

疯狂源自梦想，
技术成就辉煌

疯狂 Java

突破程序员
基本功的10课

(修订版)

■ 李刚 著

人民邮电出版社
POSTS & TELECOM PRESS

疯狂 Java

突破程序员
基本菌的16课
(修订版)

■ 李刚 著

人民邮电出版社

北京

图书在版编目 (C I P) 数据

疯狂Java: 突破程序员基本功的16课 / 李刚著. —
修订本. — 北京: 人民邮电出版社, 2012. 1
ISBN 978-7-115-26304-9

I. ①疯… II. ①李… III. ①JAVA语言—程序设计
IV. ①TP312

中国版本图书馆CIP数据核字(2011)第205911号

内 容 提 要

本书是 Java 领域著名研究专家、Java 语言培训导师、“疯狂 Java”创始人李刚老师的又一倾心力作。

在本书中,李刚老师专门针对 Java 初学者可能会遇到的各种学习问题,由点及面,详细讨论了 Java 内存管理、Java 编程过程中常遇陷阱、常用数据结构的 Java 实现和 Java 程序开发的方法与经验等内容。

这些问题,看似“司空见惯”,实际上却是很多 Java 初学者都会遇到的问题和疑难。李刚老师在本书中,试图为读者们展现出这些疑点、难点的实质,让读者能在瞬息之间,彻底掌握住这门语言的“内功心法”。

这不是一本包含了所有技术细节的手册,而是一本 Java 前辈对于晚辈们的提点和教导。书中很多内容,是李刚老师和他的众多学子曾亲身体验过的困难和应对之策,非常具有参考意义。本书承载了无数前辈的谆谆教导之言,向你展示着一个痛并快乐着的 Java 编程世界。

疯狂 Java: 突破程序员基本功的 16 课 (修订版)

- ◆ 著 李 刚
责任编辑 蒋 佳
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京鑫正大印刷有限公司印刷
- ◆ 开本: 800×1000 1/16
印张: 35
字数: 744 千字 2012 年 1 月第 1 版
印数: 1-3 500 册 2012 年 1 月北京第 1 次印刷

ISBN 978-7-115-26304-9

定价: 59.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223
反盗版热线: (010)67171154



前言

用 Java 语言进行开发的人越来越多了,大量程序员已经进入或正打算进入 Java 编程领域。这当然和 Java 语言本身的优秀不无关系,却也和 Java 语言入门简单有关。一个毫无编程基础的初学者,只要有点数据库和 SQL 的基础,大概花不到一个月时间就可以学会编写 JSP 页面,说不定这样就可以找到一份 Java 编程的工作了。如果他肯再多下点工夫,学习一下编写 Struts Action 类、配置 Action, 编写 Spring Bean 类和配置 Bean, 他甚至都自我感觉很不错了。

问题是:这种“快餐式”、“突击式”的程序员真的能满足企业要求吗?如果仅仅满足于这些简单的、重复式的开发,他们也许没有太多的问题,但他们可能很少有突破职业瓶颈的机会。究其根本原因,很大程度上是因为他们的基本功不够扎实。对他们而言,与其说 Java 是一种面向对象的语言,不如说更像一种脚本;他们从源代码层次来看程序运行(甚至只会从 Eclipse 等集成开发环境中看程序运行),完全无法从底层内存分配的角度来看程序运行;他们天天在用 Java 类库、用 Struts、用 Spring,但对这些东西的实现却知之甚少——这又如何突破自己、获得更好的提高呢?

鉴于此种现状,笔者在“疯狂 Java 实训营”的训练中,除了采用大量实际项目来驱动教学之外,往往会花很多时间、精力来培养学员的基本功。比如讲授 Spring 框架时,如果只关注如何编写 Bean 类、配置 Bean,那么一天的时间就足够了。而笔者往往会深入 Spring 框架的底层实现,带领学生从工厂模式、策略模式、门面模式、代理模式、命令模式的角度来深度分析 Spring 框架实现,然后进行对比,向学生提出问题,启发他们思考,例如,总结 Spring 框架的优势到底在哪里?不使用 Spring 框架是否有替换解决方案?从而让学生感受设计模式对实际开发的帮助。

上面这些内容,看似“高深”,但其实质依然离不开 Java 编程的基本功。也可以这样说:一旦读者真正把基本功打扎实了,将看得更高、更透彻。

在这样的背景下，偶然之间我获得了这样一个想法：把这些容易被 Java 程序员所忽视的内容整理成一本书，也许可以帮助这些需要突破基本功的准程序员、初级程序员。本书的写法对笔者来说，是一种新的尝试：笔者以前所写之书，通常都会围绕某方面知识“画一个圆”——把这个方面的相关知识，全面、详细地向读者介绍出来。本书并没有采用这种写法，本书的每课都围绕一个相对独立的专题进行讲解，形式上有点类似于“针对性地补课”。当读者感觉自己在某些方面的基本功不够扎实时，可以直接选择相应的内容进行学习，完全无须从头到尾地按顺序阅读。

本书内容

《疯狂 Java：突破程序员基本功的 16 课（修订版）》的创作过程非常困难，当初刚刚获得这个想法之时，我也曾感到有些力不从心，程序员的基本功看似非常简单，但覆盖范围却非常广泛，因此似乎很难形成一个完整、系统的知识体系。

但随着写作过程的不断推进，笔者心中才渐渐明朗起来：该书是一本解惑之书，它如能将 Java 学习、实际开发中最容易被大家忽视又非常重要的内容收集起来，便足够了。

期间，ACS（天津）公司邀请笔者为其公司员工进行了一次技术内训，内容主要围绕 Java 程序的内存管理问题。内训结束时该公司员工反应很好，这给笔者带来了很大的触动。Java 程序的内存管理既是 Java 程序员容易忽视的地方，但又是 Java 编程的重点。后来，笔者即将此次内训的知识进行整理，并最终形成了本书第 1 部分（1~4 课）的内容。

本书第 3 部分（9~12 课）所介绍的常见数据结构、排序算法的 Java 实现，则是笔者一直想介绍的内容。在笔者的“疯狂 Java 实训营”内，常见数据结构、排序算法是必须掌握的——也许你初涉编程时感受不到这些经典算法的用途，因为你可以直接利用别人的实现，但如果你希望突破自己，上升到另外一个高度时，你就不可避免地需要自己开发类库，而不是总使用别人的类库，那这些经典算法的作用就显现出来了。

本书第 2 部分（5~8 课）和第 4 部分（13~16 课）则主要来自于参加“疯狂 Java 实训营”的学生，正如每个动手编程的初学者，他们都曾经遭遇着各种各样的陷阱，因此笔者总是提醒他们应该将这些陷阱收集起来，以免再次陷进去。本书第 2 部分收集了 Java 编程中各种常见的陷阱，第 4 部分的内容则解决了他们进入实际开发之前的诸多困扰，包括程序开发的基本方法，有效进行程序调试的方法，如何看待、使用 IDE 工具，软件测试等相关内容。

修订说明

本书的第 1 版《疯狂 Java: 突破程序员基本功的 16 课》自从 2010 年 5 月出版以来, 受到了广大读者的一致好评, 已经多次印刷。许多高等院校、职业学校和培训机构也将本书作为教学参考。许多读者提出了宝贵的意见和中肯的建议。根据读者提出的建议, 笔者对本书做了修订。

修订主要从 4 个方面入手。第一, 根据读者反馈, 修正了书中的一些错误, 并且对一些有歧义的地方进行了说明。第二, 理论基础部分删掉了一些过时的内容, 同时对一些算法进行了优化改进。第三, 对全文案例中所涉及的代码, 加粗强调了重点代码行, 并添加了相应的注释行。毕竟, 一本书的篇幅有限, 掌握扎实的基本功不能只靠看书获得, Java 初学者可以针对本书的分类, 充分利用搜索引擎去挖掘更多宝藏。

衷心感谢

本书得以成书, 离不开人民邮电出版社诸位编辑的热心帮助, 他们用心地整理出本书的知识结构, 并为本书制定了初步大纲, 这些工作给笔者提供很大的帮助。

本书写给谁看

如果你是一名 Java 语言的初学者, 并已经学会了 Java 基本的语法, 但却经常在动手编程时感到困难重重, 或者你已经是一个 Java 程序员了, 但在实际开发中却经常感觉力不从心, 那么本书将非常适合你。本书会帮助你找出自己的技术短板, 迅速突破 Java 编程的基本功, 使你成为一名真正的 Java 达人。



2011-09-15



目 录

第 1 课 数组与内存控制	1
1.1 数组初始化	2
1.1.1 Java 数组是静态的	2
1.1.2 数组一定要初始化吗	5
1.1.3 基本类型数组的初始化	7
1.1.4 引用类型数组的初始化	9
1.2 使用数组	12
1.2.1 数组元素就是变量	13
1.2.2 没有多维数组	15
1.3 小结	21
第 2 课 对象与内存控制	22
2.1 实例变量和类变量	23
2.1.1 实例变量和类变量的属性	25
2.1.2 实例变量的初始化时机	27
2.1.3 类变量的初始化时机	32
2.2 父类构造器	34
2.2.1 隐式调用和显式调用	34
2.2.2 访问子类对象的实例变量	37
2.2.3 调用被子类重写的方法	41
2.3 父子实例的内存控制	43

2.3.1 继承成员变量和继承方法的区别	43
2.3.2 内存中子类实例	47
2.3.3 父、子类的类变量	52
2.4 final 修饰符	53
2.4.1 final 修饰的变量	54
2.4.2 执行“宏替换”的变量	59
2.4.3 final 方法不能被重写	65
2.4.4 内部类中的局部变量	67
2.5 小结	70
第 3 课 常见 Java 集合的实现细节	71
3.1 Set 和 Map	72
3.1.1 Set 和 Map 的关系	72
3.1.2 HashMap 和 HashSet	79
3.1.3 TreeMap 和 TreeSet	90
3.2 Map 和 List	98
3.2.1 Map 的 values()方法	98
3.2.2 Map 和 List 的关系	106
3.3 ArrayList 和 LinkedList	107
3.3.1 Vector 和 ArrayList 的区别	109
3.3.2 ArrayList 和 LinkedList 的实现差异	113
3.3.3 ArrayList 和 LinkedList 的性能分析和适用场景	117
3.4 Iterator 迭代器	118
迭代时删除指定元素	120
3.5 小结	123
第 4 课 Java 的内存回收	124
4.1 Java 引用的种类	125
4.1.1 对象在内存中状态	125
4.1.2 强引用	128
4.1.3 软引用	129
4.1.4 弱引用	132
4.1.5 虚引用	136

4.2	Java 的内存泄漏	138
4.3	垃圾回收机制	142
4.3.1	垃圾回收的基本算法	143
4.3.2	堆内存的分代回收	144
4.3.3	与垃圾回收的附加选项	146
4.3.4	常见垃圾回收器	147
4.4	内存管理的小技巧	150
4.4.1	尽量使用直接量	150
4.4.2	使用 StringBuilder 和 StringBuffer 进行字符串连接	151
4.4.3	尽早释放无用对象的引用	151
4.4.4	尽量少用静态变量	152
4.4.5	避免在经常调用的方法、循环中创建 Java 对象	152
4.4.6	缓存经常使用的对象	153
4.4.7	尽量不要使用 finalize 方法	153
4.4.8	考虑使用 SoftReference	154
4.5	小结	154
第 5 课	表达式中的陷阱	155
5.1	关于字符串的陷阱	156
5.1.1	JVM 对字符串的处理	156
5.1.2	不可变的字符串	160
5.1.3	字符串比较	162
5.2	表达式类型的陷阱	164
5.2.1	表达式类型的自动提升	165
5.2.2	复合赋值运算符的陷阱	166
5.3	输入法导致的陷阱	168
5.4	注释的字符必须合法	169
5.5	转义字符的陷阱	170
5.5.1	慎用字符的 Unicode 转义形式	170
5.5.2	中止行注释的转义字符	171
5.6	泛型可能引起的错误	172
5.6.1	原始类型变量的赋值	172
5.6.2	原始类型带来的擦除	175

5.6.3 创建泛型数组的陷阱	178
5.7 正则表达式的陷阱	180
5.8 多线程的陷阱	182
5.8.1 不要调用 run 方法	182
5.8.2 静态的同步方法	183
5.8.3 静态初始化块启动新线程执行初始化	187
5.8.4 注意多线程执行环境	192
5.9 小结	198
第 6 课 流程控制的陷阱	199
6.1 switch 语句陷阱	200
6.1.1 default 分支永远会执行吗	200
6.1.2 break 的重要性	201
6.1.3 switch 表达式的类型	203
6.2 标签引起的陷阱	205
6.3 if 语句的陷阱	206
6.3.1 else 隐含的条件	206
6.3.2 小心空语句	209
6.4 循环体的花括号	212
6.4.1 什么时候可以省略花括号	212
6.4.2 省略花括号的危险	213
6.5 for 循环的陷阱	214
6.5.1 分号惹的祸	214
6.5.2 小心循环计数器的值	218
6.5.3 浮点数作循环计数器	219
6.6 foreach 循环的循环计数器	221
6.7 小结	223
第 7 课 面向对象的陷阱	224
7.1 instanceof 运算符的陷阱	225
7.2 构造器的陷阱	229
7.2.1 构造器之前的 void	230

7.2.2 构造器创建对象吗	231
7.2.3 无限递归的构造器	237
7.3 持有当前类的实例	238
7.4 到底调用哪个重载的方法	240
7.5 方法重写的陷阱	244
7.5.1 重写 private 方法	244
7.5.2 重写其他访问权限的方法	245
7.6 非静态内部类的陷阱	246
7.6.1 非静态内部类的构造器	246
7.6.2 非静态内部类不能拥有静态成员	248
7.6.3 非静态内部类的子类	249
7.7 static 关键字	252
7.7.1 静态方法属于类	252
7.7.2 静态内部类的限制	253
7.8 native 方法的陷阱	254
7.9 小结	256
第 8 课 异常捕捉的陷阱	257
8.1 正确关闭资源的方式	258
8.2 finally 块的陷阱	262
8.2.1 finally 的执行规则	262
8.2.2 finally 块和方法返回值	265
8.3 catch 块的用法	267
8.3.1 catch 块的顺序	267
8.3.2 不要用 catch 代替流程控制	269
8.3.3 只能 catch 可能抛出的异常	270
8.3.4 做点实际的修复	274
8.4 继承得到的异常	276
8.5 小结	278
第 9 课 线性表	279
9.1 线性表概述	280

9.1.1 线性表的定义及逻辑结构	280
9.1.2 线性表的基本操作	281
9.2 顺序存储结构	282
9.3 链式存储结构	288
9.3.1 单链表上的基本运算	289
9.3.2 循环链表	296
9.3.3 双向链表	297
9.4 线性表的分析	306
9.4.1 线性表的实现分析	306
9.4.2 线性表的功能	306
9.5 小结	308
第 10 课 栈和队列	309
10.1 栈	310
10.1.1 栈的基本定义	310
10.1.2 栈的常用操作	311
10.1.3 栈的顺序存储结构及实现	311
10.1.4 栈的链式存储结构及实现	317
10.1.5 Java 集合中的栈	321
10.2 队列	321
10.2.1 队列的基本定义	322
10.2.2 队列的常用操作	322
10.2.3 队列的顺序存储结构及实现	323
10.2.4 循环队列	327
10.2.5 队列的链式存储结构及实现	332
10.2.6 Java 集合中的队列	336
10.3 双向队列	338
10.4 小结	339
第 11 课 树和二叉树	340
11.1 树的概述	341
11.1.1 树的定义和基本术语	341
11.1.2 树的基本操作	343

11.1.3 父节点表示法	343
11.1.4 子节点链表示法	348
11.2 二叉树	354
11.2.1 二叉树的定义和基本概念	354
11.2.2 二叉树的基本操作	355
11.2.3 二叉树的顺序存储	356
11.2.4 二叉树的二叉链表存储	360
11.2.5 二叉树的三叉链表存储	365
11.3 遍历二叉树	369
11.3.1 先序遍历	370
11.3.2 中序遍历	371
11.3.3 后序遍历	371
11.3.4 广度优先(按层)遍历	372
11.4 森林、树和二叉树的转换	373
11.4.1 森林、树和二叉树的转换	374
11.4.2 树的链表存储	375
11.5 哈夫曼树	375
11.5.1 哈夫曼树的定义和基本概念	375
11.5.2 创建哈夫曼树	376
11.5.3 哈夫曼编码	380
11.6 排序二叉树	381
11.7 红黑树	390
11.7.1 插入操作	392
11.7.2 删除操作	394
11.8 小结	407
第 12 课 常用的内部排序	408
12.1 排序的基本概念	409
12.1.1 排序概述	409
12.1.2 内部排序的分类	410
12.2 选择排序法	411
12.2.1 直接选择排序	411
12.2.2 堆排序	415

12.3	交换排序	420
12.3.1	冒泡排序	420
12.3.2	快速排序	423
12.4	插入排序	425
12.4.1	直接插入排序	425
12.4.2	折半插入排序	427
12.4.3	Shell 排序	429
12.5	归并排序	433
12.6	桶式排序	437
12.7	基数排序	440
12.8	小结	443
第 13 课	程序开发	444
13.1	扎实的基本功	445
13.1.1	快速的输入能力	445
13.1.2	编程实现能力	447
13.1.3	快速排错	447
13.2	程序开发之前	448
13.2.1	分析软件的组件模型	448
13.2.2	建立软件的数据模型	451
13.3	弄清程序的具体实现	452
13.3.1	各组件如何通信	452
13.3.2	人机交互的实现	454
13.3.3	复杂算法的分析	457
13.4	编写开发文档	459
13.4.1	绘制建模图、流程图	460
13.4.2	提供简要说明	461
13.4.3	编写伪码实现	462
13.5	编码实现和开发心态	462
13.5.1	开发是复杂的	462
13.5.2	开发过程是漫长的	463
13.6	小结	463

第 14 课 程序调试	464
14.1 程序的可调试性.....	465
14.1.1 增加注释.....	465
14.1.2 使用 log.....	466
14.2 程序调试的基本方法.....	474
14.2.1 借助编译器的代码审查.....	475
14.2.2 跟踪程序执行流程.....	477
14.2.3 断点调试.....	479
14.2.4 隔离调试.....	481
14.2.5 错误重现.....	482
14.3 记录常见错误.....	484
14.3.1 常见异常可能的错误原因.....	485
14.3.2 常见运行时异常可能的错误原因.....	486
14.4 程序调试的整体思路.....	488
14.4.1 分段调试.....	488
14.4.2 分模块调试.....	490
14.5 调试心态.....	490
14.5.1 谁都会出错.....	490
14.5.2 调试比写程序更费时.....	491
14.6 小结.....	491
第 15 课 使用 IDE 工具	492
15.1 何时开始利用 IDE 工具.....	493
15.2 IDE 工具概述.....	494
15.2.1 IDE 工具的基本功能.....	495
15.2.2 常见的 Java IDE 工具.....	497
15.3 项目管理.....	499
15.3.1 建立项目.....	499
15.3.2 自动编译.....	504
15.3.3 自动部署、运行.....	504
15.4 代码管理.....	505

15.4.1	向导式的代码生成	506
15.4.2	代码生成器	508
15.4.3	代码提示	509
15.4.4	自动代码补齐	510
15.4.5	实时错误提示	510
15.5	项目调试	511
15.5.1	设置断点	512
15.5.2	单步调试	514
15.5.3	步入、步出	514
15.6	团队协作功能	516
	作为版本控制工具的客户端	516
15.7	小结	519
第 16 课	软件测试	520
16.1	软件测试概述	521
16.1.1	软件测试的概念和目的	521
16.1.2	软件测试的分类	523
16.1.3	开发活动和测试活动	524
16.1.4	常见的 Bug 管理工具	524
16.2	单元测试	525
16.2.1	单元测试概述	525
16.2.2	单元测试的逻辑覆盖	527
16.2.3	JUnit 介绍	530
16.2.4	JUnit 的用法	531
16.3	系统测试和自动化测试	538
16.3.1	系统测试概述	538
16.3.2	自动化测试	539
16.3.3	常见自动化测试工具	540
16.4	性能测试	541
16.4.1	性能测试概述	541
16.4.2	性能测试的相关概念	542
16.4.3	常见性能测试工具	543
16.5	小结	544

第 1 课

数组与内存控制

一家国际著名软件企业的面试。

“你的简历我看了，你会使用 Java？”面试官面无表情地问道。

“是的。”参加面试的人，胸有成竹地回答。

“那好，你给我叙述一下，在 Java 中，声明一个数组的过程中，是如何分配内存的。”

“.....”

“那，Java 数组的初始化一共哪有几种方式，你能说一说吗？”

“.....”

“那你知道基本类型数组和引用类型数组之间，在初始化时的内存分配机制有什么区别吗？”

“.....”

过了一会儿之后，房间的门打开了，可怜的面试者，狼狈地走了出来。

离开的时候，他喃喃自语：“原来，小小的数组，也有这么多的知识。”

本课要点：

- Java 数组的基本语法；
- Java 数组的静态特性；
- Java 数组的内存分配机制；
- 初始化 Java 数组的两种方式；
- 初始化基本类型数组的内存分配；
- 初始化应用类型数组的内存分配；
- 数组引用变量和数组对象；
- 何时是数组引用变量，何时是数组对象；
- 数组元素等同于变量；
- 多维数组的内存分配。