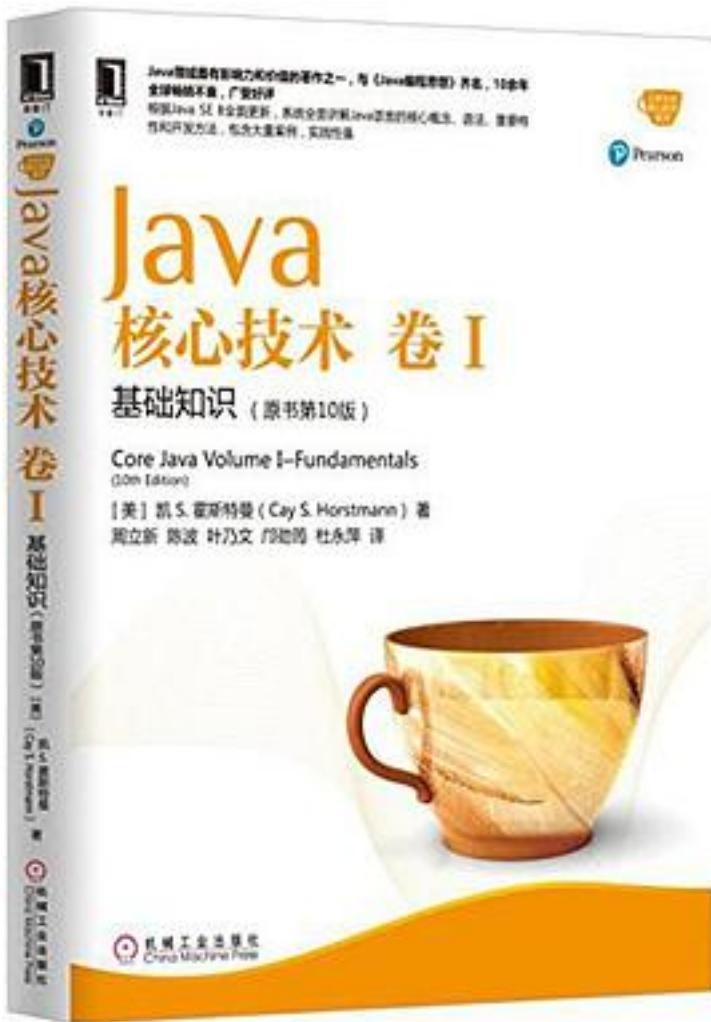


Java核心技术 · 卷 I (原书第10版)



[Java核心技术 · 卷 I \(原书第10版\) 下载链接1](#)

著者:[美] 凯.S.霍斯特曼 (Cay S. Horstmann)

出版者:机械工业出版社

出版时间:2016-9

装帧:平装

isbn:9787111547426

Java领域最有影响力和价值的著作之一，由拥有20多年教学与研究经验的资深Java技

术专家撰写(获Jolt大奖)，与《Java编程思想》齐名，10余年全球畅销不衰，广受好评。第10版根据Java SE 8全面更新，同时修正了第9版中的不足，系统全面讲解了Java语言的核心概念、语法、重要特性和开发方法，包含大量案例，实践性强。

一直以来，《Java核心技术》都被认为是面向高级程序员的经典教程和参考书，它内容翔实、客观准确，不拖泥带水，是想为实际应用编写健壮Java代码的程序员的首选。如今，本版进行了全面更新，以反映近年来人们翘首以待、变革最大的Java版本（Java SE 8）的内容。这一版经过重写，并重新组织，全面阐释了新的Java SE 8特性、惯用法和*佳实践，其中包含数百个示例程序，所有这些代码都经过精心设计，不仅易于理解，也很容易实际应用。

本书为专业程序员解决实际问题而写，可以帮助你深入了解Java语言和库。在卷I中，Horstmann主要强调基本语言概念和现代用户界面编程基础，深入介绍了从Java面向对象编程到泛型、集合、lambda表达式、Swing UI设计以及并发和函数式编程的*新方法等内容。

作者介绍：

凯 S. 霍斯特曼 (Cay S. Horstmann)，圣何塞州立大学计算机科学系教授、Java的倡导者，经常在开发人员会议上发表演讲。他是《Core Java for the Impatient》(2015)、《Java SE 8 for the Really Impatient》(2014) 和《Scala for the Impatient》(2012) 的作者，这些书均由Addison-Wesley出版。他为专业程序员和计算机科学专业学生编写过数十本图书。

目录: 译者序

前言

致谢

第1章 Java程序设计概述 1

1.1 Java程序设计平台 1

1.2 Java “白皮书”的关键术语 2

1.2.1 简单性 2

1.2.2 面向对象 2

1.2.3 分布式 3

1.2.4 健壮性 3

1.2.5 安全性 3

1.2.6 体系结构中立 4

1.2.7 可移植性 4

1.2.8 解释型 5

1.2.9 高性能 5

1.2.10 多线程 5

1.2.11 动态性 5

1.3 Java applet与Internet 6

1.4 Java发展简史 7

1.5 关于Java的常见误解 9

第2章 Java程序设计环境 12

2.1 安装Java开发工具包 12

2.1.1 下载JDK 12

2.1.2 设置JDK 13

2.1.3 安装库源文件和文档 15

2.2 使用命令行工具 16

2.3 使用集成开发环境 18
2.4 运行图形化应用程序 21
2.5 构建并运行applet 23
第3章 Java的基本程序设计结构 28
3.1 一个简单的Java应用程序 28
3.2 注释 31
3.3 数据类型 32
3.3.1 整型 32
3.3.2 浮点类型 33
3.3.3 char类型 34
3.3.4 Unicode和char类型 35
3.3.5 boolean类型 35
3.4 变量 36
3.4.1 变量初始化 37
3.4.2 常量 37
3.5 运算符 38
3.5.1 数学函数与常量 39
3.5.2 数值类型之间的转换 40
3.5.3 强制类型转换 41
3.5.4 结合赋值和运算符 42
3.5.5 自增与自减运算符 42
3.5.6 关系和boolean运算符 42
3.5.7 位运算符 43
3.5.8 括号与运算符级别 44
3.5.9 枚举类型 45
3.6 字符串 45
3.6.1 子串 45
3.6.2 拼接 46
3.6.3 不可变字符串 46
3.6.4 检测字符串是否相等 47
3.6.5 空串与Null串 48
3.6.6 码点与代码单元 49
3.6.7 String API 50
3.6.8 阅读联机API文档 52
3.6.9 构建字符串 54
3.7 输入输出 55
3.7.1 读取输入 55
3.7.2 格式化输出 58
3.7.3 文件输入与输出 61
3.8 控制流程 63
3.8.1 块作用域 63
3.8.2 条件语句 63
3.8.3 循环 66
3.8.4 确定循环 69
3.8.5 多重选择：switch语句 72
3.8.6 中断控制流程语句 74
3.9 大数值 76
3.10 数组 78
3.10.1 for each循环 79
3.10.2 数组初始化以及匿名数组 80
3.10.3 数组拷贝 81
3.10.4 命令行参数 81
3.10.5 数组排序 82
3.10.6 多维数组 85

- 3.10.7 不规则数组 88
- 第4章 对象与类 91
 - 4.1 面向对象程序设计概述 91
 - 4.1.1 类 92
 - 4.1.2 对象 93
 - 4.1.3 识别类 93
 - 4.1.4 类之间的关系 94
 - 4.2 使用预定义类 95
 - 4.2.1 对象与对象变量 95
 - 4.2.2 Java类库中的LocalDate类 98
 - 4.2.3 更改器方法与访问器方法 100
 - 4.3 用户自定义类 103
 - 4.3.1 Employee类 103
 - 4.3.2 多个源文件的使用 105
 - 4.3.3 剖析Employee类 106
 - 4.3.4 从构造器开始 106
 - 4.3.5 隐式参数与显式参数 108
 - 4.3.6 封装的优点 109
 - 4.3.7 基于类的访问权限 111
 - 4.3.8 私有方法 111
 - 4.3.9 final实例域 112
 - 4.4 静态域与静态方法 112
 - 4.4.1 静态域 112
 - 4.4.2 静态常量 113
 - 4.4.3 静态方法 114
 - 4.4.4 工厂方法 115
 - 4.4.5 main方法 115
 - 4.5 方法参数 118
 - 4.6 对象构造 123
 - 4.6.1 重载 123
 - 4.6.2 默认域初始化 123
 - 4.6.3 无参数的构造器 124
 - 4.6.4 显式域初始化 125
 - 4.6.5 参数名 125
 - 4.6.6 调用另一个构造器 126
 - 4.6.7 初始化块 127
 - 4.6.8 对象析构与finalize方法 130
 - 4.7 包 131
 - 4.7.1 类的导入 131
 - 4.7.2 静态导入 133
 - 4.7.3 将类放入包中 133
 - 4.7.4 包作用域 136
 - 4.8 类路径 137
 - 4.8.1 设置类路径 139
 - 4.9 文档注释 140
 - 4.9.1 注释的插入 140
 - 4.9.2 类注释 140
 - 4.9.3 方法注释 141
 - 4.9.4 域注释 142
 - 4.9.5 通用注释 142
 - 4.9.6 包与概述注释 143
 - 4.9.7 注释的抽取 143
 - 4.10 类设计技巧 144
 - 第5章 继承 147

5.1 类、超类和子类 147
5.1.1 定义子类 147
5.1.2 覆盖方法 149
5.1.3 子类构造器 150
5.1.4 继承层次 153
5.1.5 多态 154
5.1.6 理解方法调用 155
5.1.7 阻止继承：final类和方法 157
5.1.8 强制类型转换 158
5.1.9 抽象类 160
5.1.10 受保护访问 165
5.2 Object：所有类的超类 166
5.2.1 equals方法 166
5.2.2 相等测试与继承 167
5.2.3 hashCode方法 170
5.2.4 toString方法 172
5.3 泛型数组列表 178
5.3.1 访问数组列表元素 180
5.3.2 类型化与原始数组列表的兼容性 183
5.4 对象包装器与自动装箱 184
5.5 参数数量可变的方法 187
5.6 枚举类 188
5.7 反射 190
5.7.1 Class类 190
5.7.2 捕获异常 192
5.7.3 利用反射分析类的能力 194
5.7.4 在运行时使用反射分析对象 198
5.7.5 使用反射编写泛型数组代码 202
5.7.6 调用任意方法 205
5.8 继承的设计技巧 208
第6章 接口、lambda表达式与内部类 211
6.1 接口 211
6.1.1 接口概念 211
6.1.2 接口的特性 217
6.1.3 接口与抽象类 218
6.1.4 静态方法 218
6.1.5 默认方法 219
6.1.6 解决默认方法冲突 220
6.2 接口示例 222
6.2.1 接口与回调 222
6.2.2 Comparator接口 224
6.2.3 对象克隆 225
6.3 lambda表达式 231
6.3.1 为什么引入lambda表达式 231
6.3.2 lambda表达式的语法 232
6.3.3 函数式接口 234
6.3.4 方法引用 235
6.3.5 构造器引用 237
6.3.6 变量作用域 237
6.3.7 处理lambda表达式 239
6.3.8 再谈Comparator 242
6.4 内部类 242
6.4.1 使用内部类访问对象状态 244
6.4.2 内部类的特殊语法规则 247

6.4.3 内部类是否有用、必要和安全 248

6.4.4 局部内部类 250

6.4.5 由外部方法访问变量 250

6.4.6 匿名内部类 252

6.4.7 静态内部类 255

6.5 代理 258

6.5.1 何时使用代理 259

6.5.2 创建代理对象 259

6.5.3 代理类的特性 262

第7章 异常、断言和日志 264

7.1 处理错误 264

7.1.1 异常分类 265

7.1.2 声明受查异常 267

7.1.3 如何抛出异常 269

7.1.4 创建异常类 270

7.2 捕获异常 271

7.2.1 捕获异常 271

7.2.2 捕获多个异常 273

7.2.3 再次抛出异常与异常链 274

7.2.4 finally子句 275

7.2.5 带资源的try语句 278

7.2.6 分析堆栈轨迹元素 280

7.3 使用异常机制的技巧 282

7.4 使用断言 285

7.4.1 断言的概念 285

7.4.2 启用和禁用断言 286

7.4.3 使用断言完成参数检查 287

7.4.4 为文档假设使用断言 288

7.5 记录日志 289

7.5.1 基本日志 289

7.5.2 高级日志 289

7.5.3 修改日志管理器配置 291

7.5.4 本地化 292

7.5.5 处理器 293

7.5.6 过滤器 296

7.5.7 格式化器 296

7.5.8 日志记录说明 296

7.6 调试技巧 304

第8章 泛型程序设计 309

8.1 为什么要使用泛型程序设计 309

8.1.1 类型参数的好处 309

8.1.2 谁想成为泛型程序员 310

8.2 定义简单泛型类 311

8.3 泛型方法 313

8.4 类型变量的限定 314

8.5 泛型代码和虚拟机 316

8.5.1 类型擦除 316

8.5.2 翻译泛型表达式 317

8.5.3 翻译泛型方法 318

8.5.4 调用遗留代码 319

8.6 约束与局限性 320

8.6.1 不能用基本类型实例化类型参数 320

8.6.2 运行时类型查询只适用于原始类型 321

8.6.3 不能创建参数化类型的数组 321

8.6.4 Varargs警告	322
8.6.5 不能实例化类型变量	323
8.6.6 不能构造泛型数组	323
8.6.7 泛型类的静态上下文中类型变量无效	325
8.6.8 不能抛出或捕获泛型类的实例	325
8.6.9 可以消除对受查异常的检查	326
8.6.10 注意擦除后的冲突	327
8.7 泛型类型的继承规则	328
8.8 通配符类型	330
8.8.1 通配符概念	330
8.8.2 通配符的超类型限定	331
8.8.3 无限定通配符	334
8.8.4 通配符捕获	334
8.9 反射和泛型	337
8.9.1 泛型Class类	337
8.9.2 使用Class参数进行类型匹配	338
8.9.3 虚拟机中的泛型类型信息	338
第9章 集合	344
9.1 Java集合框架	344
9.1.1 将集合的接口与实现分离	344
9.1.2 Collection接口	346
9.1.3 迭代器	347
9.1.4 泛型实用方法	349
9.1.5 集合框架中的接口	352
9.2 具体的集合	353
9.2.1 链表	355
9.2.2 数组列表	362
9.2.3 散列集	363
9.2.4 树集	366
9.2.5 队列与双端队列	369
9.2.6 优先级队列	371
9.3 映射	372
9.3.1 基本映射操作	372
9.3.2 更新映射项	375
9.3.3 映射视图	376
9.3.4 弱散列映射	377
9.3.5 链接散列集与映射	378
9.3.6 枚举集与映射	379
9.3.7 标识散列映射	380
9.4 视图与包装器	381
9.4.1 轻量级集合包装器	382
9.4.2 子范围	382
9.4.3 不可修改的视图	383
9.4.4 同步视图	384
9.4.5 受查视图	384
9.4.6 关于可选操作的说明	385
9.5 算法	388
9.5.1 排序与混排	389
9.5.2 二分查找	391
9.5.3 简单算法	392
9.5.4 批操作	394
9.5.5 集合与数组的转换	394
9.5.6 编写自己的算法	395
9.6 遗留的集合	396

9.6.1 Hashtable类 397
9.6.2 枚举 397
9.6.3 属性映射 398
9.6.4 栈 399
9.6.5 位集 399
第10章 图形程序设计 403
10.1 Swing概述 403
10.2 创建框架 407
10.3 框架定位 409
10.3.1 框架属性 411
10.3.2 确定合适的框架大小 411
10.4 在组件中显示信息 415
10.5 处理2D图形 419
10.6 使用颜色 426
10.7 文本使用特殊字体 429
10.8 显示图像 435
第11章 事件处理 439
11.1 事件处理基础 439
11.1.1 实例：处理按钮点击事件 441
11.1.2 简洁地指定监听器 445
11.1.3 实例：改变观感 447
11.1.4 适配器类 450
11.2 动作 453
11.3 鼠标事件 459
11.4 AWT事件继承层次 465
11.4.1 语义事件和底层事件 466
第12章 Swing用户界面组件 469
12.1 Swing和模型-视图-控制器设计模式 469
12.1.1 设计模式 469
12.1.2 模型-视图-控制器模式 470
12.1.3 Swing按钮的模型-视图-控制器分析 473
12.2 布局管理概述 474
12.2.1 边框布局 477
12.2.2 网格布局 478
12.3 文本输入 481
12.3.1 文本域 482
12.3.2 标签和标签组件 483
12.3.3 密码域 484
12.3.4 文本区 485
12.3.5 滚动窗格 485
12.4 选择组件 488
12.4.1 复选框 488
12.4.2 单选钮 490
12.4.3 边框 493
12.4.4 组合框 496
12.4.5 滑动条 499
12.5 菜单 504
12.5.1 菜单创建 504
12.5.2 菜单项中的图标 507
12.5.3 复选框和单选钮菜单项 508
12.5.4 弹出菜单 508
12.5.5 快捷键和加速器 510
12.5.6 启用和禁用菜单项 511
12.5.7 工具栏 515

12.5.8 工具提示 516
12.6 复杂的布局管理 518
12.6.1 网格组布局 520
12.6.2 组布局 528
12.6.3 不使用布局管理器 537
12.6.4 定制布局管理器 537
12.6.5 遍历顺序 541
12.7 对话框 541
12.7.1 选项对话框 542
12.7.2 创建对话框 551
12.7.3 数据交换 554
12.7.4 文件对话框 559
12.7.5 颜色选择器 569
12.8 GUI程序排错 573
12.8.1 调试技巧 573
12.8.2 让AWT机器人完成工作 576
第13章 部署Java应用程序 580
13.1 JAR文件 580
13.1.1 创建JAR文件 580
13.1.2 清单文件 581
13.1.3 可执行JAR文件 582
13.1.4 资源 583
13.1.5 密封 585
13.2 应用首选项的存储 586
13.2.1 属性映射 586
13.2.2 首选项API 591
13.3 服务加载器 596
13.4 applet 598
13.4.1 一个简单的applet 599
13.4.2 applet HTML标记和属性 602
13.4.3 使用参数向applet传递信息 603
13.4.4 访问图像和音频文件 608
13.4.5 applet上下文 609
13.4.6 applet间通信 609
13.4.7 在浏览器中显示信息项 610
13.4.8 沙箱 611
13.4.9 签名代码 612
13.5 Java Web Start 614
13.5.1 发布Java Web Start应用 614
13.5.2 JNLP API 617
第14章 并发 624
14.1 什么是线程 624
14.1.1 使用线程给其他任务提供机会 629
14.2 中断线程 632
14.3 线程状态 635
14.3.1 新创建线程 635
14.3.2 可运行线程 635
14.3.3 被阻塞线程和等待线程 636
14.3.4 被终止的线程 636
14.4 线程属性 638
14.4.1 线程优先级 638
14.4.2 守护线程 639
14.4.3 未捕获异常处理器 639
14.5 同步 640

14.5.1 竞争条件的一个例子 641
14.5.2 竞争条件详解 644
14.5.3 锁对象 646
14.5.4 条件对象 648
14.5.5 synchronized关键字 653
14.5.6 同步阻塞 656
14.5.7 监视器概念 657
14.5.8 Volatile域 658
14.5.9 final变量 659
14.5.10 原子性 659
14.5.11 死锁 661
14.5.12 线程局部变量 663
14.5.13 锁测试与超时 665
14.5.14 读/写锁 666
14.5.15 为什么弃用stop和suspend方法 667
14.6 阻塞队列 668
14.7 线程安全的集合 673
14.7.1 高效的映射、集和队列 674
14.7.2 映射条目的原子更新 675
14.7.3 对并发散列映射的批操作 676
14.7.4 并发集视图 678
14.7.5 写数组的拷贝 679
14.7.6 并行数组算法 679
14.7.7 较早的线程安全集合 680
14.8 Callable与Future 681
14.9 执行器 685
14.9.1 线程池 685
14.9.2 预定执行 689
14.9.3 控制任务组 690
14.9.4 Fork-Join框架 691
14.9.5 可完成Future 694
14.10 同步器 696
14.10.1 信号量 696
14.10.2 倒计时门栓 697
14.10.3 障栅 697
14.10.4 交换器 698
14.10.5 同步队列 698
14.11 线程与Swing 698
14.11.1 运行耗时的任务 699
14.11.2 使用Swing工作线程 703
14.11.3 单一线程规则 708
附录A Java关键字 710
· · · · · (收起)

[Java核心技术 · 卷 I \(原书第10版\)](#) [下载链接1](#)

标签

Java

编程

计算机

JAVA

基础

java

计算机科学

技术

评论

不忘黑C++，swing内容太多，而且一本讲语言的书，没有习题真的是不行的。

是入门书，但是有些地方还是讲得不是很清晰。适合有编程基础的自学用。

通勤地铁上读完，最好的Java入门书，扣掉一星是因为Swing、AWT、Applet的内容完全没必要加在这本书里

原著作者的境界比《Thinking in Java》low 好几个档次。为了码字骗钱，也是够拼的。

C.S.Horstmann是SJSU的教授 传说中的硅谷腹地就业神校 == 所以你看他写的书
并不是特别权威深入 但就是特别畅销 特别接地气 跟着java的版本一起更新
每次更新版都大卖 这比书里的内容本身更值得我学习(..)a

读得原版，有C++基础就当语法知识入门，太深的知识还是需要其它书籍，不过swing和GUI确实很多余，这年头谁还会用java来做前端

太厚，没习题，不容易记住

有与C++比较的点

一半的内容都不用看...

很好的入门书籍，特别是有C++基础的人，文中有特别指出一些Java和C++之间的区别概念之类的

建议和英文原版一起看。

在读它的英文版。内容真的是很入门了，感觉更适合编程新手。《java编程思想》才是真爱啊。不要再拿来对比了。

翻译根本就不懂啥时候该英文，啥时候该中文。对我来说，看原版或许比较好。另外，java库向后兼容做得不错，导致库有些恶心。我没有大规模项目经验，所以我觉得c++的模板确实比java泛型更好。正所谓，keep it simple, stupid。

并发那一部分安排的不好，这是第二遍读了。

Java 学习必读之书。

看着看着就看不懂了, 作为一本入门书竟然不够基础

部分章节晦涩难懂, 不过整体不错。作者站的视角比较高。

so many

看swing部分看得我发慌

不适合作为第一本学习Java的书籍。对于常用类库涉及较少, 不过字字珠玑, 好好琢磨会有更深的理解。没有练习题可惜了, 应该把书中的代码都实现一遍。

[Java核心技术 · 卷 I \(原书第10版\) 下载链接1](#)

书评

写个大纲, 希望有所帮助:

1-6章: 概述和基本语法, 为便于C++程序员迁移, 附有大量C++对比注释, 还有各种“注意”“警告”的tips, 这编排很好。

7-9章: Java基于awt和swing包的界面开发由于缺少良好的IDE支持, 界面设计和布局问题让习惯了VS的初学者很烦躁, 789可跳读, 核心...

学习Java有两部很好的书籍: 《Thinking in Java》和《Core Java》。前者内容丰富, 在讲述Java语言的同时把Java编程思想也娓娓道来, 让人如沐春风; 后者偏重于讲解Java技术, 举例说明了大量API的使用。这两部书在讲解Java上殊途同归, 都值得放在书桌上时常翻阅。不过看完后比...

每次我的朋友来问我，学java看什么书好，我总是说：《Thinking in Java》是你一定要看的，但不是现在，那本书不适合入门。要入门，来，借这本书去吧，然后递给他这本《Core Java》（卷一）。Java十年，发展如此迅速，J2EE之火热，令我大感惊讶，我为我几年前选择学Java刚到庆...

卷1+卷2就是一套JavaSE的大百科全书了。看到《Java核心技术》新版更新已经出到第8版时，虽不狂喜，却也欣喜异常的。作为一本力求讲解全部java基本知识的经典书籍，研读相关细节，所获心得颇多。从jdk的下载安装到部署应用程序；从数据类型控制流程到对象与类再到集合；从图形...

这本使JAVA里面最重要的一本书，是我开始JAVA之旅的第一本书，希望大家能够好好研读一下，可以学到很多的东西，这本书所涵盖的内容比其他的要多地多，即包括基础知识，又包括高级的应用，不论你是自己编程还是做企业级的开发系统，我想这本书可以给你足够的参考。我那时看得是...

ActionScript和JAVA真的很像，很多类和常用方法是完全相同的。
第1章建议详看，因为这样可以对JAVA多一些了解；
第2章很短，可以详看，同时应该删除Flash
Builder独立版，改装插件版。因为独立版不支持JAVA开发，需要自己装一堆插件才行。
。在国内的网络环境下，去在线装哪些插件...

本书的内容很全面，但是不得不吐槽一下翻译，比如在P231，倒数第8行，“这个方法的调用者可以34使用……”，推测这里的34应该是没有用的，这种错误太低级了。另外翻译的表达不清晰，有效的降低了我对本书的理解，建议大家看原版

这本使JAVA里面最重要的一本书，是我开始JAVA之旅的第一本书，希望大家能够好好研读一下，可以学到很多的东西，这本书所涵盖的内容比其他的要多地多，即包括基础知识，又包括高级的应用，不论你是自己编程还是做企业级的开发系统，我想这本书可以给你足够的参考。我那时看得是...

我从来没有看过一本这样轻松的计算机的书，我是有生以来第一次读到这么好的翻译的书。在我对翻译的书绝对无望的时候，它力挽狂澜的告诉我，事情还没有坏到这个地步。

我以前发誓此生不再看XXX人编著的书，不再看XXXX等人和著的书，我只看XXX著的书。但看过了core java 之后...

该书第十版GUI仍在讲Swing

并发一章是该书缺陷的典型例子，内容包含了不同年代实现的大量并发编程的不同方法，却完全未指出当前大部分工程主要采用的措施，没有各类情况采取何种措施的指南，更没有不同方法间的性能对比，极其枯燥而缺乏实际价值。该章还出现了给出一大段程序然...

还要写题目，总感觉是在苦思冥想一个难以命名的变量，思考出来的结果又是一股高考作文的既视感。

作为一本入门参考书，绝对比国内现在的大部分教材要优秀得多。对于Java初学者来说，也是一本值得时常翻阅的技术手册。书中有较多地方进行java与c++的比较，这是很有...

这是一本讲的很详细的书，既可以用作入门书，也可以当手册，需要的时候查询。

700页的书，基本每个例子都敲了一遍，看完之后很有成就感。

当然，只看书是不够的，关键还是实践；基本功很重要，但是用不上的基本功就是废柴。加油！准备开...

很基础，目前看到图形界面350页，适合初学者的一本java教材。前面的基础讲解的不错。后面的图形界面的有说明也有应用的实例，不知道后面的异常和多线程内容怎么样，看前面的部分来推测应该不错。建议学Java的看看。

《java核心技术(卷1)》书评

最近花了一个多星期的时间看了《Java核心技术》卷一，感觉这本书是一本介绍java内容比较全面的一本书，几乎所有Java的知识点都有涉及，而且讲解的也很详细，对java的初学者来说是一本很好的入门书籍。该书对java中jdk的下载和安装部署都进行了详细的...

core

java内容翔实，丰富，生动，就算对于java已经很熟悉的人手头上一本虽是用来查阅

也是十分方便的。在看完core java后去品味TIJ会有令眼看世界的感觉。

我发现这本书对加强Java基本概念的理解非常有帮助。它是为有编程经验的人设计的，所以不会涉及令人痛苦的细节，但它确实帮我厘清我一直纠结的基础概念。阅读这本书花了我两天时间，我一定会买卷II的。感谢花时间撰写这种高质量图书的程序员。我用C++，所以C++注释部分对我非常...

《Java核心技术卷一》是唯一可以和《Java编程思想》媲美的一本Java入门书。单从技术的角度来看，前者更好一些。但上升到思想层面嘛，自然后者更好，两者的偏重点不同。思想不会变，所以《Java编程思想》还停留在第4版（2007年）；而技术是要更新迭代的，所以《Java核心...

在上册的4.10 Class Design Hints小节，缺少第5部分。（P200）

断断续续快看完第九版第一卷英文版。刚开始时候，感觉这本书很贴近实际，开门见山便亮出正统完整的类似helloworld的例子，以及简单GUI的例子。在简洁地介绍java开发环境后，详细介绍一个完整的java程序涉及到的元素，充满了实用气息。然而随着深入章节，到了第6章，以及第9章...

[Java核心技术 · 卷 I \(原书第10版\) 下载链接1](#)