

参考大全

涵盖 JDK
1.3 最新
版本

JavaTM 2

(第四版)

内容涵盖最新
Java 语言标准和
类库

详细讲解 AWT
Swing, JavaBeans
和多线程编程

深入剖析类集
框架和最新的
Timer 类

[美] Herbert Schildt 著 (逾 3 千万册编程类畅销图书作者)

张玉清 吴溥峰 尚勇 等译



清华大学出版社

<http://www.tup.tsinghua.edu.cn>

McGraw-Hill

北京科海培训中心

Java 2 参考大全

[美] Herbert Schildt 著

张玉清 吴溥峰 尚勇 等译

清华大学出版社

(京)新登字 158 号

著作权合同登记号:01-2001-3287

内 容 提 要

本书是世界一流的程序设计大师 Herbert Schildt 的力作。

全书从介绍 Java 语言的基础知识入手,通过丰富的示例系统地介绍了 Java 语言基础知识、Java 标准库及 Java 开发环境,让读者通过由简单到复杂的示例程序来逐步理解 Java 语言的精髓。Java 的强大功能体现在它的标准库中,书中对 Java 标准库及其应用进行了完整深入的介绍,方便读者查阅,这部分内容是本书的重点。最后介绍了 Java 开发环境及几个综合应用示例,让读者达到学以致用目的。

本书涵盖了 Java 应用程序和小应用程序的开发、编译、调试和运行所需要的所有内容,并包括了 Java 开发工具包 JDK 1.3 的新增特性,是每一个 Java 程序员理想的参考大全。

Java™ 2: The Complete Reference, Fourth Edition

Copyright ©2001 by The McGraw-Hill Companies.

本书中文简体字版由美国麦格劳-希尔教育(亚洲)出版公司与清华大学出版社和北京科海培训中心合作出版。未经出版者书面允许不得以任何方式复制或抄袭本书内容。

版权所有,盗版必究。

本书封面贴有 McGraw-Hill 激光防伪标签,无标签者不得进入各书店。

MSL 108

书 名: Java 2 参考大全

作 者: Herbert Schildt

译 者: 张玉清 吴溥峰 尚 勇

出版者: 清华大学出版社(北京清华大学校内,邮编 100084)

印刷者: 北京门头沟胶印厂

发行者: 新华书店总店北京科技发行所

开 本: 787×1092 1/16 印张: 48.125 字数: 1170 千字

版 次: 2002 年 1 月第 1 版 2002 年 1 月第 1 次印刷

印 数: 0001~5000

书 号: ISBN 7-302-05016-3/TP·2931

定 价: 79.00 元

关于作者

Herbert Schildt是世界一流的程序设计语言的作者。他是C, C++和Java语言的权威，同时也精通Windows编程。他编写的有关程序设计语言的书籍在世界上已销售了250多万册，并翻译成多种文字发行。他编著过许多畅销书，其中包括：《Java 2: The Complete Reference》，《Java 2 Programmers Reference》，《C++: The Complete Reference》，《Windows 2000 Programming from the Ground Up》和《C: The Complete Reference》。Schildt获得伊利诺斯州州立大学计算机科学的硕士学位，可以通过他的咨询办公室的电话（217）586-4683与他联系。

序

过去几年证明了这样的事实：**Web**已无可阻挡地重铸了计算的面目，不愿意掌握它的程序员们将会落伍。

这种说法尽管沉重，不过却是事实。各式各样的计算机应用将越来越密切地与**Web**交接。应用具体干什么不再显得那么重要，差不多无所不在的**Web**访问正在不断地拖拉、推动并引诱程序员们给这个在线的世界编写程序——**Java**就是许多程序员们用来这么做的计算机语言。坦率地说，流利的**Java**编程对专业程序员来说已不再可有可无，而是必备的能力。本书将帮助你学到这种能力。

Java不仅仅是因特网上的卓越语言，它的重要性还体现为改变了计算机语言的发展进程。由**Java**首先吸收的许多特性目前正在流入其他语言。**Java**知识在编程领域敞开了最近若干革新的大门。一句话，**Java**是最为重要的计算机语言之一。

一本面向所有程序员的书

使用本书不需要任何编程经验。然而，如果你学过**C/C++**，将有助于你更快地掌握**Java**语言。我们都知道，**Java**与**C/C++**非常相似，所以具备**C/C++**语言知识对学习**Java**语言无疑是很帮助的，但并不是必须具备的。即使你以前从未编写过程序，你也能通过本书来学习**Java**程序设计。

本书的内容

这本书包括了**Java**程序设计语言的方方面面。第1部分深入浅出地介绍了**Java**语言，这部分内容既讲述了数据类型、控制语句和类等基本知识，也讨论了**Java**语言的异常处理机制、多线程子系统、包和接口。

第2部分详尽介绍了**Java**语言的标准库，正如你将要学到的，**Java**的强大功能体现在它的库中。本部分主要包括如下主题：字符串、I/O、网络连接、标准实用工具、类集框架、小应用程序、GUI控件和图像。

第3部分介绍了与**Java**语言开发环境有关的一些问题，包括**Java Beans**组件和**Swing**组件综述。

第4部分介绍了几个功能强大的**Java**小应用程序，通过示例具体说明如何扩展**Java**的应用能力。其中最后一个小应用程序**Scrabblet**，是个完整的多用户网络游戏，它显示了如何处理基于**Web**编程中的一些最棘手的问题。

本书所有示例源代码都可从www.osborne.com网址免费得到。

版本4的新内容

本书为《Complete reference Java 2》的第4版。本书与其早期版本的差异在于，它新增了Java 2 JDK版本1.3的内容。JDK 1.3并没有对Java语言和库进行重要修改，它只添加了一些内容，例如，给java.util增加了两个类——Timer和TimerTask，这允许你安排一个线程于将来的某个时间执行。JDK 1.3也对许多现有的类添加了一些新方法。本书将充分反映出这些新增内容。

因为Sun公司努力使Java 2版本1.2和Java 2版本1.3之间保持向上兼容，本书中的绝大多数程序和小应用程序都能在Java 2的这两个版本上运行。然而，那些隶属于JDK1.3特性编写的程序将不能在其早期版本上运行。因为这个原因，JDK 1.3中增加的新特性将在文字中特别说明。

特 别 致 谢

特别感谢Java语言的创始者之一，Patrick Naughton。他帮助编写了本书第1版，其中第17，18，23，28，29，30和31章的大多数材料就是由Patrick提供的，他超人的洞察力、丰富的专家经验及敏锐的思维能力对这本书的成功做出了巨大贡献。

同时也要感谢Joe O'Neil为第24，25和26章提供的最初草稿，他帮助我编写过好几本书，我也一直欣赏和感谢他的努力和帮助。

HERBERT SCHILDT

2000, 11, 8

Mahomet, Illinois

目 录

第1部分 Java语言

第1章 Java的起源	1
1.1 Java的由来	1
1.1.1 现代的编程语言的诞生：C语言	1
1.1.2 对C++的需要	3
1.1.3 Java出现的时机已经到来	3
1.2 Java的产生	4
1.3 Java对Internet为什么重要	5
1.3.1 Java小应用程序和应用程序	5
1.3.2 安全性	6
1.3.3 可移植性	6
1.4 Java的魔力：字节码	6
1.5 Java常用语	7
1.5.1 简单	7
1.5.2 面向对象	8
1.5.3 健壮	8
1.5.4 多线程	8
1.5.5 结构中立	9
1.5.6 解释性和高性能	9
1.5.7 分布式	9
1.5.8 动态	9
1.6 继续革命	9
1.7 Java不是增强的HTML	10
第2章 Java语言概述	11
2.1 面向对象编程	11
2.1.1 两种范型	11
2.1.2 抽象	11
2.1.3 面向对象编程的3个原则	12
2.2 第1个简单程序	16
2.2.1 键入程序	16
2.2.2 编译程序	17
2.2.3 详细讨论第1个示例程序	17

2.3	第2个示例程序	19
2.4	两个控制语句	21
2.4.1	if控制语句	21
2.4.2	for循环	22
2.5	使用程序块	23
2.6	基本词汇	25
2.6.1	空白分隔符 (whitespace)	25
2.6.2	标识符 (identifiers)	25
2.6.3	常量 (literal)	25
2.6.4	注释 (comments)	25
2.6.5	分隔符 (separators)	26
2.6.6	Java关键字	26
2.6.7	Java类库	26
第3章	数据类型、变量、数组	28
3.1	Java语言是强类型语言	28
3.2	简单数据类型	28
3.3	整数类型	29
3.3.1	字节型 (byte)	29
3.3.2	短整型 (short)	30
3.3.3	整型 (int)	30
3.3.4	长整型 (long)	30
3.4	浮点型 (Floating-Point Types)	31
3.4.1	单精度浮点型 (float)	31
3.4.2	双精度型 (double) 浮点型	32
3.5	字符	32
3.6	布尔型	33
3.7	进一步研究字面量	34
3.7.1	整数字面量	34
3.7.2	浮点字面量	35
3.7.3	布尔型字面量	35
3.7.4	字符字面量	35
3.7.5	字符串字面量	36
3.8	变量	36
3.8.1	声明一个变量	36
3.8.2	动态初始化	37
3.8.3	变量的作用域和生存期	37
3.9	类型转换与强制类型转换	39
3.9.1	Java的自动转换	40

3.9.2 不兼容类型的强制转换	40
3.10 表达式中类型的自动提升	41
3.10.1 类型提升的约定	42
3.11 数组	43
3.11.1 一维数组	43
3.11.2 多维数组	45
3.11.3 另一种数组声明语法	49
3.12 字符串的简单介绍	49
3.13 C/C++程序员请注意指针的用法	49
第4章 运算符	51
4.1 算术运算符	51
4.1.1 基本算术运算符	51
4.1.2 模运算符	52
4.1.3 算术赋值运算符	53
4.1.4 递增和递减运算	54
4.2 位运算符	55
4.2.1 位逻辑运算符	56
4.2.2 左移运算符	58
4.2.3 右移运算符	60
4.2.4 无符号右移	61
4.2.5 位运算符赋值	62
4.3 关系运算符	63
4.4 布尔逻辑运算符	64
4.4.1 短路 (short-circuit) 逻辑运算符	65
4.5 赋值运算符	66
4.6 ? 运算符	66
4.7 运算符优先级	67
4.8 使用圆括号	68
第5章 程序控制语句	69
5.1 Java的选择语句	69
5.1.1 if语句	69
5.1.2 switch语句	72
5.2 循环语句	75
5.2.1 while语句	76
5.2.2 do-while循环	77
5.2.3 for循环	79
5.2.4 for循环的一些变化	82
5.2.5 循环嵌套	83

5.3 跳转语句	83
5.3.1 使用break语句	84
5.3.2 使用continue语句	87
5.3.3 使用return语句	88
第6章 介绍类	90
6.1 类基础	90
6.1.1 类的通用格式	90
6.1.2 一个简单的类	91
6.2 声明对象	93
6.2.1 深入研究new运算符	94
6.3 给对象引用变量赋值	95
6.4 方法	96
6.4.1 为Box类添加一个方法	96
6.4.2 返回值	98
6.4.3 加入带自变量的方法	99
6.5 构造函数	101
6.5.1 带自变量的构造函数	103
6.6 this关键字	104
6.6.1 隐藏的实例变量	104
6.7 垃圾回收	105
6.8 finalize () 方法	105
6.9 一个堆栈类	106
第7章 进一步研究方法和类	109
7.1 方法重载	109
7.1.1 构造函数重载	111
7.2 把对象作为参数	113
7.3 参数是如何传递的	116
7.4 返回对象	117
7.5 递归	118
7.6 介绍访问控制	120
7.7 理解static	123
7.8 介绍final	125
7.9 重新温习数组	126
7.10 介绍嵌套类和内部类	127
7.11 探索String类	130
7.12 使用命令行参数	132

第8章 继承	134
8.1 继承的基础	134
8.1.1 成员的访问和继承	135
8.1.2 更实际的例子	136
8.1.3 超类变量可以引用子类对象	138
8.2 使用super	139
8.2.1 使用super调用超类构造函数	139
8.2.2 Super的第2种用法	142
8.3 创建多级类层次	143
8.4 何时调用构造函数	146
8.5 方法重载	147
8.6 动态方法调度	149
8.6.1 为什么要重载方法	151
8.6.2 应用方法重载	151
8.7 使用抽象类	153
8.8 继承中使用final	155
8.8.1 使用final阻止重载	155
8.8.2 使用final阻止继承	156
8.9 Object类	156
第9章 包和接口	158
9.1 包	158
9.1.1 定义包	158
9.1.2 理解类路径 (CLASSPATH)	159
9.1.3 一个简短的包的例子	160
9.2 访问保护	161
9.2.1 一个访问的例子	162
9.3 引入包	164
9.4 接口 (interface)	166
9.4.1 接口定义	167
9.4.2 实现接口	167
9.4.3 应用接口	170
9.4.4 接口中的变量	173
9.4.5 接口可以扩展	174
第10章 异常处理	176
10.1 异常处理基础	176
10.2 异常类型	177
10.3 未被捕获的异常	177
10.4 使用try和catch	178

10.4.1 显示一个异常的描述	179
10.5 使用多重catch 语句	180
10.6 嵌套try语句	181
10.7 引发 (throw)	183
10.8 throws	184
10.9 finally	186
10.10 Java的内置异常	187
10.11 创建自己的异常子类	188
10.12 使用异常	190
第11章 多线程编程	191
11.1 Java线程模型	191
11.1.1 线程优先级	192
11.1.2 同步性	192
11.1.3 消息传递	193
11.1.4 Thread 类和Runnable 接口	193
11.2 主线程	193
11.3 创建线程	195
11.3.1 实现Runnable接口	195
11.3.2 扩展Thread	197
11.3.3 选择合适方法	198
11.4 创建多线程	198
11.5 使用isAlive () 和join ()	200
11.6 线程优先级	202
11.7 线程同步	204
11.7.1 使用同步方法	205
11.7.2 同步语句	207
11.8 线程间通信	208
11.8.1 死锁	212
11.9 挂起、恢复和终止线程	214
11.9.1 Java 1.1或更早版本的线程的挂起、恢复和终止	214
11.9.2 Java 2中挂起、恢复和终止线程	216
11.10 使用多线程机制	218
第12章 输入/输出、小应用程序和其他主题	219
12.1 输入/输出基础	219
12.1.1 流的概念	219
12.1.2 字节流和字符流	219
12.1.3 预定义流	221
12.2 读取控制台输入	222

12.2.1 读取字符	222
12.2.2 读取字符串	223
12.3 向控制台写输出	225
12.4 PrintWriter类	225
12.5 文件的读写	226
12.6 小应用程序基础	229
12.7 Transient和volatile修饰符	232
12.8 使用instanceof	232
12.9 strictfp	234
12.10 本机方法	235
12.11 使用本机方法的问题	238

第2部分 Java库

第13章 字符串处理	239
13.1 String构造函数	239
13.2 字符串长度	241
13.3 特殊的字符串操作	241
13.3.1 字符串文字	241
13.3.2 字符串连接	242
13.3.3 字符串与其他类型数据的连接	242
13.3.4 字符串转换和toString()	243
13.4 字符截取	244
13.4.1 charAt()	244
13.4.2 getChars()	245
13.4.3 getBytes()	245
13.4.4 toCharArray()	245
13.5 字符串比较	246
13.5.1 equals()和 equalsIgnoreCase()	246
13.5.2 regionMatches()	247
13.5.3 startsWith()和endsWith()	247
13.5.4 equals()与==的比较	247
13.5.5 compareTo()	248
13.6 搜索字符串	249
13.7 修改字符串	251
13.7.1 substring()	251
13.7.2 concat()	252
13.7.3 replace()	252

13.7.4 trim()	252
13.8 利用valueOf()方法实现数据转换	253
13.9 改变字符串内字符的大小写	254
13.10 StringBuffer	255
13.10.1 StringBuffer构造函数	255
13.10.2 length()和capacity()	255
13.10.3 ensureCapacity()	256
13.10.4 setLength()	256
13.10.5 charAt()和setCharAt()	256
13.10.6 getChars()	257
13.10.7 append()	257
13.10.8 insert()	258
13.10.9 reverse()	258
13.10.10 delete()和deleteCharAt()	259
13.10.11 replace()	259
13.10.12 substring()	260
第14章 java.lang研究	261
14.1 简单类型包装器	261
14.1.1 Number	262
14.1.2 Double和Float	262
14.1.3 Byte, Short, Integer 和 Long	265
14.1.4 Character	271
14.1.5 Boolean	274
14.2 Void	274
14.3 Process	274
14.4 Runtime	275
14.4.1 内存管理	276
14.4.2 执行其他的程序	277
14.5 System	278
14.5.1 使用currentTimeMillis()记录程序执行的时间	280
14.5.2 使用arraycopy()	280
14.5.3 环境属性	281
14.6 Object	281
14.7 使用clone()和Cloneable接口	282
14.8 Class	284
14.9 ClassLoader	286
14.10 Math	287
14.10.1 超越函数	287

14.10.2	指数函数	287
14.10.3	舍入函数	287
14.10.4	其他的数学方法	288
14.11	StrictMath	289
14.12	Compiler	289
14.13	Thread, ThreadGroup和Runnable	289
14.13.1	Runnable接口	289
14.13.2	Thread	289
14.13.3	ThreadGroup	291
14.14	ThreadLocal和InheritableThreadLocal	295
14.15	Package	295
14.16	RuntimePermission	296
14.17	Throwable	296
14.18	SecurityManager	297
14.19	Comparable接口	297
14.20	java.lang.ref和java.lang.reflect包	297
14.20.1	java.lang.ref	297
14.20.2	java.lang.reflect	298
第15章	java.util第1部分: 类集框架	299
15.1	类集概述	300
15.2	类集接口	301
15.2.1	类集接口	301
15.2.2	List接口	303
15.2.3	集合接口	304
15.2.4	SortedSet接口	304
15.3	Collection类	305
15.3.1	ArrayList类	305
15.3.2	LinkedList类	308
15.3.3	HashSet类	310
15.3.4	TreeSet类	311
15.4	通过迭代函数访问类集	312
15.4.1	使用迭代函数	312
15.5	将用户定义类存储于Collection中	314
15.6	处理映射	315
15.6.1	映射接口	316
15.6.2	映射类	318
15.7	比较函数	321
15.7.1	使用比较函数	322

15.8 类集算法	324
15.9 Arrays (数组)	327
15.10 从以前版本遗留下来的类和接口	330
15.10.1 Enumeration接口	331
15.10.2 Vector	331
15.10.3 Stack	334
15.10.4 Dictionary	336
15.10.5 Hashtable	337
15.10.6 Properties	340
15.10.7 使用store()和load()	343
15.11 类集总结	344
第16章 java.util第2部分: 更多的实用工具类	345
16.1 StringTokenizer (字符串标记)	345
16.2 BitSet (置位)	346
16.3 Date (日期)	348
16.3.1 比较日期	350
16.4 Calendar (日历)	350
16.5 GregorianCalendar (标准阳历)	353
16.6 TimeZone (时区)	354
16.7 SimpleTimeZone	355
16.8 Locale (地区)	356
16.9 Random	357
16.10 Observable (观测)	359
16.10.1 观测接口	359
16.10.2 观测程序举例	360
16.11 Timer和TimerTask	362
16.12 java.util.zip包	364
16.13 java.util.jar包	364
第17章 输入/输出: 探究java.io	365
17.1 Java输入/输出类和接口	365
17.2 File (文件类)	366
17.2.1 目录	368
17.2.2 使用FilenameFilter	369
17.2.3 listFiles() 方法	370
17.2.4 创建目录	370
17.3 流类	370
17.4 字节流	371
17.4.1 InputStream (输入流)	371

17.4.2	OutputStream (输出流)	371
17.4.3	FileInputStream (文件输入流)	372
17.4.4	FileOutputStream (文件输出流)	374
17.4.5	ByteArrayInputStream (字节数组输入流)	375
17.4.6	ByteArrayOutputStream (字节数组输出流)	376
17.4.7	过滤字节流	377
17.4.8	缓冲字节流	377
17.4.9	SequenceInputStream (顺序输入流)	380
17.4.10	PrintStream (打印流)	382
17.4.11	RandomAccessFile (随机访问文件类)	382
17.5	字符流	382
17.5.1	Reader	383
17.5.2	Writer	383
17.5.3	FileReader	384
17.5.4	FileWriter	384
17.5.5	CharArrayReader	385
17.5.6	CharArrayWriter	386
17.5.7	BufferedReader	387
17.5.8	BufferedWriter	388
17.5.9	PushbackReader	388
17.5.10	PrintWriter	389
17.6	使用流式输入/输出	390
17.6.1	用StreamTokenizer (流标记) 来改善wc()	391
17.7	序列化	393
17.7.1	Serializable接口	394
17.7.2	Externalizable接口	394
17.7.3	ObjectOutput接口	394
17.7.4	ObjectOutputStream类	394
17.7.5	ObjectInput	395
17.7.6	ObjectInputStream	396
17.7.7	序列化示例	397
17.8	流的益处	398
第18章	网络	399
18.1	网络基础	399
18.1.1	套接字概述	399
18.1.2	客户/服务器模式	400
18.1.3	保留套接字	400
18.1.4	代理服务器	401