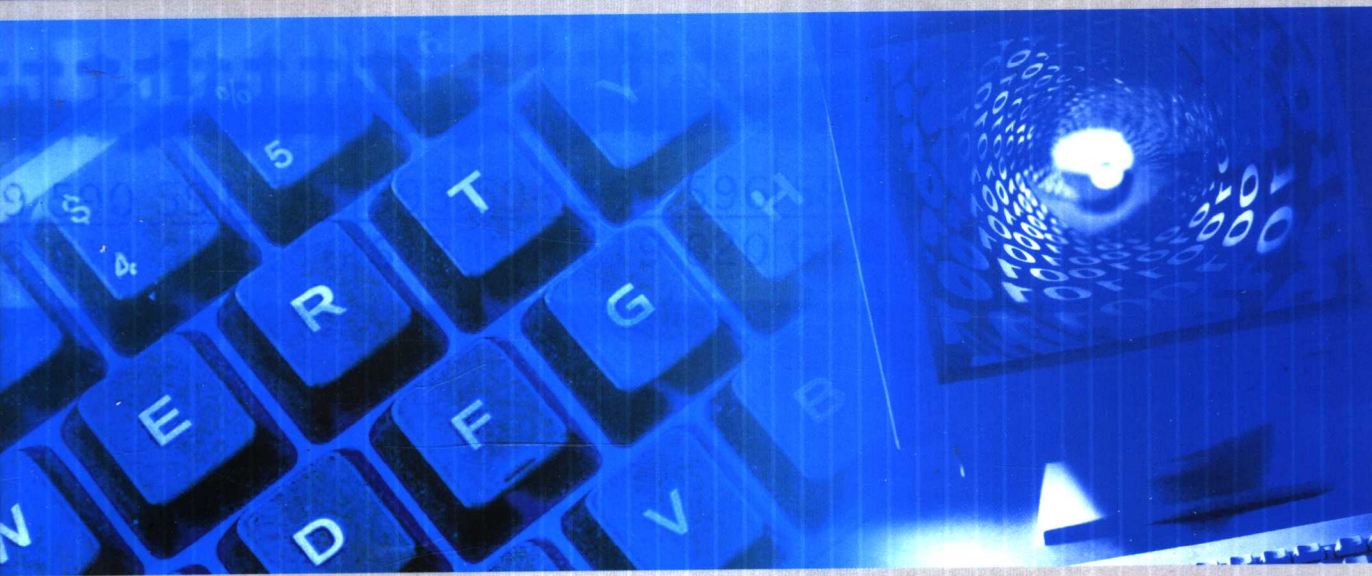




21 世纪全国高校应用人才培养信息技术类规划教材



Java语言程序设计

周少琦 李铁兵 董 东 编著



北京大学出版社
PEKING UNIVERSITY PRESS

21 世纪全国高校应用人才培养信息技术类规划教材

Java 语言程序设计

周少琦 李铁兵 董 东 编著



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书是一本实用教材,突出实际应用,围绕实际应用选材、决定繁简和安排先后。全书总体上可分为三部分:第一部分为基础部分,介绍了Java语言语法基础和面向对象理论基础;第二部分为工具部分,介绍了Java语言类库包中的语言包、实用包和输入输出包;第三部分为实用部分,介绍了Java语言图形用户界面、数据库、网络和多媒体程序设计。实用部分超过全书的一半,大量例程实际可用。

本书又是一本简明教材,全书系统性极强,各章节安排合理、相互连接、前因后果、循序渐进;全书共188个小节,每小节介绍一个问题(一个类或一个接口),语言叙述通顺、流畅、精确、恰当;全书共220个实例程序,与小节内容贴切且短小、简练、完善、齐全。看简明问题、用实例和计算机对话,将使学习过程变得轻松愉快。全书129道问答思考题和249道程序设计题全部做了解答,可在渤海大学信息学院网站上免费下载,特别是指定为综合性实验的程序设计题都是课程设计的最佳题材。

本书既可作为高等院校计算机及相关专业的教材,又可以作为计算机技术爱好者的自学用书,还可作为软件开发人员的参考资料。

图书在版编目(CIP)数据

Java语言程序设计/周少琦,李铁兵,董东编著. —北京:北京大学出版社,2007.8

(21世纪全国高校应用人才培养信息技术类规划教材)

ISBN 978-7-301-12283-9

I. J… II. ①周… ②李… ③董… 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080644号

书 名: Java语言程序设计

著作责任者: 周少琦 李铁兵 董东 编著

责任编辑: 温丹丹 沈 欣

标准书号: ISBN 978-7-301-12283-9/TP·0871

出版者: 北京大学出版社

地 址: 北京市海淀区成府路205号 100871

电 话: 邮购部 62752015 发行部 62750672 编辑部 62765126 出版部 62754962

网 址: <http://www.pup.cn>

电子信箱: xxjs@pup.pku.edu.cn

印刷者: 北京大学印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

787毫米×980毫米 16开本 21印张 478千字

2007年8月第1版 2007年8月第1次印刷

定 价: 35.00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; 电子信箱: fd@pup.pku.edu.cn

前 言

Java 语言从 1995 年正式发布以来,就以它的面向对象、平台无关、安全可靠、多线程以及强大的网络功能赢得了广泛的赞誉,微软公司总裁比尔·盖茨就高度评价说:“Java 语言是长期以来最卓越的计算机程序设计语言。”今天,Java 语言已不仅仅是一门计算机程序设计语言,而是发展成为影响一个国家经济发展、科学进步的多领域的先进技术。因此,搞好 Java 语言的教学工作、推广 Java 语言的技术应用已经成为计算机专业教师责无旁贷的历史重任。

当今的世界上,人口增长的速度超过了经济增长的速度,就业难已成为全球普遍存在的问题之一,Java 语言技术的广泛应用将为这一问题的局部解决开辟一个新的途径。近两年,我国用 Java 语言技术开发应用程序的公司纷纷建立,各公司都急需高水平、实干的 Java 语言技术人员。因此,搞好 Java 语言的教学工作、为我们的学生奉献一把开启未来美好生活的金钥匙也已经成为计算机专业教师不可推卸的分内职责。

基于以上两点考虑,多年来我们不断总结教学经验,认真听取学生们的反馈意见,广泛收集资料、择优选取、反复加工、不断完善,终于使本教材拥有如下一些主要特点。

1. 理顺知识:本教材做到每节内容只涉及本节或前述章节的知识点,而不涉及后续章节的知识点,更未涉及全教材都未讲到的知识点。这一点显著地体现在对小应用程序的处理上,当前已出版的教材多数开篇就介绍小应用程序,由于小应用程序既涉及图形用户界面的知识,本身又有相对复杂的设计和运行规律,所以把它放在前面造成学生难于接受。此外,关于输入方法,许多教材放在异常前面介绍,也造成了学生们的不少疑惑。将这两部分内容相对放后,在多年的使用中,学生们一直反映本教材简单、明了、易学、易懂。

2. 循序渐进:根据学生们的反馈意见,我们将有难度的两个问题从细分解,再按照从简到繁、由浅入深的顺序安排教学内容。第一个是类的定义,我们先是给出最简格式,在面向对象理论一章中不断提出新知识、介绍新概念分节完善类定义的格式。第二个是图形用户界面的程序设计,我们结合事件处理程序设计,提出了主方法、构造方法和继承顶层容器式三种程序设计结构,结合具体的设计示例讲清变动的缘由、给出格式的名称既便于学生们的理解,又便于学生们的记忆。

3. 面向实用:本教材有关基础部分内容的筛选是以后续应用需要为准则,最大限度地压缩基础部分的篇幅,相对地充实和完善了数据库、网络、多媒体的应用内容。这三大应用的内容不仅各成体系,而且联系实际可以直接付之实用,如果把图形用户界面看作应用基础的话,那么实际应用的内容就占全教材篇幅的二分之一以上,这一点历年来也深受学生们的欢迎,极大地调动了学生们学习的积极性。

4. 系统性强:本教材除第 1 章语法基础、第 2 章面向对象理论外,其余 7 章基本上按照常用的 Java 类库包自然分章,在每章内先简要介绍全章(包)概貌,然后按照应用的主次安排讲解的详略,既保证知识的系统性,又突出应用的重点性。值得一提的是第 2 章,由于面向对象理论的内容十分丰富,所以在取材及内容的系统性上颇费斟酌,本教材则立足于应用仅将其中最基本的内容概括为类、对象、继承、接口和包 5 个问题并使其环环相扣、承接

无暇。全书各章的安排也遵循系统性的原则，前基础、中工具、后应用，便于学生们的理解、记忆和应用。

5. 示例简明：本教材中每小节都有程序示例，全教材共 220 个示例，在示例中只反映本小节的知识内容，以便将学生们的注意力集中在新的知识点上，后面的知识点则绝不涉及。这些示例短小、简练、齐全、完善，都是作者经过多年反复推敲、反复改进编写的，并且都尽量联系学生们的生活实际使其富有生动性和趣味性。

6. 习题完善：Java 语言程序设计是一门实践性极强的学科，因此要学好这一门课程，真正掌握这一门技术，除了需要认真理解相关的理论、概念和方法外，还需要大量的上机实践。为此，在每章后都配有足够的问答思考题和程序设计题，全书 129 道问答思考题和 249 道程序设计题全部做了解答，特别是指定为综合性实验的程序设计题都是课程设计的最佳题材，它们比教材中的示例还要丰富、完善和实用。本教材中的示例程序、习题解答和教学课件都可在网站上免费下载，具体的地址是：<http://www.bhu.edu.cn/page/depart/xxxy/gzls/java.asp>。

7. 分类实验：本教材采取在每章后对实验内容进行简单分类的做法：对于那些对相应章节内容掌握不够透彻的学生，可以把教材中的实例拿来上机运行，这称之为验证性实验；对于那些对相应章节内容掌握较好的学生，可以按大多数程序设计题的要求自己编写源程序上机运行，这称之为设计性实验；而对于那些学有余力的学生，则可以把指定为综合性实验的程序设计题作为课程设计自行编写、调试、运行，这必将收到喜人的效果。

8. 篇幅紧凑：本教材除了尽量使语言简练而外，所有示例程序都采用一行多句的格式，而且每一个层次仅缩进两个字符，同时将运行结果略去。这样做肯定给教材的阅读带来不便，不过我们在网上提供的源程序则都是一行一句，每层 4 个字符缩进，都有运行结果，稍复杂的程序还有较详细的注释。另外，教材最初设计时每章后面都有综合实例，此次编排则将绝大多数综合实例作为习题安排在各章的最后，并指定为综合性实验。

Java 语言技术的特点可以概括为 8 个字“无所不能、如诗如画”。说它“无所不能”，不是专指本教材所限定的内容，而是包括在此基础上发展起来的多领域应用技术。说它“如诗如画”读者可以在本教材中领略它的美妙。第 5 章足可以让人诗兴大发；第 6 章可以初见友好画面；第 7 章可以把纷繁的数据调理得有条不紊；第 8 章可以遨游在 Internet 的广阔天地；第 9 章就可以欣赏优美的图画、聆听动人的乐曲，读者在如诗如画般美丽的境界里走向辉煌的明天。

本教材由周少琦总体规划，并担任统稿工作，第 1~2 章由董东编写；第 3~5 章由李铁兵编写；第 6~9 章由周少琦编写。在编写的过程中，作者力求精益求精、尽善尽美，但是由于水平有限、时间仓促，欠妥、疏漏甚至错误的地方都在所难免，殷切期望广大师生及读者批评指正，联系邮件可寄到 zh_shqi1116@163.com。

在本教材的编写过程中得到了渤海大学信息科学与工程学院领导和老师们的大力支持和热情帮助，特别是沈泽刚老师帮助解决了不少疑难问题，在此一并表示衷心的感谢。本教材的出版还要感谢北京大学出版社温丹丹编辑的精心校对和成功润色，正是她的勤奋和敬业才使本教材以完满的形象与读者见面。

周少琦

2007 年 5 月 10 日 于渤海大学

目 录

第 1 章 Java 语言语法基础	1
1.1 Java 语言概述	1
1.1.1 Java 语言的对象和类	1
1.1.2 第一个 Java 语言程序	2
1.1.3 Java 语言源程序的编译和运行	4
1.2 基本数据类型	7
1.2.1 标识符、变量和常量	7
1.2.2 基本数据类型	8
1.2.3 各类型数据间的相互转换	10
1.3 运算符及表达式	11
1.3.1 赋值运算符	12
1.3.2 算术运算符和算术表达式	12
1.3.3 关系运算符和关系表达式	13
1.3.4 逻辑运算符和逻辑表达式	13
1.3.5 位运算符和位运算表达式	14
1.3.6 条件运算符与条件表达式	14
1.3.7 扩展赋值运算符	15
1.3.8 运算符的优先级和表达式语句	15
1.4 程序流程控制语句	16
1.4.1 选择控制语句	16
1.4.2 循环控制语句	19
1.4.3 转向控制语句	22
1.5 数组	24
1.5.1 一维数组的定义	24
1.5.2 一维数组元素的引用	24
1.5.3 二维数组的定义	25
1.5.4 二维数组元素的引用	26
1.6 方法	26
1.6.1 方法的定义和调用	27

1.6.2	方法的按值调用和引用调用	29
1.6.3	方法应用的程序设计	30
1.6.4	方法的重载	31
1.7	复习题	31
1.7.1	问答思考题	31
1.7.2	程序设计题	32
1.8	上机实验	34
1.8.1	验证性实验	34
1.8.2	设计性实验	34
1.8.3	综合性实验	34
第 2 章	面向对象理论基础	35
2.1	类与对象	35
2.1.1	类的定义	35
2.1.2	实例化对象	36
2.1.3	构造方法	37
2.1.4	构造方法的重载	38
2.1.5	类变量和类方法	38
2.2	对象的使用	40
2.2.1	调用对象的成员	40
2.2.2	对象做类的成员	41
2.2.3	方法中使用对象	42
2.2.4	对象数组	43
2.2.5	this 对象	43
2.3	类与子类	45
2.3.1	类的继承	45
2.3.2	变量的隐藏和方法的覆盖	45
2.3.3	super 对象	46
2.3.4	私有变量和私有方法	47
2.3.5	父类对象和子类对象	48
2.3.6	抽象类和最终类	50
2.4	类与接口	52
2.4.1	接口的定义与实现	52
2.4.2	一个类实现多个接口	53
2.4.3	接口的多重继承	54
2.5	类与包	55
2.5.1	包的创建	56

2.5.2	包的使用	57
2.5.3	访问权限修饰符	59
2.5.4	Java 类库包概述	62
2.6	复习题	63
2.6.1	问答思考题	63
2.6.2	程序设计题	64
2.7	上机实验	66
2.7.1	验证性实验	66
2.7.2	设计性实验	66
2.7.3	综合性实验	66
第3章	Java 类库语言包	67
3.1	异常类	68
3.1.1	异常类概述	68
3.1.2	运行时异常	69
3.1.3	异常捕获	70
3.1.4	异常声明	71
3.1.5	异常类的方法	72
3.1.6	自定义异常	73
3.2	系统运行类	75
3.2.1	根类 Object	75
3.2.2	类描述类 Class	76
3.2.3	系统类 System	78
3.3	数学运算类	81
3.3.1	基本数据类型包装类	81
3.3.2	数学运算类 Math	84
3.4	字符串类	86
3.4.1	String 类	86
3.4.2	StringBuffer 类	89
3.4.3	main 方法及命令行参数	92
3.5	复习题	93
3.5.1	问答思考题	93
3.5.2	程序设计题	94
3.6	上机实验	95
3.6.1	验证性实验	95
3.6.2	设计性实验	95

3.6.3 综合性实验	95
第 4 章 Java 类库实用包	96
4.1 通用类	97
4.1.1 日期类 Date	97
4.1.2 日历类 Calendar	97
4.1.3 格列高里历类 GregorianCalendar	98
4.1.4 字符串标记类 StringTokenizer	99
4.2 集合类	100
4.2.1 聚集接口 Collection	101
4.2.2 集合接口 Set	101
4.2.3 散列集合类 HashSet	102
4.3 列表类	103
4.3.1 列表接口 List	103
4.3.2 数组列表类 ArrayList	104
4.3.3 向量类 Vector	105
4.3.4 链接列表类 LinkedList	107
4.4 遍历接口	108
4.4.1 通用迭代器接口 Iterator	108
4.4.2 列表迭代器接口 ListIterator	109
4.4.3 枚举接口 Enumeration	110
4.5 映射类	110
4.5.1 映射接口	111
4.5.2 散列映射类 HashMap	111
4.5.3 散列表类 Hashtable	112
4.6 复习题	114
4.6.1 问答思考题	114
4.6.2 程序设计题	114
4.7 上机实验	115
4.7.1 验证性实验	115
4.7.2 设计性实验	115
4.7.3 综合性实验	116
第 5 章 Java 类库输入/输出包	117
5.1 文件类	118
5.1.1 文件的创建与删除	118
5.1.2 文件名称与路径的操作	119

5.1.3	文件属性的操作	120
5.1.4	文件过滤器的应用	121
5.2	字节基本输入/输出流类	122
5.2.1	字节输入/输出流类概述	122
5.2.2	文件输入/输出流类	124
5.2.3	字节数组输入/输出流类	126
5.2.4	串行输入流类	128
5.3	字节过滤输入/输出流类	129
5.3.1	数据流 DataStream	130
5.3.2	缓冲流 BufferedStream	132
5.3.3	打印流 PrintStream	133
5.4	字符输入/输出流类 Reader/Writer	133
5.4.1	字符输入/输出流类概述	134
5.4.2	输入流读类和输出流写类	135
5.4.3	文件读类、文件写类、缓冲读类和缓冲写类	136
5.4.4	打印写类	138
5.4.5	管道流类 PipedStream	138
5.5	随机存取文件流类	140
5.6	对象串行化	141
5.6.1	对象的串行化	141
5.6.2	对象的反串行化	142
5.6.3	对象串行化示例	142
5.7	复习题	143
5.7.1	问答思考题	143
5.7.2	程序设计题	143
5.8	上机实验	145
5.8.1	验证性实验	145
5.8.2	设计性实验	145
5.8.3	综合性实验	145
第 6 章	图形用户界面类库包	146
6.1	AWT 类库包简介	146
6.1.1	组件类 Component	146
6.1.2	容器类 Container	148
6.1.3	颜色类 Color	148
6.1.4	窗口 Window 和标签 Label	149

6.1.5	框架 Frame 和按钮 Button	150
6.1.6	面板 Panel 及其嵌套	152
6.1.7	布局管理器类 LayoutManagers	152
6.2	GUI 的事件处理模型	156
6.2.1	事件类 Event	156
6.2.2	事件监听器接口	157
6.2.3	委托事件模型的程序设计和窗口事件	159
6.2.4	一个监听器类实现多个监听器接口和行动事件	160
6.2.5	事件适配器和容器事件	161
6.2.6	内部类和组件事件	163
6.2.7	匿名类和焦点事件	165
6.2.8	画布、鼠标事件、键盘事件和自监听	166
6.3	Swing 类库包常用容器	168
6.3.1	框架 JFrame 和标签 JLabel	169
6.3.2	对话框 JDialog	171
6.3.3	面板 JPanel	172
6.3.4	滚动窗格 JScrollPane 和图像图标 ImageIcon	173
6.3.5	选择窗格	174
6.4	Swing 类库包常用控件	177
6.4.1	抽象按钮 AbstractButton 和按钮 JButton	177
6.4.2	肘节按钮 JToggleButton 和广播按钮 JRadioButton	179
6.4.3	校验盒 JCheckBox 与项目事件	180
6.4.4	文本控件及其应用	182
6.4.5	菜单控件及其应用	186
6.4.6	滚动条与调整事件	189
6.4.7	进度条与变动事件	191
6.4.8	工具条	192
6.5	Swing 类库包高级控件	194
6.5.1	组合盒 JComboBox	194
6.5.2	列表 JList	196
6.5.3	表格 JTable	199
6.6	复习题	205
6.6.1	问答思考题	205
6.6.2	程序设计题	205
6.7	上机实验	207

6.7.1 验证性实验	207
6.7.2 设计性实验	207
6.7.3 综合性实验	207
第 7 章 数据库程序设计	208
7.1 简单数据库程序设计	208
7.1.1 创建数据库	209
7.1.2 选配 ODBC 数据源	210
7.1.3 加载数据库驱动程序	210
7.1.4 创建连接对象	211
7.1.5 创建声明对象	211
7.1.6 获得结果集	212
7.1.7 解析返回结果	213
7.2 数据库操作程序设计	214
7.2.1 创建表格语句、插入操作语句和删除表格语句	215
7.2.2 修改表格语句、更新操作语句和删除操作语句	216
7.2.3 查询操作语句	217
7.3 SQL 多接口的应用	219
7.3.1 数据库元数据接口	219
7.3.2 结果集元数据接口	221
7.3.3 预编译声明接口	222
7.3.4 可调用声明接口	226
7.3.5 可滚动的结果集	227
7.3.6 可更新的结果集	229
7.4 图形用户界面数据库程序设计	230
7.4.1 表示层程序设计	231
7.4.2 业务层程序设计	232
7.4.3 逐条记录显示事件处理设计	234
7.4.4 数据层程序设计	235
7.5 学生成绩管理信息系统	236
7.5.1 表格界面程序设计	237
7.5.2 事件处理程序	238
7.5.3 数据库管理程序	240
7.6 复习题	241
7.6.1 问答思考题	241
7.6.2 程序设计题	242

7.7	上机实验	242
7.7.1	验证性实验	242
7.7.2	设计性实验	242
7.7.3	综合性实验	243
第 8 章	网络程序设计	244
8.1	多线程设计	244
8.1.1	线程的创建与运行	245
8.1.2	线程的状态	247
8.1.3	线程的优先级	248
8.1.4	线程的共享与同步	249
8.2	网络程序设计基础	251
8.3	网络地址类程序设计	254
8.3.1	统一资源定位器类	254
8.3.2	统一资源定位连接类	256
8.3.3	互联网地址类	258
8.4	TCP 协议的程序设计	259
8.4.1	套接口类和服务器套接口类	259
8.4.2	服务器方程序设计	260
8.4.3	客户机方程序设计	262
8.4.4	多客户机的程序设计	262
8.5	UDP 协议的程序设计	264
8.5.1	数据报套接口类和数据报包类	264
8.5.2	服务器方程序设计	265
8.5.3	客户机方程序设计	266
8.5.4	广播通信程序设计	267
8.6	基于客户机/服务器模式聊天室的建立	268
8.6.1	服务器方程序	269
8.6.2	客户机方程序	270
8.7	复习题	271
8.7.1	问答思考题	271
8.7.2	程序设计题	271
8.8	上机实验	272
8.8.1	验证性实验	272
8.8.2	设计性实验	272
8.8.3	综合性实验	273

第9章 多媒体程序设计	274
9.1 小应用程序简介	274
9.1.1 小应用程序的生命周期及主要方法	275
9.1.2 小应用程序的运行	276
9.1.3 小应用程序参数的传递	278
9.2 绘图与文字程序设计	279
9.2.1 Graphics 类绘图	280
9.2.2 Graphics2D 类绘图	281
9.2.3 文字输出	284
9.3 图像程序设计	287
9.3.1 图像的加载与显示	287
9.3.2 图像观察器接口	289
9.3.3 媒体跟踪器类	291
9.3.4 创建图像	292
9.3.5 抓取像素	295
9.3.6 图像处理	297
9.4 声音程序设计	299
9.4.1 小应用播放声音	300
9.4.2 采样包播放声音	301
9.4.3 采样包录制声音	304
9.4.4 乐器数字接口 MIDI 播放乐曲	307
9.4.5 乐器数字接口 MIDI 合成乐曲	309
9.5 动画的程序设计	311
9.5.1 多文字的打字输出	312
9.5.2 显示卷动的画面	312
9.5.3 绘制变动的图画	314
9.5.4 播放连续的图像	316
9.6 复习题	317
9.6.1 问答思考题	317
9.6.2 程序设计题	317
9.7 上机实验	318
9.7.1 验证性实验	318
9.7.2 设计性实验	318
9.7.3 综合性实验	318
参考文献	319

第 1 章 Java 语言语法基础

本章教学目标

- 掌握 Java 语言程序的运行环境和源程序的编辑、编译、运行的方法
- 掌握 Java 语言的标识符、变量、常量、基本数据类型及其相互间的转换
- 掌握 Java 语言的运算符、表达式及运算符的优先级
- 掌握 Java 语言的程序流程控制语句
- 掌握 Java 语言数组的定义和引用
- 掌握 Java 语言方法的定义、调用及方法的重载

Java 语言语法基础介绍的是 Java 语言程序中最基本的组成成分和程序结构,这一部分内容与所有计算机程序设计语言都相类似,包括数据类型、运算符、表达式、流程控制语句和数组。此外,1.1 节对 Java 语言做了简要的概述,主要给出一个最简程序以说明 Java 语言程序编辑、编译和运行的方法和过程。1.6 节介绍了面向对象语言特有的方法,这一方面是因为它与面向过程语言的函数十分相似,另一方面是为了减轻第 2 章介绍面向对象理论的难度。基础的东西有些简单,但它们又是特别重要,掌握好基础的东西,后面的学习就会一帆风顺。

1.1 Java 语言概述

1.1.1 Java 语言的对象和类

Java 语言是一种新型的面向对象的程序设计语言,由于 Java 语言具有跨平台、可移植、安全、简单以及分布式处理的能力,使其在 Internet 应用软件的开发上首屈一指,因而得到了迅速的发展和广泛的应用。

面向对象的程序设计语言是计算机程序设计语言发展至今的最高一个技术层次,它是相对于面向机器程序设计语言和面向过程程序设计语言而言的,它代表了一种全新的思维模式。所谓对象指的是人类行动或思考时作为目标的人或事物,所谓面向对象就是把客观存在的人或事物作为计算机程序设计的目标。在人们正常的逻辑思维中对每一个客观存在的目标所考虑的不外乎是他具有的特征以及可能的行为。例如,我们想到“赵一”这个学生,他就成为我们的一个对象;根据涉及到的问题我们可能想到他的特征,诸如:年龄、性别、身高及各科成绩等;想到他的行为,诸如:选课、听课、做作业及考试等。

Java 语言就是模仿人类的思维方法,将问题中涉及到的客观事物抽象为特征和行为的逻辑统一体称之为对象。在 Java 语言中用变量表示客观事物的特征,用方法表示客观事物的行为,用对象名来唯一地标识,所以说 Java 语言中的对象就是表示客观事物特征和行为的一组变量和相关方法的集合。例如,我们可以用 20 表示赵一的年龄,“男”表示赵一的性别,用 100 表示赵一的某科成绩,用回答问题的方法表示赵一考试的行为,用“赵一”这个标识符

表示赵一这个对象。

类是面向对象的程序设计语言中又一重要的基本概念，物以类聚、人以群分这是人们常说的一句话，它表明了“类”就是许多相同或相似的事物的综合。这里所说的“相同或相似”指的是不同事物具有相同或相似的特征和行为，这里的综合则是一种逻辑上的抽象。例如，我们只能看到一个个具体的学生，名字可能为“赵一”、“王二”等，但我们看不到一个没有具体姓名、年龄等的学生。然而我们却可以从成千上万不同层次的学生中抽出他们共同的特征和行为加以综合，从而形成学生这个概念，这就是学生“类”。

Java 语言中的对象是一组变量和相关方法的集合，由对象抽象出来的类就成为一种封装了变量和方法的复合数据类型。类是 Java 语言中一种最基本的复合数据类型，当需要研究某问题中的某一具体对象时，就用类实例化出一个对象来，这样实现了代码的复用大大提高了程序设计的效率。这很像工厂的铸造工作，类就像是一个铸模，实例化就像是铸造过程，而它的铸件就是对象。

类的概念在 Java 语言中是一个相当重要而又应用十分广泛的概念，在 Java 语言中不仅把具有相同或相似的特征和行为的对象抽象为类，而且还把具有某种功能的一组变量和方法的集合也称为类，由此也特别明显地突出类仅仅是变量和方法的集合这一典型特征。我们把反映客观世界人或事物的类称为模型类，而把具有某种功能的一组变量和方法的集合称为功能类，本书第 1 章中的类都属于功能类。

1.1.2 第一个 Java 语言程序

Java 语言是一种准解释型的高级编程语言，所以 Java 语言程序的开发通常需要编写源程序、编译生成字节码文件和解释运行 3 个过程。本节我们将通过一个简单而完整的例子来说明 Java 语言源程序的基本格式、基本内容与基本要求。

1. 编写源程序

Java 语言的开发工具 JDK (Java Developer's Kit, Java 开发者工具包) 没有提供源代码的编辑工具，我们可以使用文本编辑软件来编辑源程序，如使用 Windows 的记事本，或使用 DOS 下的 Edit (但不要使用 WPS 2000 或 Word 2000 这类排版软件，Java 语言源程序中不允许出现排版格式)。利用这些文字编辑软件将源程序录入到计算机中，形成一个文件保存起来。值得注意的是，无论使用哪一种编辑软件来创建 Java 语言源程序，最后都要保存为“文件名.java”的形式，并且保存的格式必须是普通的文本文件。对于隐藏默认保存格式为*.txt 的文件为消除扩展名.txt，Windows 98 需进入系统菜单“查看”，而 Windows 2000 需进入系统菜单“工具”，选中“文件夹选项”，在文件夹选项的窗口内再选“查看”标签，在所列诸项中有“隐藏已知文件类型的扩展名”一项，单击方框消除对号即可。下面就是用 Windows 的记事本编写的第一个 Java 语言程序的源代码。

例 1.1 第一个 Java 语言源程序：MyGreeting.java

```
class MyGreeting                                // 定义一个类
{
    public static void main(String args[])      // 定义一个方法
    {
        System.out.println("Good morning class!"); // 可执行语句
    }
}
```

```
}  
}
```

2. 第一个 Java 语言源程序说明

Java 语言程序都是由类组成的,组成 Java 语言程序的类都需要定义。Java 语言中类的定义分为两个部分,即类头的声明和类体的定义。本例中的第一行就声明了一个类,名称为 `MyGreeting`,类头声明的标志是关键字 `class`,在 `class` 和类名之间要有一个空格隔开。

本例中的第 2 行~7 行定义了类体,Java 语言的类体是由变量和方法组成的,实际应用的类中都有多个变量和多个方法,这里只是第一个最简示例,故只定义了一个方法。类体中的变量和方法需要用大括号{和}括起来,标志着一个类定义的开始和结束。

本例中的第 3~6 行定义了一个方法,Java 语言中方法的定义也分为两个部分,即方法头的声明和方法体的定义。方法头声明的标志是方法名后紧挨着一对小括号,小括号内是调用参数,方法名前是方法的返回值类型标志和方法应用属性的修饰符,方法名、返回值类型和属性修饰符之间也要用空格隔开。

本例中声明的方法名称为 `main`,这是 Java 语言应用程序中必需定义且只能定义一个的方法,所以这个最简程序就只能定义这个方法。这个方法有它固定的声明格式必需严格遵照执行,本例就是按照要求的格式书写的。第 1 个属性修饰符为 `public`,它表示 `main` 方法允许其他类访问。第 2 个属性修饰符为 `static`,它表示 `main` 方法是一个静态方法,有关静态方法的详细介绍留在第 2 章。第 3 个是返回值类型标志, `void` 说明它运行后不返回任何数值。小括号中的参数是字符串数组,这是留给执行程序时在命令行中输入参数的,这一点将在第 3 章中予以介绍。

本例的第 4~6 行定义了 `main` 方法的方法体,Java 语言的方法体是由变量和可执行语句组成的,这些变量和可执行语句共同完成方法中预定的功能,这些变量和可执行语句也要用大括号{和}括起来,括号内的每一语句都要用分号结束。在这个最简示例中没有涉及变量,所以也就没有变量的声明和应用。

本例的第 5 行是一条执行输出操作的语句,也是本程序唯一一条可执行语句,这个操作是由 `System` 类的成员对象 `out` 的方法 `println()` 来完成的。有关 `System` 类、`out` 成员对象、`println()` 方法的详细内容将在第 3 章和第 5 章中讲到,这里只需记住 `println()` 方法向屏幕输出一行字符并回车,这行字符就是 `println()` 方法括号里用双引号括起的字符串。语句的末尾是用“;”结束的,分号是 Java 语句中不可缺少的一部分。

在编写 Java 语言源程序的时候,需要注意如下要求。

(1) Java 语言是区分大小写的语言,关键字的大小写不能搞错,如果把 `class` 写成 `Class` 或者 `CLASS` 都会导致错误。

(2) Java 语言源程序每一语句都要用分号结束,一行可以书写多条语句,每条语句都要用分号结束。例 1.1 为了便于分析,一条语句及大括号都独占一行,后面为了缩小篇幅则一律采用一行多句。

(3) 源程序编写好之后,要以文件的形式保存在硬盘或软盘上,称为源程序文件,或称为源文件。这里需要注意,这个源文件的名字不能随便取,它必须与程序的主类名相一致,所谓主类是指包含有 `main` 方法的类。一个 Java 应用程序可能有多个类,但是由于在一个 Java 应用程序中必需定义且只能定义一个 `main` 方法,所以一个 Java 应用程序中的主类也就唯一