

Java面向对象编程



[Java面向对象编程_下载链接1](#)

著者:孙卫琴

出版者:电子工业出版社

出版时间:2006-7

装帧:简装本

isbn:9787121025389

《Java 面向对象编程》内容由浅入深，紧密结合实际，利用大量典型实例，详细讲解Java面向

对象的编程思想、编程语法和设计模式，介绍常见Java类库的用法，总结优化Java编程的各种宝贵经验，深入阐述Java虚拟机执行Java程序的原理。书中实例均基于最新的JDK

1.5版本。全书内容包括面向对象的编程思想、Java语言的基础知识、异常处理、类与对象的生命周期、多线程、Java集合、输入/输出和GUI编程等。其最大特色是以六条主线贯穿全书：面向对象编程思想、Java语言的语法、Java虚拟机执行Java程序的原理、在实际项目中的运用、设计模式和性能优化技巧。

作者介绍:

孙卫琴，1975年出生，籍贯江苏常州，1997年毕业于上海交通大学，随后主要从事基于Java的软件开发工作，在此期间还从事过Java方面的培训和咨询工作。2002年开始Java技术领域的图书创作。

代表著作有：《Java面向对象编程》、《精通Hibernate:Java对象持久化技术详解》、《精通Struts：基于MVC的Java Web设计与开发》、《Tomcat与Java Web开发技术详解》。

目录: 第1章 面向对象开发方法概述.

1.1 结构化的软件开发方法简介

1.2 面向对象的软件开发方法简介

1.2.1 对象模型

1.2.2 UML：可视化建模语言

1.2.3 Rational Rose：可视化建模工具

1.3 面向对象开发中的核心思想和概念

1.3.1 问题领域、对象、属性、状态、行为、方法、实现

1.3.2 类、类型

1.3.3 消息、服务

1.3.4 接口

1.3.5 封装、透明

1.3.6 抽象

1.3.7 继承、扩展、覆盖

1.3.8 组合

1.3.9 多态、动态绑定

1.4 UML语言简介

1.4.1 用例图

1.4.2 类框图

1.4.3 时序图

1.4.4 协作图

1.4.5 状态转换图

1.4.6 组件图

1.4.7 部署图

1.5 类之间的关系

1.5.1 关联 (Association)

1.5.2 依赖 (Dependency)

1.5.3 聚集 (Aggregation)

1.5.4 泛化 (Generalization)

1.5.5 实现 (Realization)

1.5.6 区分依赖、关联和聚集关系

1.6 实现Panel系统

1.6.1 扩展Panel系统

1.6.2 用配置文件进一步提高Panel系统的可维护性

1.6.3 运行Panel系统

- 1.7 小结
- 1.8 思考题
- 第2章 第一个Java应用
- 2.1 创建Java源文件
 - 2.1.1 Java源文件结构
 - 2.1.2 包声明语句
 - 2.1.3 包引入语句
 - 2.1.4 方法的声明
 - 2.1.5 程序入口main()方法的声明
 - 2.1.6 给main()方法传递参数
 - 2.1.7 注释语句
 - 2.1.8 关键字
 - 2.1.9 标识符
 - 2.1.10 编程规范
- 2.2 用JDK管理Java应用
 - 2.2.1 JDK简介
 - 2.2.2 编译Java源文件
 - 2.2.3 运行Java程序
 - 2.2.4 给Java应用打包
- 2.3 JavaDoc文档
 - 2.3.1 JavaDoc标记
 - 2.3.2 javadoc命令的用法
- 2.4 Java虚拟机的运行时数据区
- 2.5 小结
- 2.6 思考题
- 第3章 数据类型和变量
- 3.1 基本数据类型
 - 3.1.1 boolean类型
 - 3.1.2 byte、short、int和long类型
 - 3.1.3 char类型与字符编码
 - 3.1.4 float和double类型
- 3.2 引用类型
 - 3.2.1 基本类型与引用类型的区别
 - 3.2.2 用new关键字创建对象
- 3.3 变量的作用域
 - 3.3.1 静态变量和实例变量的生命周期
 - 3.3.2 局部变量的生命周期
 - 3.3.3 成员变量和局部变量同名
 - 3.3.4 将局部变量的作用域最小化
- 3.4 对象的默认引用：this
- 3.5 参数传递
- 3.6 变量的初始化及默认值
 - 3.6.1 成员变量的初始化
 - 3.6.2 局部变量的初始化
- 3.7 直接数
 - 3.7.1 直接数的类型
 - 3.7.2 直接数的赋值
- 3.8 小结
- 3.9 思考题
- 第4章 操作符
- 4.1 操作符简介
- 4.2 整型操作符
 - 4.2.1 一元整型操作符
 - 4.2.2 二元整型操作符

- 4.3 浮点型操作符
- 4.4 比较操作符和逻辑操作符
- 4.5 三元操作符 “?:”
- 4.6 字符串连接操作符 “+”
- 4.7 操作符 “==” 与对象的equals()方法
 - 4.7.1 操作符 “==”
 - 4.7.2 对象的equals()方法
- 4.8 instanceof 操作符
- 4.9 变量的赋值和类型转换
 - 4.9.1 基本数据类型转换
 - 4.9.2 引用类型的类型转换
- 4.10 小结
- 4.11 思考题
- 第5章 流程控制
 - 5.1 分支语句
 - 5.1.1 if...else语句
 - 5.1.2 switch语句
 - 5.2 循环语句
 - 5.2.1 while语句
 - 5.2.2 do...while语句
 - 5.2.3 for语句
 - 5.2.4 多重循环
 - 5.3 流程跳转语句
 - 5.4 综合例子：八皇后问题
 - 5.5 小结
 - 5.6 思考题
- 第6章 继承
 - 6.1 继承的基本语法
 - 6.2 方法重载 (Overload)
 - 6.3 方法覆盖 (Override)
 - 6.4 方法覆盖与方法重载的异同
 - 6.5 super关键字
 - 6.6 多态
 - 6.7 继承的利弊和使用原则
 - 6.7.1 继承树的层次不可太多
 - 6.7.2 继承树的上层为抽象层
 - 6.7.3 继承关系最大的弱点：打破封装
 - 6.7.4 精心设计专门用于被继承的类
 - 6.7.5 区分对象的属性与继承
 - 6.8 比较组合与继承
 - 6.8.1 组合关系的分解过程对应继承关系的抽象过程
 - 6.8.2 组合关系的组合过程对应继承关系的扩展过程
 - 6.9 小结
 - 6.9 思考题
- 第7章 Java语言中的修饰符
 - 7.1 访问控制修饰符
 - 7.2 abstract修饰符
 - 7.3 final修饰符
 - 7.3.1 final类
 - 7.3.2 final方法
 - 7.3.3 final变量
 - 7.4 static修饰符
 - 7.4.1 static变量
 - 7.4.2 static方法

7.4.3 static代码块

7.5 小结

7.6 思考题

第8章 接口

8.1 接口的概念和基本特征

8.2 比较抽象类与接口

8.3 与接口相关的设计模式

8.3.1 定制服务模式

8.3.2 适配器模式

8.3.3 默认适配器模式

8.3.4 代理模式

8.3.5 标识类型模式

8.3.6 常量接口模式

8.4 小结

8.5 思考题

第9章 异常处理

9.1 Java异常处理机制概述

9.1.1 Java异常处理机制的优点

9.1.2 Java虚拟机的方法调用栈

9.1.3 异常处理对性能的影响

9.2 运用Java异常处理机制

9.2.1 try...catch语句：捕获异常

9.2.2 finally语句：任何情况下必须执行的代码

9.2.3 throws子句：声明可能会出现的异常

9.2.4 throw语句：抛出异常

9.2.5 异常处理语句的语法规则

9.2.6 异常流程的运行过程

9.3 Java异常类

9.3.1 运行时异常

9.3.2 受检查异常

9.3.3 区分运行时异常和受检查异常

9.4 用户定义异常

9.4.1 异常转译和异常链

9.4.2 处理多样化异常

9.5 异常处理原则

9.5.1 异常只能用于非正常情况

9.5.2 为异常提供说明文档

9.5.3 尽可能地避免异常

9.5.4 保持异常的原子性

9.5.5 避免过于庞大的try代码块

9.5.6 在catch子句中指定具体的异常类型

9.5.7 不要在catch代码块中忽略被捕获的异常

9.6 小结

9.7 思考题

第10章 类的生命周期

10.1 Java虚拟机及程序的生命周期

10.2 类的加载、连接和初始化

10.2.1 类的加载

10.2.2 类的验证

10.2.3 类的准备

10.2.4 类的解析

10.2.5 类的初始化

10.2.6 类的初始化的时机

10.3 类加载器

10.3.1 类加载的父亲委托 (Parent Delegation) 机制

10.3.2 创建用户自定义的类加载器

10.3.3 URLClassLoader类

10.4 类的卸载

10.5 小结

10.6 思考题

第11章 对象的生命周期

11.1 创建对象的方式

11.2 构造方法

11.2.1 重载构造方法

11.2.2 默认构造方法

11.2.3 子类调用父类的构造方法

11.2.4 构造方法的作用域

11.2.5 构造方法的访问级别

11.3 静态工厂方法

11.3.1 单例 (Singleton) 类

11.3.2 枚举类

11.3.3 不可变 (immutable) 类与可变类

11.3.4 具有实例缓存的不可变类

11.3.5 松耦合的系统接口

11.4 垃圾回收

11.4.1 对象的可触及性

11.4.2 垃圾回收的时间

11.4.3 对象的finalize()方法简介

11.4.4 对象的finalize()方法的特点

11.4.5 比较finalize()方法和finally代码块..

11.5 清除过期的对象引用

11.6 对象的强、软、弱和虚引用

11.7 小结

11.8 思考题

第12章 内部类

12.1 内部类的基本语法

12.1.1 实例内部类

12.1.2 静态内部类

12.1.3 局部内部类

12.2 内部类的继承

12.3 子类与父类中的内部类同名

12.4 匿名类

12.5 内部接口及接口中的内部类

12.6 内部类的用途

12.6.1 封装类型

12.6.2 直接访问外部类的成员

12.6.3 回调 (CallBack)

12.7 内部类的类文件

12.8 小结

12.9 思考题

第13章 多线程

13.1 Java线程的运行机制

13.2 线程的创建和启动

13.2 线程的创建和启动

13.2.1 扩展java.lang.Thread类

13.2.2 实现Runnable接口

13.3 线程的状态转换

13.3.1 新建状态 (New)

- 13.3.2 就绪状态 (Runnable)
- 13.3.3 运行状态 (Running)
- 13.3.4 阻塞状态 (Blocked)
- 13.3.5 死亡状态 (Dead)
- 13.4 线程调度
 - 13.4.1 调整各个线程的优先级
 - 13.4.2 线程睡眠: Thread.sleep()方法
 - 13.4.3 线程让步: Thread.yield()方法
 - 13.4.4 等待其他线程结束: join()
- 13.5 获得当前线程对象的引用
- 13.6 后台线程
- 13.7 定时器Timer
- 13.8 线程的同步
 - 13.8.1 同步代码块
 - 13.8.2 线程同步的特征
 - 13.8.3 同步与并发
 - 13.8.4 线程安全的类
 - 13.8.5 释放对象的锁
 - 13.8.6 死锁
- 13.9 线程通信
- 13.10 中断阻塞
- 13.11 线程控制
 - 13.11.1 被废弃的suspend()和resume()方法
 - 13.11.2 被废弃的stop()方法
 - 13.11.3 以编程的方式控制线程
- 13.12 线程组
- 13.13 处理线程未捕获的异常
- 13.14 ThreadLocal类
- 13.15 小结
- 13.16 思考题
- 第14章 数组
 - 14.1 数组变量的声明
 - 14.2 创建数组对象
 - 14.3 访问数组的元素和长度
 - 14.4 数组的初始化
 - 14.5 多维数组
 - 14.6 调用数组对象的方法
 - 14.7 把数组作为方法参数或返回值
 - 14.8 数组排序
 - 14.9 数组的二叉查找算法
 - 14.10 哈希表
 - 14.11 数组实用类: Arrays
 - 14.12 小结
 - 14.13 思考题
- 第15章 Java集合
 - 15.1 Collection和Iterator接口
 - 15.2 Set (集)
 - 15.2.1 Set的一般用法
 - 15.2.2 HashSet类
 - 15.2.3 TreeSet类
 - 15.3 List (列表)
 - 15.3.1 访问列表的元素
 - 15.3.2 为列表排序
 - 15.3.3 ListIterator接口

- 15.3.4 获得固定长度的List对象
- 15.3.5 比较Java数组和各种List的性能
- 15.4 Map（映射）
- 15.5 HashSet和HashMap的负载因子
- 15.6 集合的编译时类型检查
- 15.7 用for循环遍历集合
- 15.8 集合实用类：Collections
- 15.9 历史集合类
- 15.10 小结
- 15.11 思考题
- 第16章 Java I/O系统
- 16.1 输入流和输出流概述
- 16.2 输入流
 - 16.2.1 字节数组输入流：ByteArrayInputStream类
 - 16.2.2 文件输入流：FileInputStream类
 - 16.2.3 字符串输入流：StringBufferInputStream
 - 16.2.4 管道输入流：PipedInputStream
 - 16.2.5 顺序输入流：SequenceInputStream类
- 16.3 过滤输入流：FilterInputStream
 - 16.3.1 装饰器设计模式
 - 16.3.2 过滤输入流的种类
 - 16.3.3 DataInputStream类
 - 16.3.4 LineNumberInputStream类
 - 16.3.5 BufferedInputStream类
 - 16.3.6 PushbackInputStream类
- 16.4 输出流
 - 16.4.1 字节数组输出流：ByteArrayOutputStream类
 - 16.4.2 文件输出流：FileOutputStream
- 16.5 过滤输出流：FilterOutputStream
 - 16.5.1 DataOutputStream
 - 16.5.2 BufferedOutputStream
 - 16.5.3 PrintStream类
- 16.6 Reader/Writer概述
- 16.7 Reader类
 - 16.7.1 字符数组输入流：CharArrayReader类
 - 16.7.2 字符串输入流：StringReader类
 - 16.7.3 InputStreamReader类
 - 16.7.4 FileReader类
 - 16.7.5 BufferedReader类
- 16.8 Writer类
 - 16.8.1 字符数组输出流：CharArrayWriter类
 - 16.8.2 OutputStreamWriter类
 - 16.8.3 FileWriter类
 - 16.8.4 BufferedWriter类
 - 16.8.5 PrintWriter类
- 16.9 标准I/O
 - 16.9.1 重新包装标准输入和输出
 - 16.9.2 标准I/O重定向
- 16.10 随机访问文件类：RandomAccessFile
- 16.11 新I/O类库
 - 16.11.1 缓冲器Buffer概述
 - 16.11.2 通道Channel概述
 - 16.11.3 字符编码Charset类概述
 - 16.11.4 用FileChannel读写文件

- 16.11.5 控制缓冲区
- 16.11.6 字符编码转换
- 16.11.7 缓冲区视图
- 16.11.8 文件映射缓冲区：MappedByteBuffer
- 16.11.9 文件加锁
- 16.12 对象的序列化与反序列化
- 16.13 File类
 - 16.13.1 查看、创建和删除文件或目录
 - 16.13.2 文件过滤器
- 16.14 小结
- 16.15 思考题
- 第17章 图形用户界面
 - 17.1 AWT容器
 - 17.2 布局管理器
 - 17.2.1 FlowLayout流式布局管理器
 - 17.2.2 BorderLayout边界布局管理器
 - 17.2.3 GridLayout网格布局管理器
 - 17.2.4 CardLayout卡片布局管理器
 - 17.2.5 GridBagLayout网格包布局管理器
 - 17.3 事件处理
 - 17.3.1 事件处理的软件实现
 - 17.3.2 事件源、事件和监听器的类层次和关系
 - 17.4 AWT绘图
 - 17.5 Graphics类
 - 17.6 小结
 - 17.7 思考题
- 第18章 Swing组件
 - 18.1 Swing组件的基本用法
 - 18.2 边框
 - 18.3 按钮组件
 - 18.4 文本框
 - 18.5 文本区域与滚动面板
 - 18.6 复选框与单选按钮
 - 18.7 下拉列表
 - 18.8 列表框
 - 18.9 页签面板
 - 18.10 菜单
 - 18.11 对话框
 - 18.12 文件对话框
 - 18.13 消息框
 - 18.14 绘图
 - 18.15 JApplet
 - 18.16 在JApplet中播放声音
 - 18.17 BoxLayout布局管理器
 - 18.18 设置Swing界面的外观和感觉
 - 18.19 小结
 - 18.20 思考题
- 第19章 Java常用类
 - 19.1 Object 类
 - 19.2 String类和StringBuffer类
 - 19.2.1 String类
 - 19.2.2 “hello” 与new String("hello")的区别
 - 19.2.3 StringBuffer类
 - 19.2.4 比较String类与StringBuffer类

19.2.5 正则表达式
19.3 包装类
19.3.1 包装类的构造方法
19.3.2 包装类的常用方法
19.3.3 包装类的特点
19.4 Math类
19.5 Random类
19.6 处理日期的类
19.6.1 Date类
19.6.2 DateFormat类
19.6.3 Calendar类
19.7 BigDecimal类
19.8 小结
19.9 思考题
参考文献...
· · · · · (收起)

[Java面向对象编程 下载链接1](#)

标签

JAVA

编程

孙卫琴

计算机

程序设计

Java

计算机类

java

评论

个人觉得比那套经典的Java核心技术（卷1，2）要好，更实用。

初学者读的，算是国内写得不错的啦。

JAVA进阶版

当年初学的时候是拿着这本书上的例子对着敲的。纯快速入门书。概念什么的当然不多。

对于有一些基础的人来说看着比较顺眼

Java入门书籍第二本，第一本是Java编程思想。

不知道以后能否重拾这本书。

喜欢dsfsdsafdsfasd

很适合做教材，全面又细节。联系现实的举例不错，对于理解OO里的抽象概念很有帮助。

琴阿姨写书还是挺细心的

例子写的很详细，初学者看挺好的

这本书是为了老婆学Java买的，但是买回来发现，最后是我把这本书来来回回读了几遍，虽然用Java都十个年头了，但是书中提到的很多东西还是值得学习的，收益匪浅呀。

还不错，兼顾广度和深度。

两年之后再次回归java，拿它复习了一下基本内容。内容很详细，还包括了设计模式的概念，还是非常适合初学者

和thinking in java结合着看的，也不错。但是怎么跟我的版不一样？可恶的新版

我当年开始学习Java的第一本书，总之我很喜欢，记得当年毕业时，借给一个同学，只是两年过去了，到现在他也没有还我。（其实还挺想这本书的）

从孙姐姐自己的角度，讲述了java。结合了设计模式等思想。

国内的技术书，虽然内容看着很全，但是全都是罗列要点，记流水账一样的。

国产最好的java书，讲解足够深入详细。我的java入门书籍。

是一本比较好学习编程的书本，讲解得很细致

书评

内容显得过于繁杂，感觉有拼书的痕迹，书的字很小，不便于读者阅读。讲到第12章才开始讲重要的接口概念，总之，感觉不是很好！

一入Java深似海，从android入门到后台中间件，，但相对来讲比起脚本语言，java还是相当友好的。属于江湖少林派系。c,c++,java,java除了有时啰嗦点，...

买了一本给老弟，他要我推荐学java的书，我考虑到他才大三，就选了这本，难度适中。因为我曾经在几年前看过孙卫琴老师的本书第一版，觉得讲解非常细致，行文的逻辑性和描述方式都很符合我的习惯，不像其他教材式的书，那个暑假在学校图书馆就一直看，算是启蒙之书吧，最近我在...

我是卫琴姐姐的粉丝，不仅因为她写的书好，而且因为我也还是女程序猿，卫琴姐姐给我做了榜样。记得刚开始做软件开发工作时，调试程序时，弄了一上午都没搞出来，气急败坏地一把鼻涕一把泪地跑到卫生间哭了一通，开始怀疑自己是否适合做程序猿。后来看到身边一位同事在看这本书，...

内容全面深刻。对程序猿很有启发性的一本书。
书的品质是一流的，价格也算合理的。读了之后受益匪浅。
我已经学Java两年了，再看这本书，更加有画龙点睛的效果，原先的一些技术困惑都被这本书解答了。原来会编一些java程序就沾沾自喜了，对java程序的运行时环境，Java虚拟机的...

当了三年的程序猿，Java、C++、Python，Mybatis, Spring，各种语言都学了一点，用了一下，常用的框架软件也用过，也略懂一二。去年开始蠢蠢欲动，准备跳槽，面试了几家公司，被好几道常识性的Java面试题给难倒了，记得有个面试官问我单例模式有什么好，我茫然不知，才发现自己...

Java开发人员必备之书。布噜布噜。
 到底有多好，还是自己去读比...
 ?

这本书是非常好的一本书，主要特点：概念清晰、逻辑详尽、讲解简洁。
非常适合Java的入门，同时又非常适合作为Java语言的参考书放在手边备查。
书中详细描述了3方面的内容：1、面向对象分析和设计；2、Java语言本身；3、部分的Java类库（主要是IO、集合和多线程)和JVM的一些内...

是看java编程思想入的门，不过当时有很多东西似懂非懂，看完这本书有种豁然开朗的感觉，一些不是太清楚的在这本书讲的很详细，最重要的是，文笔很流畅，读起来很舒服

有这么几本书,因为转行,所以要转让掉,都很新,只限福州,需要的请预约,所有书都是五折，但是同类的书要一起买（1类2类3类）<书名/新旧/作者/原来的价格>: 1类：
<>九成/徐明浩/49 <>九成/孙卫琴/65.8 <...

这本书对java语言，java虚拟机讲很透彻，清楚，如果是编程的新手，可能难度稍大，如果学过C++，C#等，这本书是不可多得的java入门书，孙卫琴一贯流畅的文笔，一定给你带来不一样的感受。

想要教好一门叫做“java”的课，这时才知道用java语言编了两年的程序，其实对它了解的并不多。手边也有一些关于java语言的书，但却不愿拾起重读，也许是小小的虚荣心在作怪吧。在网上偶遇了这本书的节选（第六章），觉得讲得即有条理又不晦涩，蛮好的！急急买了一本回来（要...

我曾经看过一本国内作者写的《面向对象Java编程》。打开目录扫了一下，整本书都是讲Java语法，只在其中一章提到了OO，页数占全书比例不到10%（我强烈怀疑作者是否直接拿了SUN的Java语言规范汉化了一下，就拿去出版了）。

对于JAVA初学者来说,我极力推荐这本书,文字通俗易懂,而且会通过很多实际例子帮助理解,讲得有条理!

这是我看到的非常经典的一本介绍Java编程的书，把Java思想和Java语言本身的使用完美地结合在一起。比如在介绍庞大的IO库中，介绍了IO API的设计模式，这样可以帮助读者更快地理清复杂的IO API的框架体系。再比如介绍面向对象的一些概念时，把生活中的洗衣机、手机等都信手捏来...

我老爸是书虫，时不时地买书回家，许多书到现在还是崭新的，就翻了几下。我老妈多次提意见，说他浪费钱，可他还是忍不住继续买书回家。我本来是打算把老爸作为反面教材，引以为鉴的。结果，居然自己也像饥不择食的书虫一样，常常胡乱买书。Java编程方面的书我已经陆续买了7本了...

孙老师的书通俗易懂，充分发挥了女性作家耐心细心的优势。归纳其中的一个特色：用词非常精确，所有的术语都会给出精确的定义，并且前后保持一致，不会随意地引用各种天花乱坠的术语。由于技术都是首先用英语流传起来，翻译成中文，一个术语就会有若干版本。如果作者在同一本书...

[Java面向对象编程 下载链接1](#)