



规划教材

普通高等学校应用型人才
培养系列规划教材
丛书主编 陈明

Java面向对象程序设计与应用

田登山 夏自谦 编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书详细介绍了如何应用 Java 技术进行网络开发, 主要内容包括: 开发环境搭建、Java 语法基础、流程控制语句、面向对象的基本概念和高级应用、Java 异常机制、Java 数组及 Java 集合框架、IO、线程、套接口, 最后综合各项技术设计实现了简单聊天工具 ChatEasy 系统。

本书在各个章节都提供了大量实例, 并配有课后习题, 以帮助读者巩固所学知识。通过本书, 读者不仅仅可以学会使用 Java 语言进行程序设计, 还能学到面向对象的编程思想和面向接口的设计方法。

本书适合作为高等院校“Java 程序设计”及相关课程的教材, 也可作为各类培训机构 Java 基础相关课程的教材, 以及网络技术应用人员的参考书。

图书在版编目(CIP)数据

Java 面向对象程序设计与应用 / 田登山, 夏自谦编

著. — 北京: 中国铁道出版社, 2011. 12

普通高等学校应用型人才培养系列规划教材

ISBN 978-7-113-13925-4

I. ①J… II. ①田… ②夏… III. ①

JAVA 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 238253 号

书 名: Java 面向对象程序设计与应用

作 者: 田登山 夏自谦 编著

策 划: 杨 勇

读者热线: 400-668-0820

责任编辑: 吴宏伟

编辑助理: 赵 迎

封面设计: 付 巍

封面制作: 白 雪

责任印制: 李 佳

出版发行: 中国铁道出版社(100054, 北京市西城区右安门西街 8 号)

网 址: <http://www.edusources.net>

印 刷: 三河市华丰印刷厂

版 次: 2011 年 12 月第 1 版 2011 年 12 月第 1 次印刷

开 本: 787mm×960mm 1/16 印张: 23.5 字数: 455 千

印 数: 1~3 000 册

书 号: ISBN 978-7-113-13925-4

定 价: 39.80 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社教材图书营销部联系调换。电话: (010) 63550836

打击盗版举报电话: (010) 63549504

普通高等学校应用型人才培养系列规划教材

主 任：陈 明

副主任：蒋宗礼 严晓舟 王锁柱

委 员：（按姓氏笔画排序）

王全民 刘宏志 刘贵龙 刘振华

李也白 张 钢 张晓明 陈志泊

郝 莹 秦绪好 袁 薇 解 凯

管建和 廖湖声

总策划：焦金生

编 辑：杨 勇 周海燕

经过几十年的研究与探索,现代计算机系统功能越来越强大、应用越来越广泛。计算机的广泛应用一方面对人类社会的发展做出了卓越的贡献,另一方面也在推动计算机学科的高速发展,因而一直受到社会的高度关注。

由于计算机学科呈现出的学科内涵宽泛化、分支相对独立化、社会需求多样化、专业规模巨大化和教育大众化等特点,使得计算机企业成为了朝阳企业,这些 IT 企业需要大量的具有专门计算机技能的人才,但传统的研究型计算机教育是以学术教育为基础,以培养计算机精英为目的的计算机教育,与社会和行业对计算机高等教育人才的需求产生了矛盾——大学中单一的计算机精英型教育人才已不能满足实际需要,凸显职业特征的计算机应用型教育异军突起,迅速发展,备受关注。这种矛盾促使教学模式呈现了职业性,并在培养面向知识应用和全面能力方面提出了多种职业性教学模式,如网络工程师、软件工程师、动画设计师、硬件工程师等培养模式。因此,研究和实施计算机应用型人才培养模式势在必行。

什么是计算机应用型教育?我们通过长时间的教育研究和教学经验的总结,认为计算机应用型教育的培养目标可以利用知识、能力和素质 3 个基本要素来描述。

知识是基础、载体和表现形式,从根本上影响着能力和素质。学习知识是为了获得能力和不断地提升能力。知识可以转化为能力和素质,能力对知识具有反作用,促进知识不断发展。

能力是核心,是应用型人才特征的突出表现。就计算机学科而言,培养的人才应具备计算思维能力、算法设计与分析能力、程序设计与实现能力和系统能力(系统的认知、设计、开发、应用能力)。而计算机应用型人才的能力有着其独特的属性,主要包括应用能力(专业能力)和通用能力。应用能力主要是指用所学知识解决专业实际问题的能力。通用能力是指跨职业能力,并不是具体的专业能力和职业技能,而是对不同职业的适应能力,也就是当职业发生变更时,这些能力依然在从业者身上起作用。计算机应用型本科人才应具备的 3 种通用能力是:学习能力、工作能力和创新能力。

素质是指基本素质,即具有良好的公民道德和职业道德,具有合格的政治思想素养,遵守计算机法规和法律,具有人文、科学素养和良好的职业素质等。计算机应用型人才素质主要是指工作的基本素质,并要求在从业中必须具备责任意识,能够对自己职责范围内的工作认真负责地完成。

与此同时，我们认为计算机应用型教育培养目标的实现关键在于课程体系的构建，即课程内容和课程性质的确定。因此，我们将计算机应用型教育课程分为通用课程、基础课程、专业核心课程、专业选修课程、应用课程、实验课程、实践课程等，并建立了相应的教育课程体系，如公共基础课程平台、专业基础课程平台、专业选修课程平台、基本素质平台等，力求通过相应的课程开展来实现培养目标。

目前，应用型人才培养的研究方兴未艾，我们也将会在较长一段实践教学工作中继续探讨和总结经验。此次的这套“普通高等学校应用型人才系列规划教材”系列丛书是应社会需求而建设，经过系统规划与设计，定位于高等院校计算机应用型人才的培养，整套丛书从架构和具体教材的编写上都力求突出能力培养的理念。本系列教材现正在陆续出版中，希望各位老师和读者指正。

《普通高等学校应用型人才系列规划教材》丛书主编

陳明

Java 是一种可以撰写跨平台应用软件的面向对象的程序设计语言，它的风格与 C、C++ 语言十分接近，但它是一个完全面向对象的程序设计语言。Java 技术产生于 20 世纪，具有卓越的通用性、高效性、平台移植性和安全性，并具有开源、海量免费中间件等特点，因此 Java 一直是开发者的首选编程语言。直到现在，Java 仍然是互联网上最流行的语言之一。

鉴于 Java 的实用性，本书作者本着学以致用宗旨，归纳、总结过去多年的教育经验和培训资料，以曾经参与和管理的大量 Web 系统开发项目实例为案例，采用知识点结合示例的讲解方式编写了本书，相信本书对读者学习 Java、快速应用会有很大的帮助。

一、本书特色

(1) 强调面向对象程序设计 (OOP) 的编程思想：本书着重介绍了 OOP 的编程思想，特别是继承、抽象类和接口。为使 OOP 理论内容摆脱枯燥的讲述，本书以实例讲解 OOP 基本思想及其应用，并在类库知识点引入部分实用的设计模式，加深读者对于 OOP 的认识。

(2) 重点突出，难点清晰：本书介绍了 Java SE 中网络开发所需基础知识，舍弃了 GUI 开发 (Applet、AWT、Swing) 的相关内容。内容兼顾 Java 技术的最新发展动态和实用性。

(3) 实践性强：书中最后一章应用本书所讲知识点，设计和实现了 ChatEasy 系统。读者不仅可以学会开发网络应用的方法，还能加深理解面向接口的设计方法。

书中标有星号 (“*”) 的内容为选学内容。由于学习难度较大，读者可根据自身情况选择阅读。

二、本书主要内容及建议课时

项目 章	建议课时		主 要 内 容
	理论	上机	
第 1 章	2		Java 语言的历史及其发展概况、Java 开发环境的搭建及其设置、Java 程序编译、运行及虚拟机原理
第 2 章	3	2	Java 语言编程中的主要语法，包括数据类型、运算符等
第 3 章	3	1	基本程序流程语句；if、while、for 语句的应用场景及其用法
第 4 章	4	2	面向对象的基本概念，以及如何开发面向对象的 Java 程序
第 5 章	2	2	深入介绍面向对象的主要特性，包括如何使用继承、接口和抽象类等技术
第 6 章	2	1	如何在程序中使用异常处理机制来处理异常
第 7 章	2	1	如何使用数组以及面向对象的数据结构：序列、集合和映射
第 8 章	2	1	如何读写文件，如何使用输入流和输出流，如何对流进行链接处理，如何存取配置文件

续表

项目 章	建议课时		主 要 内 容
	理论	上机	
第 9 章	2	1	介绍多线程概念, 如何设计多线程程序以提高程序的性能及如何实现线程间的通信
第 10 章	2	1	如何使用 TCP 和 UDP 套接字编写 C/S 通信程序
第 11 章	2	2	如何通过 JDBC 编写代码访问关系型数据库, 并重点介绍了几种数据库访问模式
第 12 章	4	4	利用面向对象的思路, 结合之前介绍过的 Java 技术, 特别是接口技术, 设计并实现一个简单的局域网聊天系统 ChatEasy

三、本书读者范围

- 高等院校计算机及相关专业 (课程) 的师生。
- Java 语言的初学者或没有编程经验的人。
- 有其他语言 (C/C++ 等) 开发经验, 想转向 Java 开发的人。

本书由田登山、夏自谦编著。第 2、3 章由夏自谦编写, 其他章均由田登山编写。全书由田登山负责统稿和定稿。陈明担任本书主审, 并为本书稿提出了许多宝贵的建议和修改意见。

由于时间仓促, 作者水平有限, 书中难免存在疏漏和不妥之处, 敬请广大读者批评、指正。

编 者

2011 年 10 月

目 录

CONTENTS

第 1 章 绪论	1	第 2 章 Java 语法基础	20
1.1 Java 发展概况	1	2.1 数据类型	20
1.1.1 不断扩展的计算平台	3	2.1.1 整数类型	20
1.1.2 Java 应用现状	3	2.1.2 浮点数类型	21
1.2 为什么选择 Java	4	2.1.3 布尔类型	21
1.3 Java 开发环境搭建	5	2.1.4 字符类型	21
1.3.1 安装 JDK	5	2.1.5 特殊的字符串类型	22
1.3.2 设置环境变量	9	2.2 常量与变量	22
1.3.3 检查是否安装成功	10	2.2.1 关键字	23
1.3.4 安装编辑器	10	2.2.2 常量	23
1.4 DOS 操作系统基础	11	2.2.3 变量	25
1.4.1 启动 DOS 命令提示符	11	2.3 分隔符	25
1.4.2 分区与目录	11	2.3.1 注释	25
1.4.3 显示当前工作目录下的 文件和目录	12	2.3.2 空白符	26
1.4.4 分区与目录切换	12	2.3.3 普通分隔符	26
1.4.5 可执行文件的 查找顺序	14	2.4 表达式和运算符	26
1.5 Hello World 程序设计	15	2.4.1 算术运算符	27
1.5.1 编辑源代码	15	2.4.2 关系运算符	28
1.5.2 保存代码	15	2.4.3 布尔逻辑运算符	29
1.5.3 编译代码	16	*2.4.4 位运算符	29
1.5.4 运行代码	16	2.4.5 赋值运算符	31
1.5.5 Java 虚拟机	17	2.4.6 运算符优先级	31
1.5.6 Java 工具集	17	2.5 类型转换	32
1.5.7 集成开发环境	18	2.5.1 自动转换	33
小结	18	2.5.2 强制转换	33
习题	19	小结	34
		习题	34

第 3 章 Java 流程控制	35	4.3.2 地址传递	75
3.1 顺序流程	35	4.3.3 基本类型转换为对象类型	75
3.2 条件流程	36	4.3.4 String 类型	76
3.2.1 满足条件执行	36	4.3.5 null 类型	76
3.2.2 二选一	37	4.3.6 判断是否相等	77
3.2.3 条件运算符	38	4.4 其他类型数据成员	77
3.2.4 多选一	38	4.4.1 final 数据成员	77
3.2.5 switch	42	4.4.2 static 数据成员	78
3.2.6 函数	44	4.4.3 静态方法	80
3.3 循环流程	51	小结	81
3.3.1 for 循环	51	习题	81
3.3.2 函数递归	53	第 5 章 高级面向对象编程应用	83
3.3.3 while 循环	54	5.1 包与 import	83
3.3.4 do...while 循环	55	*5.2 详解 javac、java 和 jar	85
3.3.5 循环结构对比分析	56	5.2.1 编译命令 javac	85
3.3.6 循环嵌套	56	5.2.2 字节码依赖性	86
3.3.7 break 和 continue 关键字	57	5.2.3 类库制作	89
小结	58	5.3 限定修饰符	91
习题	59	5.3.1 类修饰符	92
第 4 章 面向对象编程基础	60	5.3.2 数据成员修饰符	93
4.1 面向对象的分析	60	5.3.3 成员方法修饰符	96
4.1.1 数据抽象	60	5.4 重载	98
4.1.2 类与对象的概念	61	5.4.1 成员方法重载	98
4.2 面向对象的设计与实现	62	5.4.2 构造方法重载	99
4.2.1 设计类	62	5.5 this 关键字	100
4.2.2 定义类	64	5.6 继承	104
4.2.3 创建对象	65	5.6.1 继承的概念	105
4.2.4 访问对象的数据成员	66	5.6.2 继承的实现	107
4.2.5 使用构造方法初始化 对象状态值	67	5.6.3 继承的实质	109
4.2.6 使用成员方法模拟 对象行为	69	5.6.4 数据成员的隐藏	110
4.3 基本类型与对象类型	74	5.6.5 成员方法的覆盖	111
4.3.1 值传递	74	5.6.6 类转换	115
		5.6.7 super 的使用场合	117
		5.6.8 多态	119

5.7 抽象类与接口	121	7.2 Java 集合框架接口概述	161
5.7.1 抽象方法与抽象类	121	7.3 Collection 接口	163
5.7.2 接口	124	7.3.1 Collection 接口定义	163
5.7.3 接口与抽象类的区别	126	7.3.2 List 接口及其实现类	167
5.8 对象类与字符串处理类	128	7.3.3 Set 接口及其实现类	174
5.8.1 Object 类	128	7.4 集合内元素排序	176
5.8.2 字符串处理类	131	7.4.1 对象排序	176
5.9 Eclipse IDE 应用技巧	135	7.4.2 自定义对象的排序 规则	177
5.9.1 环境搭建	135	7.4.3 SortedSet	182
5.9.2 工作区设置	136	7.4.4 Collection 便利实现	184
5.9.3 工程管理	137	7.4.5 算法	184
5.9.4 编译与执行	140	7.5 Map 接口及其实现类	185
5.9.5 单步调试	140	7.5.1 实现类	186
5.9.6 工程备份与恢复	142	7.5.2 基本操作	188
5.10 Java Doc	143	7.5.3 批量操作	189
小结	144	7.5.4 集合视图	190
习题	145	7.5.5 集合操作	191
第 6 章 Java 异常处理机制	146	7.5.6 SortedMap	191
6.1 异常类与异常对象	146	7.6 JCF 及其启示	192
6.2 异常处理与异常处理器	147	小结	193
6.2.1 try() 语句块	149	习题	193
6.2.2 catch() {} 语句块	149	第 8 章 Java 输入/输出流	195
6.2.3 finally() 语句块	151	8.1 File 类	195
6.3 常用异常类	152	8.1.1 处理文件和目录	196
*6.4 throw 与 throws	153	8.1.2 文件处理	196
6.4.1 throw	153	8.1.3 目录处理	197
6.4.2 throws	155	8.1.4 获取最后修改时间	198
6.4.3 类库制作与使用	156	8.1.5 系统属性与文件处理	199
小结	156	8.2 字节流	200
习题	156	8.2.1 字节输入流	200
第 7 章 数组与 Java 集合框架	157	8.2.2 字节输出流	203
7.1 数组	157	8.3 字符流	209
7.1.1 一维数组	157	8.3.1 字符输入流	209
*7.1.2 二维数组	160		

8.3.2	字符输出流.....	211	*9.4	临界资源的访问	240
8.3.3	字节流与字符流的 转换	213	9.4.1	使用 synchronized 加锁	241
8.4	链接流处理类	214	9.4.2	生产者消费者问题.....	242
8.4.1	字节流类型链接流 处理类.....	214	9.5	线程管理.....	251
8.4.2	字符流类型链接流 处理类.....	221	9.5.1	守护线程	251
*8.5	配置文件读取	224	9.5.2	线程优先级	251
小结		227	*9.5.3	线程组	252
习题		228	*9.5.4	线程池	252
第 9 章	Java 多线程	229	*9.6	线程相关技术	253
9.1	线程基础	229	9.6.1	volatile	253
9.2	线程状态	231	9.6.2	ThreadLocal	253
9.2.1	由不存在状态到新建 状态	231	小结		254
9.2.2	由新建状态到就绪 状态	231	习题		254
9.2.3	由就绪状态到运行 状态	232	第 10 章	Java 套接字	255
9.2.4	由运行状态到结束 状态	233	10.1	网络基础.....	255
9.2.5	由运行状态到就绪 状态	233	10.1.1	C/S 模式与 B/S 模式	256
9.2.6	由运行状态到阻塞 状态	233	10.1.2	IP 地址.....	257
9.2.7	线程状态图	234	10.1.3	TCP/IP 协议.....	257
9.3	线程应用	234	10.2	面向连接的 Java Socket 编程	259
9.3.1	定时处理.....	234	10.2.1	建立连接	260
9.3.2	定时器 API	235	10.2.2	利用流进行通信.....	262
9.3.3	从线程外部结束 线程	237	10.2.3	C/S 通信示例.....	262
9.3.4	烧水与泡茶模拟—— join()使用示例.....	239	10.3	面向非连接的 Java Socket 编程	273
			10.3.1	构建 UDP 套接字	274
			10.3.2	使用 UDP 套接字通信 ..	275
			10.3.3	单一数据传送的 UDP 实现	275
			10.3.4	使用流改进 UDP 示例 ..	277
			小结		284
			习题		284

第 11 章	JDBC.....	285	第 12 章	ChatEasy 系统的设计	
11.1	JDBC 基础	285		与实现.....	309
11.1.1	MySQL 安装与配置	285	12.1	ChatEasy 系统问题定义.....	309
11.1.2	使用 MySQL	287	12.2	ChatEasy 系统分析与设计.....	310
11.1.3	MySQL 管理	289	12.2.1	客户端通信过程	
11.2	JDBC 应用	290		分析	310
11.2.1	获取连接.....	290	12.2.2	服务器端通信过程	
11.2.2	执行 SQL 语句	291		分析	312
11.2.3	控制数据库事务	294	12.2.3	数据封装与解析.....	313
11.2.4	PreparedStatement 与		12.2.4	面向接口的设计.....	317
	批任务	295	12.3	ChatEasy 系统代码实现.....	328
11.3	数据库访问模型	299	12.3.1	常数类定义	328
11.3.1	DAO 模型	299	12.3.2	客户端实现	328
11.3.2	ORM 模型	306	12.3.3	服务器端实现	341
11.3.3	数据库连接池模型	307	12.4	ChatEasy 系统测试	357
小结		308	小结		360
习题		308	习题		360

第

1

章

绪 论

本章学习目标

- 了解 Java 语言的历史及其发展概况
- 掌握 Java 开发环境的搭建及其设置
- 掌握常用的 DOS 命令
- 理解 Java 程序编译、运行的步骤及命令
- 理解虚拟机

作为跨平台语言, Java 语言以其简单、完全面向对象的特点受到了广大开发人员的认可。如何搭建 Java 开发环境, 如何编译、运行 Java 程序都是初学者要解决的首要问题。

本章主要介绍 Java 发展概况、Java 开发环境的搭建、Java 虚拟机、Java 程序的编译、运行步骤及相关命令。考虑到 Windows 图形用户界面理念对当代大学生产生的影响, 绝大多数 Java 的初学者对 DOS 没有任何的概念, 更谈不上通过 DOS 来辅助 Java 程序的开发了, 本章将着力介绍几个常用的 DOS 命令用于辅助编译和运行 Java 程序, 最后介绍虚拟机原理。

1.1 Java 发展概况

Java 语言来自于 Sun 公司的一个由 James Gosling 负责, 名字叫 Green 的项目小组。这个小组最初的目标是能够在诸如电视机顶盒、烤面包机、PDA 这样数字控制的电子消费产品上开发应用程序。然而消费电子产品种类繁多, 即使是同一类消费电子产品所采用的处理芯片和操作系统也不相同, 存在着跨平台的问题。当时最流行的编程语言是 C 和 C++ 语言, 开发人员就考虑是否可以采用 C++ 语言来编写消费电子产品的应用程序, 但是研究结果表明, 对于消费电子产品而言, C++ 语言过于复杂和庞大, 并不适用, 安全性也不令人满意。于是该小组就力求设计一种独立于硬件平台的计算

机语言来解决这个问题，最终设计出了一种以 C++ 语言为基础，融合了 C 和 C++ 等传统语言优点的一套面向对象的程序设计语言，并起名为 Oak（Java 语言的前身，据传是以 Gosling 窗外的一颗橡树为名）。

Oak 语言在商业应用上并未获得成功，直到 1995 年，互联网在世界上蓬勃发展，Sun 公司发现 Oak 语言所具有的跨平台、面向对象、安全性高等特点非常符合互联网的需要，于是 Sun 公司决定改进 Oak 语言的设计，将其应用于 Web 开发中，并正式注册命名为 Java。

Java 的名字来历也有一段趣闻，据传在申请注册商标时，由于 Oak（橡树）这个商标已经有人使用了，所以 Sun 公司必须重新为这个语言起一个名字。有一天，项目组几位成员在咖啡馆中边喝咖啡边讨论新名称的问题，其中一人突然发现他们喝的是一种 Java 咖啡，于是就提议叫 Java，马上得到了其他人的赞同，于是 Java 就成为了这个语言的新名称。这也就是为什么我们经常会在与 Java 相关的各种文档中看到一杯冒着热气的咖啡杯的原因。

Sun 公司用 Java 开发了一个 Web 浏览器（Hot Java），通过这个完全用 Java 语言设计的浏览器在互联网上展示出 Java 的风采。Sun 公司又决定让程序开发者免费使用 Java，这才真正地将 Java 推向了全世界。

Sun 公司在 1996 年年初首先成立了 Java Soft 组织，并在 1 月 23 日正式发布自己的 Java 1.0。作为 20 世纪业界出现的最重要的技术之一，Java 引起了编程世界的革命。直到现在，Java 仍然是互联网上最流行的语言。在 Sun 公司正式发布 Java 1.0 之后，Java 这门新生的语言就拥有了自己的会议——JavaOne，这次会议初试啼音就吸引了 600 多名参与者。除了拥有这么多的积极参与者来进行 Java 的开发之外，各大知名公司也纷纷向 Sun 申请 Java 的使用许可。一时间，NetScape、惠普、IBM、Oracle、Sybase 甚至当时刚推出 Windows 95 的微软都是 Java 的追随者。

1998 年，Java 2 平台正式发布。经过了 3 年时间的发展，Sun 公司终于知道 Java 适合干什么了。对比 Java 刚发明时的技术定位，Java 2 平台的发布算得上是有的放矢了。根据官方的文档，Java 2 是 Sun 公司意识到“一个版本不能适用所有的场合”之后，把最初的 Java 技术打包成 3 个版本的产物，也就是著名的 Java ME、Java SE、Java EE。无论今天看来当时的 Java 2 有多么的不成熟，至少经过市场和时间的检验，Java 2 规划出来的三大方向把 Java 技术指向了光明是毋庸置疑的。

Java 语言的开发者 Sun（Stanford University Network）公司，中文译为：斯坦福大学网络，起初由 Scott McNealy（原 Sun 公司首席执行官）等 4 名在斯坦福大学和加州大学 Berkeley 分校的研究生创建，1982 年 2 月正式注册，最初以工作站的设计制作为业务重点，6 个月后开始盈利；目前 Sun 公司全球雇员人数达 2 万 6 千余人。Sun 公司在 UNIX 领域一直是技术上和市场占有率的领先者，近年来又推出 Java 技术，更成为业界的标准。

北京时间 2009 年 4 月 20 日晚，甲骨文宣布，该公司将以每股 9.5 美元的价格收购 Sun 公司。该交易价值约为 74 亿美元。甲骨文将获得 Sun 公司两项软件资产——Java 和 Solaris。Java 是 IT 业

最知名的品牌之一，也是应用最广泛的技术，是甲骨文有史以来收购的最重要的软件技术。

1.1.1 不断扩展的计算平台

Java 2 平台包括：Java SE（标准版）、Java EE（企业版）和 Java ME（微缩版）3 个版本，同时这三者也构成了 SunONE（Sun Open Net Environment）体系。

Java SE（Java Platform, Standard Edition）主要用于桌面应用程序的编程，它包含那些构成 Java 语言核心的类。例如：数据库连接、接口定义、输入/输出、网络编程。

Java ME（Java Platform, Micro Edition）主要应用于嵌入式系统开发，如手机和 PDA 的编程，它包含 Java SE 中的部分类，用于消费类电子产品的软件开发。例如：智能卡、手机、PDA、机顶盒。

Java EE（Java Platform, Enterprise Edition）主要用于分布式的网络程序的开发，如电子商务网站和 ERP 系统。Java EE 包含 Java ME 中的类，并且还包含用于开发企业级应用的类。例如：EJB、Servlet、JSP、XML、事务控制。

本书只介绍 Java SE 中网络开发的部分，但是不包括图形用户界面设计，重点在于面向对象的思想、代码复用以及面向接口的编码。

1.1.2 Java 应用现状

1995 年以 Web 为主要形式的互联网正在迅速发展，Java 语言的出现以其安全、跨平台、面向对象、简单、适用于网络等显著特点迅速引起所有程序员和软件公司的极大关注。程序员们纷纷尝试用 Java 语言编写网络应用程序，并利用网络把程序发布到世界各地。包括 IBM、Oracle、微软、Netscape、Apple、SGI 等 IT 业界的大公司纷纷与 Sun 公司签订合同，要求被授权使用 Java 平台技术。微软公司总裁比尔盖茨先生也曾经研究过 Java 语言后认为“Java 语言是长时间以来最卓越的程序设计语言之一”。

到目前为止，Java 语言已经成为最流行的网络编程语言，全球大约有 400 多万软件开发商、上千万的程序员在 Java 平台上开发软件，全世界的每个主要行业领域都在使用 Java 语言，任何使用编程技术的设备、计算机和网络都在大范围地应用 Java。许多大学也纷纷开设 Java 课程，Java 语言正逐步成为世界上程序员最多的编程语言。

Java 技术的通用性、高效性、平台移植性和安全性，使之成为网络计算的理想技术。从笔记本式计算机到数据中心，从游戏控制台到科学超级计算机，从手机到互联网，Java 语言无处不在。

截至 2009 年 6 月，采用 Java 语言的设备已超过 45 亿，其中包括：

- （1）8 亿多台个人计算机。
- （2）21 亿部移动电话及其他手持式设备。

(3) 35 亿个智能卡及机顶盒、打印机、网络照相机、游戏、汽车导航系统、彩票终端、医疗设备、收费站等。

1.2 为什么选择 Java

一个庞大的专业社群每天都在测试、优化、扩展和验证 Java。开发者队伍达到惊人的数量——650 多万。这是全球最大的、最具活力的开发团队。凭借其卓越的通用性、高效性和移植性,Java 对开发者具有不可估量的价值,使他们可以:

- (1) 在一个平台上编写软件,然后即可在几乎所有其他平台上运行。
- (2) 创建可在 Web 浏览器和 Web 服务中运行的程序。
- (3) 开发适用于在线论坛、存储、投票、HTML 格式处理及其他用途的服务器端应用程序。
- (4) 将采用 Java 语言的应用程序或服务组合在一起,形成高度定制的应用程序或服务。
- (5) 为移动电话、远程处理器、低成本的消费产品及其他任何具有数字核心的设备编写强大而高效的应用程序。

Java 语言的特点主要表现在以下几个方面。

1. 简单

Java 语言来源于 C++, 其大部分语法结构都与 C++ 相似, 因此一般熟悉 C++ 语言的编程人员学习起来非常容易。同时 Java 所具有的自动内存管理机制也大大简化了 Java 程序设计开发。与 C++ 相比, Java 去掉了很多容易混淆和较少使用的特性, 增加了内存空间自动垃圾收集功能, 使开发工作变得简单而可靠。

2. 平台无关性

Java 运行于虚拟机 (JVM) 上, 实现了不同平台之间的 Java 接口, 因此 Java 开发的程序可以运行于不同的系统平台。Java 虚拟机建立在硬件和操作系统之上, 用于实现 Java 字节码 (Byte Code) 的解释和执行等功能, 为不同系统平台提供统一的 Java 接口。Java 字节码是一种近似于机器码的中间码, 不受计算机硬件设备和操作平台种类的限制, 只要计算机中有 Java 运行的环境, Java 字节码就可以在其上运行, 这也正是 Java 最为突出的特征: 一次编译, 随处运行。配合严格规定的数据类型和类结构标准, 使 Java 编写的程序具有非常良好的可移植性。

3. 面向对象

Java 是一种面向对象的程序语言, 吸收了 C++ 面向对象的概念, 将数据封装于类中, 利用类的优点, 便于维护且保证了程序的简洁。用 Java 语言设计程序时, 以面向对象的方式来思考, 程序员只需要把主要精力用在类和接口的设计及应用上即可, 而不是专注于程序流程上。

Java 提供了众多一般对象的类, 通过继承即可使用父类的方法。与 C++ 不同的是, 在 Java 中类的继承关系是非多重的, 一个子类只有一个父类。

4. 安全性

Java 的设计目的是提供一个用于网络/分布式的计算环境。因此, Java 语言特别强调安全性。

从内存管理的角度看,一方面 Java 不允许用户利用程序强制对内存进行存取操作,即对程序员来说内存分配是透明的,这种机制使内存出错的几率大大减小。另一方面,Java 编译器也没有内存分配的决定权,而是通过系统所在的软硬件平台来决定。同时 JVM 在解释 Java 字节码的同时会对每段代码进行安全检测,对所有不合法的数据和无权限的存取操作加以阻隔,所以对于不合法的字节码是无法被解释和执行的。

5. 解释执行

Java 与 C++最大的不同点在于 Java 是一种半编译半解释型的语言。用 Java 编写的程序文件都先被编译成字节码,再通过 JVM 解释成为机器码,而不像 C++编写的程序一经编译即成为机器码。

Java 源程序被编译后所形成的字节码被设计成接近真正的汇编语言机器码,因此解释器在解释字节码时速度会相当快。虽然执行速度还是比不上一次性编译,但以牺牲较小的执行速度换取出色的移植性和稳定性显然是值得的。

Java 的解释器可以直接在任何已经移植解释器的计算机上解释和执行 Java 字节码。但需要注意,在高版本环境下编译的 Java 字节码到低版本解释器环境下运行有时会产生问题。

6. 多线程

Java 的多线程 (Multithreading) 机制保证程序可以并行运行。多线程让程序同时拥有数个线程来运行不同的工作,如当一个线程处理计算工作时,另一个线程可同时处理输出操作,而不需要采用全局的事件循环机制。使用 Java 开发出来的多线程程序比传统的单一线程程序所占用的 CPU 资源少。

7. 动态

Java 的设计使其适应于不断发展的环境。在类库中可以自由地加入各种新的类和方法以适应新的环境要求,且不会影响用户程序地执行。

1.3 Java 开发环境搭建

学以致用,编程语言的学习切忌纸上谈兵,Java 也不例外。为了能够更好的学习 Java,首先需要安装 Java 的开发环境。本节以当前最新版本的 JDK 1.6.0 为例,介绍如何搭建开发环境,内容包括 JDK 的下载和安装,如何设置相关的环境变量,以及检查 JDK 是否安装成功。

1.3.1 安装 JDK

1. 获取 JDK 1.6.0

读者可以从 Oracle 公司的官方网站下载(Oracle 公司收购 Sun 公司后对其官方网站进行了整合)