

GENERAL  
EDUCATION

高等学校通识教育系列教材

# Java技术及应用

(第2版)

赵锐 李卫华 编著



清华大学出版社



GENERAL  
EDUCATION

高等学校通识教育系列教材

---

# Java技术及应用

## (第2版)

赵锐 李卫华 编著

清华大学出版社  
北京

## 内 容 简 介

全书共分 12 章。第 1 章主要介绍了 Java 的语言基础, 内容包括数据类型、表达式、控制流程、注解以及 Java 的开发环境等; 第 2 章主要介绍了 Java 语言的面向对象结构, 内容包括类与对象、类的继承、嵌套类、抽象类与接口、多态、泛型等; 第 3 章主要介绍了 lambda 表达式以及与其相关的方法引用等内容; 第 4 章主要介绍了数学类、正则表达式相关类、字符串类等一些常用实用类的使用; 第 5 章主要介绍了用于异常处理、线程以及反射的一些增强性能类; 第 6 章主要介绍了 Java 的输入输出流类及部分相关的应用; 第 7 章主要介绍了收集系列的接口与类以及它们在数据结构中的应用; 第 8 章主要介绍了 Java 的小程序及多媒体应用方面的内容; 第 9 章主要介绍了 Java 的图形用户界面以及桌面应用, 内容包括 AWT、Swing、JavaFX 的使用等; 第 10 章主要介绍了 JDBC 及在数据库中的应用, 内容包括数据库相关概念、JDBC 概述、JDBC 访问数据库、连接、事务、存储过程等; 第 11 章主要介绍了 Java 的网络应用, 内容包括 URL 应用、Socket 应用以及 Datagram 应用; 第 12 章主要介绍了 JavaBeans 的有关概念及其在 JSP 中的应用。

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 技术及应用/赵锐, 李卫华编著. —2 版. —北京: 清华大学出版社, 2017

(高等学校通识教育系列教材)

ISBN 978-7-302-47515-6

I. ①J… II. ①李… ②赵… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2017) 第 142121 号

责任编辑: 刘向威 李 晔

封面设计: 文 静

责任校对: 李建庄

责任印制: 刘海龙

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 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](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈: 010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 装 者: 北京泽宇印刷有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 20.75 字 数: 505 千字

版 次: 2009 年 6 月第 1 版 2017 年 8 月第 2 版 印 次: 2017 年 8 月第 1 次印刷

印 数: 1~2000

定 价: 49.00 元

产品编号: 069916-01

# 前 言

---

Java 是美国 Sun 公司于 1995 年 5 月推出的面向对象的通用编程语言，目前依然是世界上最优秀的计算机程序设计语言之一。本书主要讲授 Java 的技术基础及主要应用，作者根据多年的教学实践，采取深入浅出的方式描述 Java 的编程原理，并配上大量程序实例，给读者一定的引导，使其能尽快掌握 Java 的核心技术，并在自己感兴趣的领域继续深入学习。

本书是《Java 技术及应用》的第 2 版，对第 1 版内容做了如下改进。

(1) 为了更适合教学，调整了第 1 版中部分章节的顺序；

(2) 在第 1 版的原有章节中添加了一些内容作为补充，使得章节内容更为完整，便于读者学习理解；

(3) 增加了 lambda 表达式、接口方法的默认实现以及 JavaFX 等 Java 8 新引入内容的介绍；

(4) 根据 Java 8 技术的新特点对全书内容做了修正。

本书注重入门与提高，适合没有编程语言基础和面向对象编程基础的初学者。全书共分 12 章，分别介绍了 Java 的语言基础、面向对象结构、lambda 表达式及其应用、常用实用类、增强性能类、输入输出流、收集与数据结构应用、小程序及多媒体应用、图形用户界面及桌面应用、JDBC 与数据库应用、网络与 Web 服务应用、JavaBeans 及组件应用等方面的内容，一些过于深入、初学者难以理解的内容则没有引入或只是简单介绍。

本书各章后面均设计有理论问答题，让读者领会本章内容的要点；另外还附有编程实践题，供读者上机实验使用。

本书所有的实例和源程序均在 JDK 1.8 和 Eclipse 4.5 中运行通过。书中所有实例的源程序及其电子教案可以在清华大学出版社网站上免费下载使用。

由于编者水平有限，书中难免有许多缺点，敬请读者指正。

编 者

2017 年 3 月于广东工业大学

# 目 录

第 1 章 Java 基础	1
1.1 数据类型	1
1.1.1 Java 的标记集	1
1.1.2 基本数据类型的变量与声明	4
1.1.3 变量范围	5
1.1.4 数组类型	5
1.2 表达式	7
1.2.1 算术运算	7
1.2.2 关系运算	8
1.2.3 布尔逻辑运算	9
1.2.4 位运算	10
1.2.5 赋值运算	12
1.2.6 条件运算	13
1.2.7 类型转换运算	13
1.2.8 其他运算	14
1.3 控制流程	14
1.3.1 if-else 流程	14
1.3.2 switch 流程	16
1.3.3 for 流程	17
1.3.4 增强的 for 流程	18
1.3.5 while 流程	19
1.3.6 do-while 流程	20
1.3.7 break 语句	21
1.3.8 continue 语句	22
1.3.9 label 语句	23
1.3.10 return 语句	24
1.4 注解	25
1.5 编译工具	26
1.6 平台环境	27
1.6.1 PATH 和 CLASSPATH	27
1.6.2 编译与运行	27

1.6.3	命令行参数	27
1.6.4	集成开发环境	28
1.7	小结	29
	习题 1	29

## 第 2 章 面向对象结构 32

2.1	类与对象	32
2.1.1	类与对象的概念	32
2.1.2	类与对象的关系	33
2.2	类的定义	33
2.2.1	类声明部分	34
2.2.2	类体部分	35
2.2.3	成员变量	36
2.2.4	方法	38
2.2.5	构造方法	41
2.2.6	方法重载	42
2.3	对象	44
2.3.1	对象的创建	44
2.3.2	对象的使用	45
2.3.3	对象的清除	46
2.4	类的继承	47
2.4.1	合成与继承	47
2.4.2	方法重写	48
2.4.3	构造方法继承	49
2.4.4	类继承示例	49
2.5	嵌套类	51
2.5.1	静态嵌套类	52
2.5.2	内部类	52
2.5.3	局部内部类	53
2.5.4	匿名内部类	54
2.6	抽象类与接口	54
2.6.1	抽象类	54
2.6.2	接口	55
2.7	多态	60
2.7.1	抽象类与多态	61
2.7.2	接口与多态	62
2.8	泛型	63
2.8.1	定义泛型类型	64
2.8.2	限界类型参数	66

2.8.3	通配符	66
2.8.4	类型擦除	67
2.9	枚举	67
2.10	基本类型的类封装	68
2.11	包与版本识别	69
2.11.1	包	69
2.11.2	版本识别	71
2.12	小结	71
	习题 2	71
<b>第 3 章</b>	<b>lambda 表达式及其应用</b>	<b>73</b>
3.1	lambda 表达式简介	73
3.2	lambda 表达式应用	74
3.3	方法引用	77
3.4	小结	81
	习题 3	81
<b>第 4 章</b>	<b>常用实用类</b>	<b>82</b>
4.1	数学类	82
4.2	正则表达式支持类	83
4.2.1	正则表达式基础	84
4.2.2	正则表达式字符类	84
4.2.3	预定义字符集	85
4.2.4	量词	86
4.2.5	边界匹配符	86
4.2.6	Pattern 类	87
4.2.7	Matcher 类	88
4.2.8	PatternSyntaxException 类	89
4.3	字符串类	89
4.3.1	String 类	90
4.3.2	String 类和正则表达式	90
4.3.3	StringBuilder 类	91
4.4	日期时间类	93
4.4.1	Date	93
4.4.2	Calendar	94
4.4.3	GregorianCalendar	95
4.5	小结	95
	习题 4	95

<b>第5章 增强性能类</b> .....	97
5.1 异常处理.....	97
5.1.1 异常.....	97
5.1.2 捕获与声明的要求.....	99
5.1.3 处理异常.....	99
5.1.4 新形式的 try 块语句.....	104
5.1.5 抛出异常.....	105
5.1.6 创建自己的 Exception 类.....	106
5.2 并发.....	108
5.2.1 线程.....	109
5.2.2 同步与锁定.....	115
5.3 反射.....	121
5.3.1 Class 类.....	121
5.3.2 检查类信息.....	122
5.4 小结.....	123
习题 5.....	123
<b>第6章 输入输出流</b> .....	125
6.1 文件访问.....	125
6.1.1 File 类.....	125
6.1.2 RandomAccessFile 类.....	127
6.2 字节流.....	129
6.2.1 InputStream 及其子类.....	129
6.2.2 OutputStream 及其子类.....	132
6.2.3 文件字节流.....	134
6.2.4 管道流.....	135
6.2.5 数据流.....	136
6.2.6 字节缓冲流.....	139
6.2.7 字节打印流.....	140
6.2.8 字节数组流.....	142
6.2.9 对象流.....	143
6.3 字符流.....	145
6.3.1 Reader 类及其子类.....	145
6.3.2 Writer 类及其子类.....	146
6.3.3 字符缓冲流.....	146
6.3.4 转换流.....	147
6.3.5 字符打印流.....	149
6.4 新 I/O.....	149



6.4.1	Buffer 类	149
6.4.2	Channel 接口	150
6.5	扫描输入与格式化输出	150
6.5.1	Scanner 类	150
6.5.2	Formatter 类	151
6.6	小结	152
	习题 6	152
<b>第 7 章</b>	<b>收集与数据结构应用</b>	<b>154</b>
7.1	收集的概念	154
7.2	Collection 接口	155
7.3	Set	156
7.3.1	Set 的实现	156
7.3.2	Set 的数学应用	157
7.4	List	158
7.4.1	List 的实现	159
7.4.2	List 的数据结构应用	162
7.5	Queue	163
7.5.1	Queue 的实现	163
7.5.2	Queue 的数据结构应用	164
7.6	Map	165
7.6.1	Map 的实现	166
7.6.2	Map 的数学应用	168
7.7	SortedSet	169
7.8	SortedMap	171
7.9	Collections 类	173
7.9.1	静态方法	173
7.9.2	包装器	174
7.9.3	方便实现	175
7.9.4	Collections 类的数据结构应用	175
7.10	抽象实现	176
7.11	小结	177
	习题 7	177
<b>第 8 章</b>	<b>小程序及多媒体应用</b>	<b>178</b>
8.1	小应用程序	178
8.1.1	四个重要方法	178
8.1.2	绘制方法	181
8.1.3	事件处理方法	181

8.1.4	加入 java.awt 的方法	182
8.1.5	showStatus()方法	182
8.1.6	装入数据文件	182
8.1.7	使浏览器显示文档	183
8.1.8	查找同一页中运行的其他小程序	183
8.1.9	小应用程序的其他事项	183
8.2	2D 图形	184
8.2.1	Graphics 类	184
8.2.2	绘制基本图形	187
8.3	字体与颜色	191
8.3.1	字体	191
8.3.2	颜色	192
8.4	图像	194
8.4.1	装载图像	194
8.4.2	显示图像	195
8.4.3	复制图像	197
8.5	声音	197
8.6	动画	200
8.6.1	简单的多线程动画	201
8.6.2	改进动画效果的方法	202
8.6.3	增加控制组件	204
8.6.4	较完善的动画程序	204
8.7	小结	207
	习题 8	207
<b>第 9 章</b>	<b>图形用户界面及桌面应用</b>	<b>208</b>
9.1	AWT	208
9.1.1	GUI 组件类	208
9.1.2	布局管理器	218
9.1.3	事件处理	221
9.2	Swing	225
9.2.1	Swing 组件	226
9.2.2	Swing 并发性	235
9.2.3	事件监听	235
9.2.4	容器组件布局	241
9.2.5	修改视感	241
9.2.6	Swing 数据传送机制	242
9.2.7	拖和放	243
9.2.8	剪切、复制、粘贴	244

9.3	JavaFX .....	244
9.3.1	JavaFX 基础 .....	244
9.3.2	JavaFX 的控件 .....	246
9.3.3	JavaFX 的事件 .....	247
9.4	小结 .....	249
	习题 9 .....	249
<b>第 10 章</b>	<b>JDBC 与数据库应用 .....</b>	<b>251</b>
10.1	数据库的相关概念 .....	251
10.1.1	基本概念 .....	251
10.1.2	SQL .....	252
10.2	JDBC 概述 .....	254
10.2.1	JDBC 结构 .....	254
10.2.2	JDBC 的常用接口和类 .....	255
10.3	JDBC 访问数据库 .....	256
10.3.1	与数据库建立连接 .....	257
10.3.2	基本的数据访问 .....	258
10.3.3	元数据 .....	266
10.3.4	PreparedStatement .....	270
10.4	连接 .....	274
10.5	事务 .....	275
10.5.1	自动提交方式 .....	276
10.5.2	事务隔离级别 .....	276
10.5.3	保存点 .....	278
10.6	存储过程 .....	278
10.6.1	创建 CallableStatement 对象 .....	279
10.6.2	设置参数 .....	279
10.6.3	存储过程的访问 .....	280
10.7	JDBC 应用设计 .....	282
10.8	用 Applet 访问数据库 .....	282
10.9	小结 .....	285
	习题 10 .....	285
<b>第 11 章</b>	<b>网络与 Web 服务应用 .....</b>	<b>286</b>
11.1	Java 对网络通信的支持 .....	286
11.2	URL 应用 .....	287
11.2.1	URL 地址格式 .....	287
11.2.2	创建 URL 对象 .....	288
11.2.3	URL 类的方法 .....	289

11.2.4	读入 URL 资源	290
11.2.5	连接 URL	291
11.2.6	写入 URLConnection	291
11.3	Socket 应用	292
11.3.1	Socket 原理	292
11.3.2	读写 Socket	293
11.3.3	读写 ServerSocket	293
11.3.4	Socket 应用完整示例	295
11.4	Datagram 应用	297
11.4.1	Datagram 原理	297
11.4.2	编写 Datagram 服务器	298
11.4.3	编写 Datagram 客户端	299
11.4.4	Datagram 应用完整示例	300
11.5	小结	302
	习题 11	302
<b>第 12 章</b>	<b>JavaBeans 及组件应用</b>	<b>304</b>
12.1	JavaBeans 概念	304
12.2	设计简单的 bean	304
12.2.1	创建 bean	304
12.2.2	使用 bean	305
12.3	属性	306
12.3.1	简单属性	306
12.3.2	索引属性	307
12.3.3	关联属性	307
12.3.4	约束属性	307
12.4	事件	308
12.5	持续	308
12.6	自省	310
12.7	BeanContext API	311
12.8	在 JSP 中使用 JavaBeans	312
12.8.1	<jsp:useBean>	312
12.8.2	<jsp:setProperty>	313
12.8.3	<jsp:getProperty>	314
12.9	小结	314
	习题 12	315
	附录	316
	参考文献	318

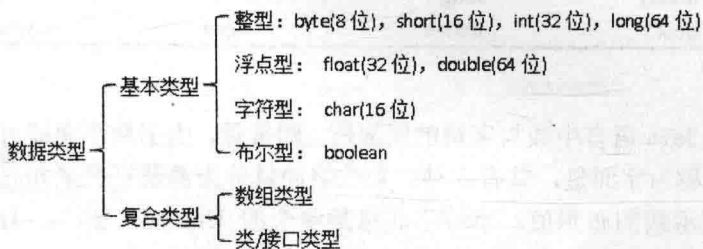
Java 语言的设计者采用 C 和 C++语言的语法格式来设计 Java 语言，目的是减轻大多数编程人员重新学习新语言的负担。但 Java 取消了不少 C 或 C++的特点，并加入了一些新的特性。例如，Java 取消了结构 (structure) 和联合 (union)、宏替换(#define)、指针、多重继承、单独的函数 (function)、goto 语句、操作符重载 (operator overloading)、自动强制转型 (automatic coercion) 等，而采用接口代替头文件 (header file)，用常量 (constants) 代替宏替换，用类取代结构和联合，函数封装到类中，用接口实现多重继承，必须用明显的语句说明类型转换等。由此看来，无论原来有没有 C 语言的基础，都要仔细了解 Java 语言的语法规范。

本章将对 Java 的数据类型、表达式、控制流程、字符串以及开发环境等逐一进行详细介绍。这些内容是各种编程语言的基础，必须首先了解清楚。

## 1.1 数据类型

数据类型是在高级语言中广泛使用的一个概念，表明这种语言中有什么数据结构，是构成编程语言极为重要的部分。

Java 语言的数据类型如下：



基本 (primitive) 类型是最简单的类型，不能由其他类型来构造。复合 (complex) 类型则是由基本类型构造而成的。本章介绍基本类型和数组类型，而把类与接口放到第 2 章。

在介绍基本类型之前先要谈谈 Java 的标记 (token) 集，也称为 Java 的词法，这是 Java 编译的基本单元。

注意：Java 语言的代码从 8 位 ASCII 码扩充到 16 位 Unicode，以支持非拉丁语言的字符。

### 1.1.1 Java 的标记集

Java 的标记共有 5 种：标识符 (identifier)、关键字 (keyword)、字面量 (literal)、运

算符 (operator) 和分隔符 (separator)。Java 程序就用这 5 类标记来编写, 再加上不属于标记集的注释 (comment)。

### 1. 标识符

标识符是赋给变量、类和方法的词, Java 的标识符必须是由字母、下画线 “\_” 及美元符号 “\$” 开头的字符数字串, 可以包括数字 (0~9)、字母 (大小写 A~Z) 以及编码小于十六进制数 00C0 的 Unicode 字符。

以下是合法的标识符:

x、y、HelloWorld、text\_1 等 (建议不用 “\_” 开头, 不使用 “\$” 符号, 不使用全部为大写字母的标识符)。

以下则是不合法的标识符:

5days、book-2、room# 等 (用关键字作标识符也不合法)。

### 2. 关键字

关键字是 Java 语言本身使用的特殊词, 不能作为标识符赋给变量、类和方法。表 1-1 列出了 Java 使用的关键字, 其中, 带\*号者保留未用, 带\*\*者是 1.2 版加入的, 带\*\*\*者是 1.4 版加入的, 带\*\*\*\*者是 5.0 版加入的。

表 1-1 Java 关键字

abstract	continue	for	new	switch
assert***	default	goto*	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum****	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp**	volatile
const*	float	native	super	while

### 3. 字面量

字面量表示 Java 语言中数与字符的明显值, 即常量。由于从字面即可判定它们是哪一类的常量, 所以称为字面量, 共有 5 种: 数的字面量分为整数字面量和浮点数字面量; 布尔字面量用于表示数的布尔值; 字符字面量是单个的 Unicode 字符, 串字面量则有多个字符。

(1) 整数字面量通常有十进制、十六进制和八进制格式。十进制就是常见的数字, 如 63、100、82986 等; 十六进制以 0x (或 0X) 开头, 如 0x10、0x5AF7 等; 八进制以 0 开头, 如 02、077 等。整数字面量通常表示 32 位二进制数范围内的数, 若上述字面量后加字母 l 或 L, 则表示 64 位。

(2) 浮点数字面量用于表示带小数部分的十进制数, 如 4.、.3、70.16、5.29E8、1.7e-19 (后两个是科学记数法表示)。浮点数的单精度 (single precision) 占 32 位存储空间, 字母 f 或 F 跟在数后 (不加字母会默认为是双精度的); 双精度 (double precision) 占 64 位存储空间, d 或 D 跟在数后面, 如 5.29E8F、1.7e-19D。

(3) 布尔字面量有 true 和 false 两种, 代表逻辑真和假, 不能用数字 1 和 0 来代替, 这是 Java 与 C 的又一区别。

(4) 字符字面量用单引号 ' ' 对包含, 可以取 Unicode 的任何单个字符, 如 'a'、'B' 等; 再用反斜线序列支持转义字符, 如 '\n' 等。常用的反斜线序列如表 1-2 所示。

表 1-2 常用的反斜线序列

序列	描述	序列	描述
\	继续下行	\\	反斜线
\n	换行	\'	单引号
\t	横向跳格	\"	双引号
\b	退格	\ddd	3 位八进制数
\r	回车	\xdd	2 位十六进制数
\f	走纸	\udddd	Unicode 字符

(5) 串字面量是用双引号 " " 对包括起来的任意数量的字符。在 Java 中, 串是用 String 类实现的, 每个串字面量都是 String 类的一个新实例。以下是一些串字面量: "" (空串)、"A"、"\"、"This is a class."、"One line\nTwo line" 等。

#### 4. 运算符

Java 语言用一些符号表示一些基本运算形式, 这些符号叫运算符或操作符。表 1-3 按运算优先级从高到低列出了所有运算符。

表 1-3 运算符

运算类型	运算符号
后缀(postfix)运算	. [] () expr++ expr--
一元(unary)运算	++expr --expr +expr -expr ! ~
建立(creation)或转型(cast)	new (type)expr
乘(multiplicative)	* / %
加(additive)	+ -
移位(shift)	<< >> >>>
关系(relational)	< > <= >= instanceof
相等(equality)	== !=
按位与(bitwise AND)	&
按位异或(bitwise exclusive OR)	^
按位或(bitwise inclusive OR)	
逻辑与(logical AND)	&&
逻辑或(logical OR)	
条件(conditional)	?:
赋值(assignment)	= op=

#### 5. 分隔符

分隔符用于告诉编译器代码从哪里分开和组合, 常用的有 ()、{}、;、, 等。

#### 6. 注释

Java 有三种注释格式。

(1) /\* ... \*/ 用于多行注释;

(2) // ...用于单行注释, 到行尾自动终止;

(3) /\*\* ... \*/用于自动文档产生器 javadoc 产生注释文档, 它必须位于声明 (declaration) 部分之前。

### 1.1.2 基本数据类型的变量与声明

变量用于表示 Java 在内存中存储的一个数据, 它用带有类型的标识符表示, 需要用声明语句来建立:

```
type identifier [,identifier];
```

该语句告诉编译器建立名为 identifier、类型为 type 的一个变量。若声明多个同类的变量, 则用逗号隔开, 最后用分号结束。声明了变量之后就可以赋值给它, 并可以操作, Java 允许在声明变量时对变量赋初值:

```
type identifier=value;
```

下面分别介绍 4 种基本类型变量的声明。

#### 1. 整型变量

整型变量有 4 种, 分别是 8 位的 byte 类、16 位的 short 类、32 位的 int 类和 64 位的 long 类。全部是有符号数 (其中 byte 的范围是 -256~255, 即最高位也可以用于表示正数)。其声明为

```
byte op1;  
short op2=2;  
int op3;  
long op4=4;
```

#### 2. 浮点型变量

浮点型变量有 float 型和 double 型两种, 前者是 32 位单精度浮点数, 后者是 64 位双精度浮点数。其声明为

```
float op1;  
double op2=0.5;
```

如果对 float 型数据赋初值, 必须在初值后加 f (或 F), 如:

```
float op1=30.0f;
```

而在 double 型数据后可不加 d 或 D, 因为 Java 默认把浮点数定义为 double 型。

#### 3. 字符型变量

Java 的字符类型变量 char 是 16 位无符号整数, 用于表示 16 位的 Unicode。字符变量只存储单个字符, 其声明为

```
char op1;  
char op2='A', op3='\n';
```



## 4. 布尔型变量

布尔型变量 `boolean` 可取逻辑真和假两种值，但不代表数字 1 或 0，其声明为：

```
boolean op1;  
boolean op2=false,op3=true;
```

### 1.1.3 变量范围

变量声明之后，它的作用范围（`scope`）就确定了：从变量声明的位置开始到它所在的代码块（`block`）结束之处。所谓块，就是由花括号对 `{}` 包含的一段代码。举例如下：

```
class myApp {  
    public static void main(String args[]) {  
        int x;  
        ...  
    }  
    public void mymethod() {  
        char y;  
        ...  
    }  
}
```

整数变量 `x` 的作用范围在 `main()` 方法中，不能用在 `mymethod()` 方法中；而字符变量 `y` 只能用于 `mymethod()` 中，不能用于 `main()` 中。

如果在一个大范围内嵌套了一个小范围，而两个范围的某个变量同名，当程序进入小范围时，大范围的同名变量会被隐藏不用，直到退出小范围才恢复使用。因此，在定义变量名时要注意。

### 1.1.4 数组类型

数组是 Java 的一种复合类型，是由同类型的对象组成的，这些对象可由索引（`indexing`）来引用。数组中的对象也可以是数组，即数组的嵌套，但并不像 C++ 那样称为多维数组。声明数组用方括号对 `[]` 加在标识符后（或类型后），如：

```
int A[]; 或 int[] A;  
char B[][];  
float C[][][];
```

可以用 `{}` 赋初值，如：

```
int op1[]={1,2,3};  
char[] op2={'a', 'b', 'c', 'd'};
```

但这种赋值不能用于非声明的场合。

**注意：**声明数组时不能像 C++ 那样在方括号内加数字表示数组长度，而必须用 `new` 运算来分配。如：