程序的一个独立运行系统，它可以一种稳定、高性能的方式运行那些独立于平台的 Java 字节码)，以及帮助文档生成器命令 Javadoc.exe 等。所有这些命令都可以在命令行下运行。除了这些命令，JDK 和帮助文档可以在 Sun 公司的网站下载：<http://Java.sun.com/>。

1.4.1 JDK 1.5 系统安装

1. JDK 1.5 获取与安装

(1) JDK 1.5 的获取

学习 Java 通常是从 J2SE 着手开始的，因而本书基于 J2SE 而表述。在此，仅以 JDK 1.5 (即 J2SE Java Development Kit 5.0 或谓 Java EE 5 SDK) 为例扼要介绍如何安装 JDK 开发包。启动 IE 浏览器，登录 Sun 公司网站 (<http://java.sun.com/downloads/>)，在如图 1-8 所示的 Java 资源下载网站对话框中，单击 downloads links 下 Java EE 5 SDK，即可实施下载，直至完成。



图 1-8 Java 资源下载网站



图 1-9 解压、安装 JDK1.5 示意图

(2) JDK 1.5 的安装

运行下载的可执行安装程序（如 `jdk-1_5_0_04-windows-i586-p.exe`），系统开始解压这个安装包，出现如图 1-9 所示的画面。在之后的步骤里，我们可以接受 Sun 公司的许可条约，按屏幕提示选择安装目录和安装组件以及是否为系统现有的浏览器安装 Java 插件等诸多环节，直至完成安装。

2. 系统路径与主要文件

若 JDK 1.5 安装到 `E:\jdk1.5` 目录下，则会生成图 1-10 所示的目录结构及部分相关文档。JDK 1.5 安装完毕后，在其安装路径下会产生以下几个子主要目录。



图 1-10 Java 目录结构示意图

📁 `bin` 目录：存放 Java 编程所用的实用工具程序。

📁 `demo` 目录：存放演示实例。

📁 `include` 目录：存放系统 C 语言的头文件等。

📁 `jre` 目录：存放运行时相关文件。

📁 `lib` 目录：存放供程序员调用的 Java 类库文件。

📁 `sample` 目录：分类存放 Java 程序的实例文件，以资参考。

作为 JDK 实用程序，该目录的 `.bin` 中包含了如下几个主要程序。

📁 `Javac.exe`：Java 编译器，用于将 Java 源代码文件（`.java`）编译成能被 Java 解释器运行的类（字节码）文件（`.class`）。命令格式为：

```
javac [选项] 源文件名
```

如：

```
javac test.java
```

📁 `Java.exe`：Java 字节码解释器：即 Java 虚拟机，用于 Java 程序的解释执行，直接从类（字节码）文件执行 Java 应用程序。命令格式为：

```
java [选项] 类名 [参数]
```