

Java 基础案例教程

主 编 陈文兰 刘红霞



北京大学出版社
PEKING UNIVERSITY PRESS

21 世纪全国高职高专计算机案例型规划教材

Java 基础案例教程

主 编 陈文兰 刘红霞
副主编 王 品 谭 敏



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书从解决问题的角度出发,通过案例教学的形式,对 Java 程序设计语言的各个知识点精心设计了独立又有联系的大量案例,将知识讲解融入到案例中,通过任务驱动方式展开教学,使读者在完成任务的过程中,全面掌握 Java 编程基本技能。

本书可作为高职高专院校计算机及相关专业的教材,也可以作为 Java 开发基础培训和自学用书,配合 JSP、J2EE 使用。为配合教学,本书在网上提供有课件、习题解答、程序源码,供学生教师参考。

图书在版编目(CIP)数据

Java 基础案例教程/陈文兰,刘红霞主编. —北京:北京大学出版社,2009.6

(21 世纪全国高职高专计算机案例型规划教材)

ISBN 978-7-301-15232-4

I. J… II. ①陈…②刘… III. Java 语言—程序设计—高等学校:技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 075461 号

书 名: Java 基础案例教程

著作责任者: 陈文兰 刘红霞 主编

责任编辑: 徐 凡

标准书号: ISBN 978-7-301-15232-4/TP · 1011

出 版 者: 北京大学出版社

地 址: 北京市海淀区成府路 205 号 100871

网 址: <http://www.pup.cn> <http://www.pup6.com>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62750667 出版部 62754962

电子邮箱: pup_6@163.com

印 刷 者:

发 行 者: 北京大学出版社

经 销 者: 新华书店

787mm×1092mm 16 开本 16.75 印张 383 千字

2009 年 6 月第 1 版 2009 年 6 月第 1 次印刷

定 价: 26.00 元

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有 侵权必究

举报电话: 010-62752024

电子邮箱: fd@pup.pku.edu.cn

21 世纪全国高职高专计算机案例型规划教材 专家编写指导委员会

主 任	刘瑞挺	南开大学
副主任	安志远	北华航天工业学院
	丁桂芝	天津职业大学
委 员	(按拼音顺序排名)	
	陈 平	马鞍山师范高等专科学校
	褚建立	邢台职业技术学院
	付忠勇	北京政法职业技术学院
	高爱国	淄博职业学院
	黄金波	辽宁工程技术大学职业技术学院
	李 缨	中华女子学院山东分院
	李文华	湖北仙桃职业技术学院
	李英兰	西北大学软件职业技术学院
	田启明	温州职业技术学院
	王成端	潍坊学院
	王凤华	唐山工业职业技术学院
	薛铁鹰	北京农业职业技术学院
	张怀中	湖北职业技术学院
	张秀玉	福建信息职业技术学院
	赵俊生	甘肃省合作民族师范高等专科学校
	周 奇	广东新安职业技术学院
顾 问	马 力	微软(中国)公司 Office 软件资深教师
	王立军	教育部教育管理信息中心

目 录

第 1 章 初识 Java	1	4.1.2 相关知识	71
1.1 配置 Java 开发环境任务	2	4.1.3 知识扩展	80
1.1.1 任务实现过程	2	4.2 “模拟旅行”任务案例	86
1.1.2 相关知识	6	4.2.1 任务实现过程	86
1.2 “打印 ‘Hello World!’ ”任务案例	10	4.2.2 相关知识	88
1.2.1 任务实现过程	10	4.3 小结	91
1.2.2 相关知识	15	4.4 习题	91
1.2.3 知识扩展	15	第 5 章 字符串处理	94
1.3 小结	16	5.1 “单词查找并计数”任务案例	95
1.4 习题	16	5.1.1 任务实现过程	95
第 2 章 数据类型与流程控制	18	5.1.2 相关知识	96
2.1 “计算圆面积”任务案例	19	5.1.3 知识扩展	99
2.1.1 任务实现过程	19	5.2 小结	102
2.1.2 相关知识	20	5.3 习题	102
2.2 “找最大与最小值，并打印程序结果”任务案例	30	第 6 章 异常处理	104
2.2.1 任务实现过程	30	6.1 “用户登录异常处理”任务案例	105
2.2.2 相关知识	32	6.1.1 任务实现过程	105
2.3 小结	42	6.1.2 相关知识	107
2.4 习题	42	6.2 小结	113
第 3 章 类和对象	46	6.3 习题	113
3.1 “格式化输出员工信息”任务案例	47	第 7 章 输入/输出处理	115
3.1.1 任务实现过程	47	7.1 “员工信息管理”任务案例	116
3.1.2 相关知识	50	7.1.1 任务实现过程	116
3.2 小结	59	7.1.2 相关知识	123
3.3 习题	59	7.1.3 知识扩展	135
第 4 章 面向对象高级编程	62	7.2 小结	139
4.1 “员工信息管理系统”任务案例	63	7.3 习题	139
4.1.1 任务实现过程	63	第 8 章 图形用户界面	141
		8.1 “添加员工信息用户界面”任务案例	142

8.1.1 任务实现过程	142
8.1.2 相关知识	149
8.2 “员工管理系统主界面”任务 案例	186
8.2.1 任务实现过程	186
8.2.2 相关知识	190
8.2.3 知识扩展	194
8.3 小结	196
8.4 习题	196
第 9 章 类集框架	200
9.1 “通讯录”任务案例	201
9.1.1 任务实现过程	201
9.1.2 相关知识	203
9.1.3 知识扩展	210
9.2 小结	211
9.3 习题	212

第 10 章 JDBC 与数据库通信	214
10.1 “向数据库中添加员工信息” 任务案例	215
10.1.1 任务实现过程	215
10.1.2 相关知识	217
10.2 “员工管理系统”任务案例	228
10.3 小结	238
10.4 习题	238
第 11 章 多线程	240
11.1 “二维弹球游戏”任务案例	241
11.1.1 任务实现过程	241
11.1.2 相关知识	245
11.2 小结	253
11.3 习题	253
参考文献	254

信息技术的案例型教材建设

(代丛书序)

刘瑞挺

北京大学出版社第六事业部在 2005 年组织编写了《21 世纪全国应用型本科计算机系列实用规划教材》，至今已出版了 50 多种。这些教材出版后，在全国高校引起热烈反响，可谓初战告捷。这使北京大学出版社的计算机教材市场规模迅速扩大，编辑队伍茁壮成长，经济效益明显增强，与各类高校师生的关系更加密切。

2008 年 1 月北京大学出版社第六事业部在北京召开了“21 世纪全国应用型本科计算机案例型教材建设和教学研讨会”。这次会议为编写案例型教材做了深入的探讨和具体的部署，制定了详细的编写目的、丛书特色、内容要求和风格规范。在内容上强调面向应用、能力驱动、精选案例、严把质量；在风格上力求文字精练、脉络清晰、图表明快、版式新颖。这次会议吹响了提高教材质量第二战役的进军号。

案例型教材真能提高教学的质量吗？

是的。著名法国哲学家、数学家勒内·笛卡儿(Rene Descartes, 1596—1650)说得好：“由一个例子的考察，我们可以抽出一条规律。(From the consideration of an example we can form a rule.)”事实上，他发明的直角坐标系，正是通过生活实例而得到的灵感。据说是在 1619 年夏天，笛卡儿因病住进医院。中午他躺在病床上，苦苦思索一个数学问题时，忽然看到天花板上有一只苍蝇飞来飞去。当时天花板是用木条做成正方形的格子。笛卡儿发现，要说出这只苍蝇在天花板上的位置，只需说出苍蝇在天花板上的第几行和第几列。当苍蝇落在第四行、第五列的那个正方形时，可以用(4, 5)来表示这个位置……由此他联想到可用类似的办法来描述一个点在平面上的位置。他高兴地跳下床，喊着“我找到了，找到了”，然而不小心把国际象棋撒了一地。当他的目光落到棋盘上时，又兴奋地一拍大腿：“对，对，就是这个图”。笛卡儿锲而不舍的毅力，苦思冥想的钻研，使他开创了解析几何的新纪元。千百年来，代数与几何，井水不犯河水。17 世纪后，数学突飞猛进的发展，在很大程度上归功于笛卡儿坐标系和解析几何学的创立。

这个故事，听起来与阿基米德在浴池洗澡而发现浮力原理，牛顿在苹果树下遇到苹果落到头上而发现万有引力定律，确有异曲同工之妙。这就证明，一个好的例子往往能激发灵感，由特殊到一般，联想出普遍的规律，即所谓的“一叶知秋”、“见微知著”的意思。

回顾计算机发明的历史，每一台机器、每一颗芯片、每一种操作系统、每一类编程语言、每一个算法、每一套软件、每一款外部设备，无不像闪光的珍珠串在一起。每个案例都闪烁着智慧的火花，是创新思想不竭的源泉。在计算机科学技术领域，这样的案例就像大海岸边的贝壳，俯拾皆是。

事实上，案例研究(Case Study)是现代科学广泛使用的一种方法。Case 包含的意义很广：包括 Example 例子，Instance 事例、示例，Actual State 实际状况，Circumstance 情况、事件、境遇，甚至 Project 项目、工程等。

我们知道在计算机的科学术语中，很多是直接来自日常生活的。例如 Computer 一词早在 1646 年就出现于古代英文字典中，但当时它的意义不是“计算机”而是“计算工人”，

即专门从事简单计算的工人。同理，Printer 当时也是“印刷工人”而不是“打印机”。正是由于这些“计算工人”和“印刷工人”常出现计算错误和印刷错误，才激发查尔斯·巴贝奇(Charles Babbage, 1791—1871)设计了差分机和分析机，这是最早的专用计算机和通用计算机。这位英国剑桥大学数学教授、机械设计专家、经济学家和哲学家是国际公认的“计算机之父”。

20 世纪 40 年代，人们还用 Calculator 表示计算机。到电子计算机出现后，才用 Computer 表示计算机。此外，硬件(Hardware)和软件(Software)来自销售人员。总线(Bus)就是公共汽车或大巴，故障和排除故障源自格瑞斯·霍普(Grace Hopper, 1906—1992)发现的“飞蛾子”(Bug)和“抓蛾子”或“抓虫子”(Debug)。其他如鼠标、菜单……不胜枚举。至于哲学家进餐问题，理发师睡觉问题更是操作系统文化中脍炙人口的经典。

以计算机为核心的信息技术，从一开始就与应用紧密结合。例如，ENIAC 用于弹道曲线的计算，ARPANET 用于资源共享以及核战争时的可靠通信。即使是非常抽象的图灵机模型，也受到二战时图灵博士破译纳粹密码工作的影响。

在信息技术中，既有许多成功的案例，也有不少失败的案例；既有先成功而后失败的案例，也有先失败而后成功的案例。好好研究它们的成功经验和失败教训，对于编写案例型教材有重要的意义。

我国正在实现中华民族的伟大复兴，教育是民族振兴的基石。改革开放 30 年来，我国高等教育在数量上、规模上已有相当的发展。当前的重要任务是提高培养人才的质量，必须从学科知识的灌输转变为素质与能力的培养。应当指出，大学课堂在高新技术的武装下，利用 PPT 进行的“高速灌输”、“翻页宣科”有愈演愈烈的趋势，我们不能容忍用“技术”绑架教学，而是让教学工作乘信息技术的东风自由地飞翔。

本系列教材的编写，以学生就业所需的专业知识和操作技能为着眼点，在适度的基础知识与理论体系覆盖下，突出应用型、技能型教学的实用性和可操作性，强化案例教学。本套教材将会有机融入大量最新的示例、实例以及操作性较强的案例，力求提高教材的趣味性和实用性，打破传统教材自身知识框架的封闭性，强化实际操作训练，使本系列教材做到“教师易教，学生乐学，技能实用”。有了广阔的应用背景，再造计算机案例型教材就有了基础。

我相信北京大学出版社在全国各地高校教师的积极支持下，精心设计，严格把关，一定能够建设出一批符合计算机应用型人才培养模式的、以案例型为创新点和兴奋点的精品教材，并且通过一体化设计、实现多种媒体有机结合的立体化教材，为各门计算机课程配齐电子教案、学习指导、习题解答、课程设计等辅导资料。让我们用锲而不舍的毅力，勤奋好学的钻研，向着共同的目标努力吧！

刘瑞挺教授 本系列教材编写指导委员会主任、全国高等院校计算机基础教育研究会副会长、中国计算机学会普及工作委员会顾问、教育部考试中心全国计算机应用技术证书考试委员会副主任、全国计算机等级考试顾问。曾任教育部理科计算机科学教学指导委员会委员、中国计算机学会教育培训委员会副主任。PC Magazine《个人电脑》总编辑、CHIP《新电脑》总顾问、清华大学《计算机教育》总策划。

前 言

本书以教育部《关于全面提高高等职业教育教学质量的若干意见》为指导，贯彻高职教育“工学结合”教育思想，配合国家示范院校的课程改革进行的“行动导向，任务驱动”的教学改革成果。作者将近几年在企业工作和培训中积累的项目开发实践经验融入到教程中，用通俗、简明的语言深入浅出地讲解了 Java 程序设计技术，同时，辅以前后独立又有联系的一系列案例，以工作任务为驱动，先由任务引出问题，再探讨解决问题的方法，设计解决问题的步骤，引导学生从简单任务逐步提升，最终完成一个较为完整的“员工信息管理系统”，并将知识进行扩展。书中所有程序代码均在开发环境中调试通过。

在本书的编写过程中，作者努力遵循如下原则：以 Java 语言的程序开发为主线，学程序设计语言只是手段，掌握编程的方法才是目的；内容选取力求科学、全面、实用，强调基础，强化应用；在编写上通过提问题、找方法，使读者在实践中掌握语言规则，学会使用语言编程；文字简明，表达力求通俗、易懂；例题、习题前后呼应、丰富多样；难点、重点分散，分析力求深入、透彻。

本书共分为 11 章，第 1 章是 Java 概述，重点介绍了 Java 的开发环境、开发过程。Java 的开发工具很多，在计算机开发语言的历史中，从来没有哪种语言像 Java 那样受到如此众多厂商的支持，有如此多的开发工具。本书精选了两款开发工具：一个是 SUN 公司的 Java 开发工具集(JDK)，它是最古老也是最经典的开发工具，简单易学，适合初学 Java 者，有利于理解 Java 面向对象的设计思想；另一个是国内软件企业使用最多的 Eclipse，这款 Java 开发工具适合大规模企业级 Java 应用开发。第 2 章介绍了 Java 基本语法。本章主旨是通过一些简明的基础案例，使学生理解 Java 的语法规则，学会 Java 基本应用。第 3、4 章是 Java 面向对象编程技术。这部分的内容设计力求实用，通过引入“员工信息管理信息系统”的相关案例，使学生理解面向对象程序设计的基本概念和方法，掌握 Java 中的包及封装、继承、多态等面向对象高级编程技术。第 5 章至第 11 章讲 Java 开发应用，这一部分主要通过“员工信息管理系统”案例的逐步完善，使学生逐渐掌握 Java 中字符串处理、异常处理、I/O 流和文件、图形用户界面和数据库访问等 Java 应用技术，同时在这一部分又精选了“通讯录”“二维弹球游戏”等实用有趣的案例，使学生进一步掌握类集框架和多线程技术，圆满地结束 Java 语言基础开发的学习。

为方便读者，本书还提供了电子课件、程序源码、习题参考答案等参考资料。每章后除基础知识习题外，还精心设计了任务编程题目。建议学生在教材案例基础上先模仿，在模仿的基础上改进，在改进的基础上提高，最终较好地完成每部分的编程任务。

本书可作为高职高专院校计算机及相关专业的教材，也可以作为 Java 开发基础培训和自学用书，配合 JSP、J2EE 使用。

在本书的编写过程中，作者参考了大量的资料，吸取了多位同仁的经验，得到了系部领导和同事们的大力支持，在此诚致谢意。由于编者水平有限，书中不妥之处在所难免，诚恳希望读者提出宝贵意见与建议。

编 者

2009 年 2 月

第 1 章

初识 Java

教学目标和要求：

本章主要介绍 Java 语言的发展及现状、Java 开发工具以及 Java 程序的开发过程等。通过本章学习，读者应该能够掌握以下知识点。

- 了解 Java 语言的历史与现状。
- 了解 Java 虚拟机的工作原理。
- 学会 JDK 的安装以及环境变量的设置。
- 了解 Java 当前常用开发工具的特点。
- 掌握使用 Eclipse 开发 Java 应用程序。

教学难点与重点：

Java 开发环境配置及 Java 程序的开发过程。

1.1 配置 Java 开发环境任务

1.1.1 任务实现过程

【任务说明】

小李刚到软件公司上班，公司将要开发一个项目，经分析，选定 Java 语言来开发，小李接到的任务是为项目开发搭建 Java 语言的开发环境。

【任务目的】

- (1) 了解 Java 语言及其特点和发展状况。
- (2) 了解 Java 语言的开发环境及各自的特点。
- (3) 学习下载安装 Java 的开发工具包(JDK)并设置环境变量。
- (4) 了解常用的 Java 开发工具。

【技术要点】

开发 Java 程序，必须配置相应的开发环境。

【实现步骤】

1. 下载 JDK

- (1) 打开 JDK 下载网址：<http://Java.sun.com/Javase/downloads/index.jsp>。
- (2) 选择不同操作系统下的安装版本，图 1.1 所示为选择 Windows 所示平台的安装文件。
- (3) 下载 JDK1.6。

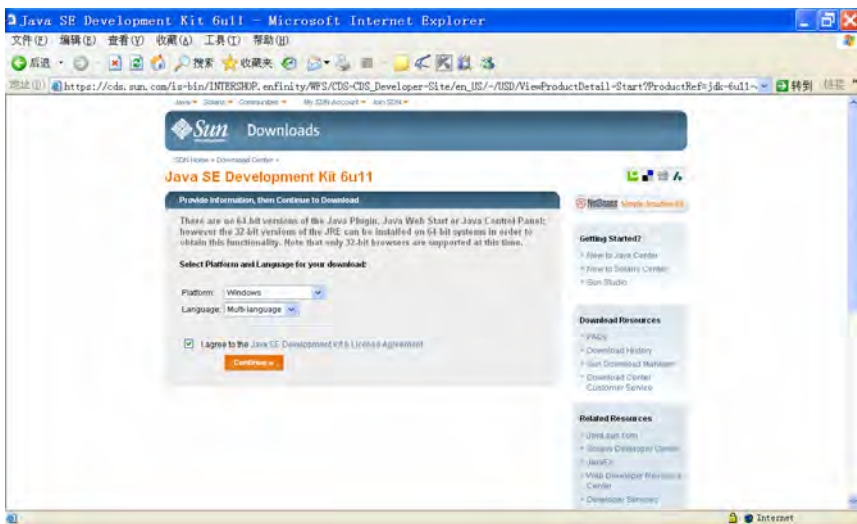


图 1.1 JDK 版本选择

2. 安装 JDK

- (1) 双击所下载的 JDK 安装文件 jdk-6u7-windows-i586-p.exe，开始安装。
- (2) 选择要安装的程序功能，如图 1.2 所示。

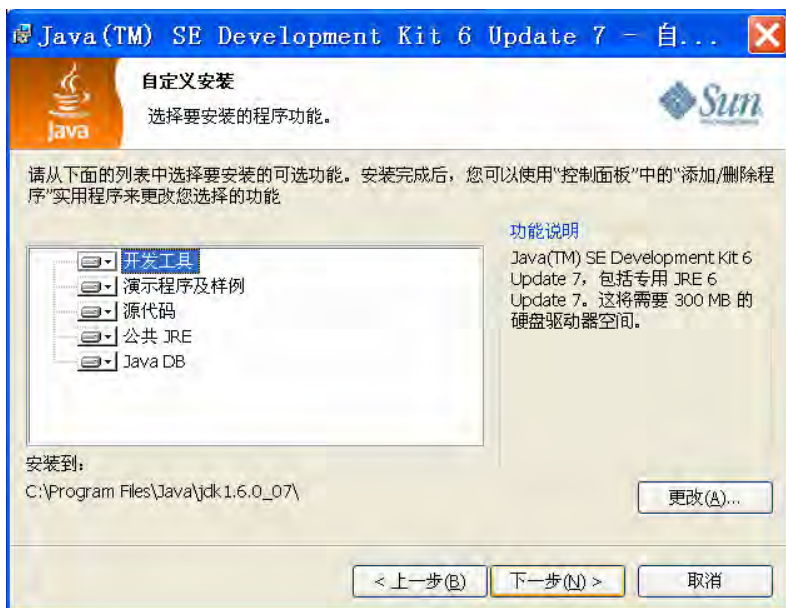


图 1.2 安装 JDK

- (3) 根据系统提示操作，出现图 1.3 所示的对话框后，单击【完成】按钮，结束安装。



图 1.3 JDK 安装完成

3. 设置环境变量

(1) 鼠标右击【我的电脑】，在弹出的快捷菜单中选择【属性】菜单项，打开【系统属性】对话框，如图 1.4 所示。

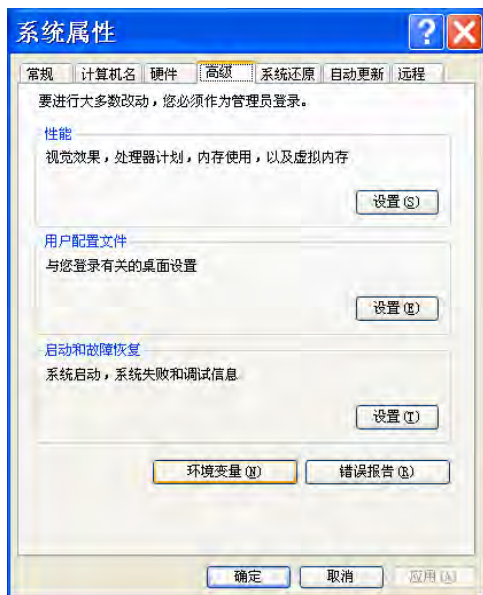


图 1.4 【系统属性】对话框

(2) 选择【高级】选项卡，单击【环境变量】按钮，打开【环境变量】对话框，如图 1.5 所示。



图 1.5 【环境变量】对话框

(3) 编辑系统变量 Path，在变量值末尾添加所安装的 JDK 的目录，如 C:\Program Files\Java\jdk1.6.0_07\bin，如图 1.6 所示。

(4) 单击【新建】按钮，添加变量名 classpath，变量值为.;C:\Program Files\Java\jdk1.6.0_07\lib\dt.jar，如图 1.7 所示。



图 1.6 修改 path 值



图 1.7 添加 classpath 变量

(5) 设置好环境变量后，选择【所有程序】|【附件】|【命令提示符】命令，打开 DOS 窗口，运行 Java 命令，将出现图 1.8 所示的页面，否则安装不成功。

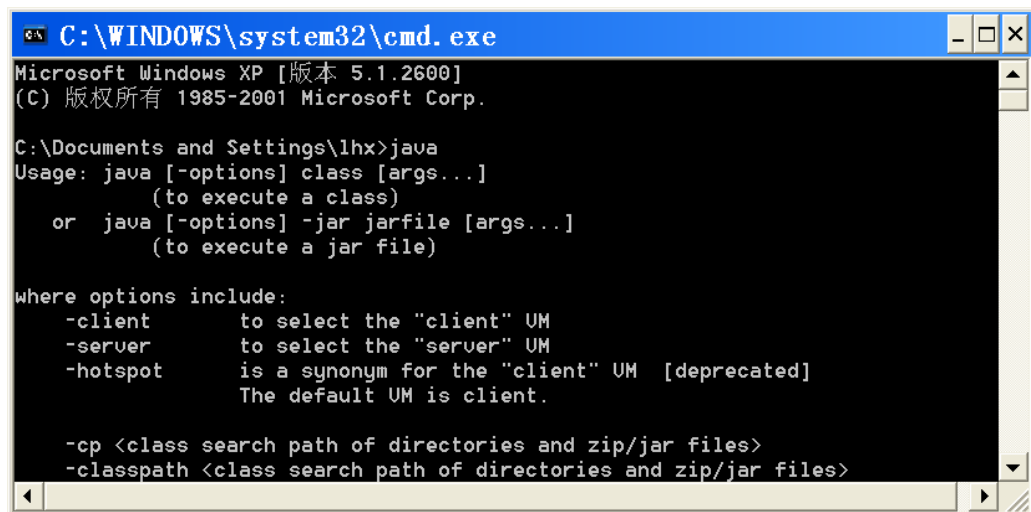


图 1.8 JDK 安装成功测试

1.1.2 相关知识

1. Java 语言的发展及特点

1) Java 的发展

1991 年初, 美国加州的 Sun 公司成立了一个以 James Gosling 为首、名为 Green 的项目研发小组, 其目标是开发消费类电子产品市场。这类产品必须具有平台独立性, 即在任何软硬件平台上都能运行。于是, 该公司自行开发出一种新的语言, 并将该语言命名为 Oak(橡树)。

1994 年, Gosling 继续完善 Oak(发布时改名为 Java), 同时使用 Java 开发出一种名为 HotJava 的 WWW 浏览器, 并于 1995 年初在 Internet 上免费发布。由于 Java 确实是一种分布式、安全性高、包括编译器在内又非常小的适合网络开发环境的语言, 因而一经发布, 立即得到各 WWW 厂商的广泛支持。而今, Java 已成为首选的网络编程语言。

Java 发展到今天, 已从编程语言发展成为全球第一大通用开发平台。Java 技术已被计算机行业的主要公司所采纳。1999 年, Sun 公司推出了以 Java 2 平台为核心的 J2EE、J2SE 和 J2ME 这 3 大平台。

(1) Java 2 Platform, Micro Edition(J2ME)

J2ME 即 Java 2 平台微型版。Sun 公司将 J2ME 定义为“一种以广泛的消费性产品为目标、高度优化的 Java 运行环境”。自 1999 年 6 月在 Java One Developer Conference 上声明之后, J2ME 进入了小型设备开发的行列。鉴于 Java 的特性, 遵循 J2ME 规范开发的 Java 程序可以运行在各种不同的小型设备上。

(2) Java 2 Platform, Standard Edition(J2SE)

J2SE 即 Java 2 平台标准版, 适用于桌面系统应用程序的开发。本书例程就是利用 J2SE 6.0 版的相关 API 包来开发的。

(3) Java 2 Platform, Enterprise Edition(J2EE)

J2EE 是一种利用 Java 2 平台来简化企业解决方案的开发、部署和管理等相关复杂问题的体系结构。J2EE 技术的核心是 Java 平台或 Java 2 平台的标准版, 它不仅保留了标准版的许多优点, 如“一次编写、随处运行”的特性、方便存取数据库的 JDBC API、CORBA 技术, 以及能够在 Internet 应用中保护数据的安全模式等, 同时还提供了对 EJB(Enterprise JavaBeans)、Java Servlets API、JSP(Java Server Pages)以及 XML 技术的全面支持。

随着三大平台的迅速推进, 全球形成了一股巨大的 Java 应用浪潮。

2) Java 的特点

Java 是一种跨平台、适合于分布式计算环境的面向对象编程语言。具体来说, 它具有如下特性: 简单性、面向对象、分布式、解释型、可靠、安全、平台无关、可移植、高性能、多线程、动态性等。下面将重点介绍 Java 语言的面向对象、平台无关和安全等特性。

(1) 面向对象

对象其实是现实世界模型的自然延伸, 现实世界中的任何实体都可以看作是对象, 对象之间通过消息相互作用。另外, 现实世界中的任何实体都可归属于某类事物, 任何对象都是某一类事物的实例。如果说传统的过程式编程语言是以过程为中心、以算法为驱动的,

面向对象的编程语言则是以对象为中心、以消息为驱动的。用公式表示，过程式编程语言为：程序=算法+数据；面向对象编程语言为：程序=对象+消息。

所有面向对象编程语言都支持3个概念：封装、多态性和继承。

Java 语言的封装性较强，因为 Java 无全程变量，无主函数，在 Java 中绝大部分成员是对象，只有简单的数字类型、字符类型和布尔类型除外。而对于这些类型，Java 也提供了相应的对象类型以便与其他对象交互操作。

多态性就是多种表现形式，具体来说，可以用“一个对外接口，多个内在实现方法”表示。针对不同的数据类型，编程人员不必手工选择，只需使用统一的接口名，系统就可自动选择。

继承是指一个对象直接使用另一个对象的属性和方法。Java 提供给用户一系列类(class)，且这些类有层次结构，子类可以继承父类的属性和方法。与其他面向对象编程语言不同的是 Java 只支持单一继承。

(2) 平台无关性

Java 的平台无关性是指用 Java 写的应用程序不用修改就可在不同的软硬件平台上运行。Java 主要靠 Java 虚拟机(JVM)在目标码级实现平台无关性。

Java 的平台无关性具有深远意义。它使得编程人员所梦寐以求的事情(“一次编写、随处运行”)变成事实，这将大大加快和促进软件产品的开发。

(3) 安全性

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，防止不匹配问题的发生。如果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，不能采用地址计算的方法即通过指针访问内存单元，这大大减少了错误发生的可能性；而且 Java 的数组并非用指针实现，这样就可以避免数组越界的发生。无用内存自动回收机制也增加了 Java 的鲁棒性。

2. Java 工作原理

1) Java 虚拟机

Java 虚拟机是软件模拟的计算机，它附着在具体的操作系统之上，本身具有一套虚拟机指令，并有自己的栈、寄存器组等。但 JVM 通常是在软件上而不是在硬件上实现的。

JVM 可以在任何处理器上安全并且兼容地执行保存在.class 文件中的字节码。Java 虚拟机的“机器码”保存在.class 文件中，也可以称之为字节码文件。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，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进行运行。

Java 源程序需要通过编译器编译成为.class 文件(字节码文件)，构成 Java 软件程序的字节码在运行时被加载、校验并在解释器中运行。Java 程序的编译和执行过程如图 1.9 所示。当运行 Applet 时，字节码可被下载，然后由浏览器中的 JVM 进行解释。解释器具备两种功能：一是执行字节码；二是对底层硬件做适当调用。