

# 《Java核心技术·卷1：基础知识》

## 图书基本信息

书名：《Java核心技术·卷1：基础知识（原书第9版）》

13位ISBN编号：9787111445142

10位ISBN编号：7111445147

出版时间：2013-11-1

出版社：机械工业出版社

作者：（美）Cay S. Horstmann,（美）Gary Cornell

页数：720

译者：周立新,陈波,叶乃文,邝劲筠,杜永萍

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu000.com](http://www.tushu000.com)

# 《Java核心技术·卷1：基础知识》

## 内容概要

Java领域最有影响力和价值的著作之一，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9版根据JavaSE7全面更新，同时修正了第8版中的不足，系统全面讲解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。

《Java核心技术·卷1：基础知识》共14章。第1章概述了Java语言与其他程序设计语言不同的性能；第2章讲解了如何下载和安装JDK及本书的程序示例；第3章介绍了变量、循环和简单的函数；第4章讲解了类和封装；第5章介绍了继承；第6章解释了接口和内部类；第7章概述了图形用户界面程序设计知识；第8章讨论AWT的事件模型；第9章探讨了SwingGUI工具箱；第10章讲解如何部署自己的应用程序或applet；第11章讨论异常处理；第12章概要介绍泛型程序设计；第13章讲解Java平台的集合框架；第14章介绍了多线程。本书最后还有一个附录，其中列出了Java语言的保留字。

## 作者简介

Cay S. Horstmann，圣何塞州立大学计算机科学系教授、Java语言的倡导者，也是《Scala for the Impatient》一书（Addison-Wesley，2012）的作者和《Core JavaServer·Faces，3rd》一书（Prentice Hall，2010）的合著者。他还经常在计算机会议上发表演讲。

Cray Cornell，已经教授程序设计专业课程20多年，并撰写了多部专著。他是Apress的创始人之一，他写的程序设计专业书籍非常畅销，曾荣获Jolt震撼大奖，并获得Visual Basic Magazine的读者最喜爱作品大奖。

## 书籍目录

译者序

前言

致谢

第1章 Java程序设计概述

1.1 Java程序设计平台

1.2 Java “白皮书”的关键术语

1.2.1 简单性

1.2.2 面向对象

1.2.3 网络技能

1.2.4 健壮性

1.2.5 安全性

1.2.6 体系结构中立

1.2.7 可移植性

1.2.8 解释型

1.2.9 高性能

1.2.10 多线程

1.2.11 动态性

1.3 Java applet与Internet

1.4 Java发展简史

1.5 关于Java的常见误解

第2章 Java程序设计环境

2.1 安装Java开发工具箱

2.1.1 下载JDK

2.1.2 设置执行路径

2.1.3 安装库源文件和文档

2.1.4 安装本书中的示例

2.1.5 导航Java目录

2.2 选择开发环境

2.3 使用命令行工具

2.4 使用集成开发环境

2.5 运行图形化应用程序

2.6 建立并运行applet

第3章 Java的基本程序设计结构

3.1 一个简单的Java应用程序

3.2 注释

3.3 数据类型

3.3.1 整型

3.3.2 浮点类型

3.3.3 char类型

3.3.4 boolean类型

3.4 变量

3.4.1 变量初始化

3.4.2 常量

3.5 运算符

3.5.1 自增运算符与自减运算符

3.5.2 关系运算符与boolean运算符

3.5.3 位运算符

- 3.5.4 数学函数与常量
- 3.5.5 数值类型之间的转换
- 3.5.6 强制类型转换
- 3.5.7 括号与运算符级别
- 3.5.8 枚举类型
- 3.6 字符串
  - 3.6.1 子串
  - 3.6.2 拼接
  - 3.6.3 不可变字符串
  - 3.6.4 检测字符串是否相等
  - 3.6.5 空串与Null串
  - 3.6.6 代码点与代码单元
  - 3.6.7 字符串API
  - 3.6.8 阅读联机API文档
  - 3.6.9 构建字符串
- 3.7 输入输出
  - 3.7.1 读取输入
  - 3.7.2 格式化输出
  - 3.7.3 文件输入与输出
- 3.8 控制流程
  - 3.8.1 块作用域
  - 3.8.2 条件语句
  - 3.8.3 循环
  - 3.8.4 确定循环
  - 3.8.5 多重选择：switch语句
  - 3.8.6 中断控制流程语句
- 3.9 大数值
- 3.10 数组
  - 3.10.1 for each循环
  - 3.10.2 数组初始化以及匿名数组
  - 3.10.3 数组拷贝
  - 3.10.4 命令行参数
  - 3.10.5 数组排序
  - 3.10.6 多维数组
  - 3.10.7 不规则数组
- 第4章 对象与类
  - 4.1 面向对象程序设计概述
    - 4.1.1 类
    - 4.1.2 对象
    - 4.1.3 识别类
    - 4.1.4 类之间的关系
  - 4.2 使用预定义类
    - 4.2.1 对象与对象变量
    - 4.2.2 Java类库中的Gregorian-Calendar类
    - 4.2.3 更改器方法与访问器方法
  - 4.3 用户自定义类
    - 4.3.1 Employee类
    - 4.3.2 多个源文件的使用
    - 4.3.3 剖析Employee类

- 4.3.4 从构造器开始
- 4.3.5 隐式参数与显式参数
- 4.3.6 封装的优点
- 4.3.7 基于类的访问权限
- 4.3.8 私有方法
- 4.3.9 final实例域
- 4.4 静态域与静态方法
  - 4.4.1 静态域
  - 4.4.2 静态常量
  - 4.4.3 静态方法
  - 4.4.4 工厂方法
  - 4.4.5 main方法
- 4.5 方法参数
- 4.6 对象构造
  - 4.6.1 重载
  - 4.6.2 默认域初始化
  - 4.6.3 无参数的构造器
  - 4.6.4 显式域初始化
  - 4.6.5 参数名
  - 4.6.6 调用另一个构造器
  - 4.6.7 初始化块
  - 4.6.8 对象析构与finalize方法
- 4.7 包
  - 4.7.1 类的导入
  - 4.7.2 静态导入
  - 4.7.3 将类放入包中
  - 4.7.4 包作用域
- 4.8 类路径
- 4.9 文档注释
  - 4.9.1 注释的插入
  - 4.9.2 类注释
  - 4.9.3 方法注释
  - 4.9.4 域注释
  - 4.9.5 通用注释
  - 4.9.6 包与概述注释
  - 4.9.7 注释的抽取
- 4.10 类设计技巧
- 第5章 继承
  - 5.1 类、超类和子类
    - 5.1.1 继承层次
    - 5.1.2 多态
    - 5.1.3 动态绑定
    - 5.1.4 阻止继承：final类和方法
    - 5.1.5 强制类型转换
    - 5.1.6 抽象类
    - 5.1.7 受保护访问
  - 5.2 Object：所有类的超类
    - 5.2.1 equals方法
    - 5.2.2 相等测试与继承

5.2.3 hashCode方法

5.2.4 toString方法

5.3 泛型数组列表

5.3.1 访问数组列表元素

5.3.2 类型化与原始数组列表的兼容性

5.4 对象包装器与自动装箱

5.5 参数数量可变的方法

5.6 枚举类

5.7 反射

5.7.1 Class类

5.7.2 捕获异常

5.7.3 利用反射分析类的能力

5.7.4 在运行时使用反射分析对象

5.7.5 使用反射编写泛型数组代码

5.7.6 调用任意方法

5.8 继承设计的技巧

第6章 接口与内部类

6.1 接口

6.1.1 接口的特性

6.1.2 接口与抽象类

6.2 对象克隆

6.3 接口与回调

6.4 内部类

6.4.1 使用内部类访问对象状态

6.4.2 内部类的特殊语法规则

6.4.3 内部类是否有用、必要和安全

6.4.4 局部内部类

6.4.5 由外部方法访问final变量

6.4.6 匿名内部类

6.4.7 静态内部类

6.5 代理

第7章 图形程序设计

7.1 Swing概述

7.2 创建框架

7.3 框架定位

7.3.1 框架属性

7.3.2 确定合适的框架大小

7.4 在组件中显示信息

7.5 处理2D图形

7.6 使用颜色

7.7 文本使用特殊字体

7.8 显示图像

第8章 事件处理

8.1 事件处理基础

8.1.1 实例：处理按钮点击事件

8.1.2 建议使用内部类

8.1.3 创建包含一个方法调用的监听器

8.1.4 实例：改变观感

8.1.5 适配器类

8.2 动作

8.3 鼠标事件

8.4 AWT事件继承层次

第9章 Swing用户界面组件

9.1 Swing和模型-视图-控制器设计模式

9.1.1 设计模式

9.1.2 模型-视图-控制器模式

9.1.3 Swing按钮的模型-视图-控制器分析

9.2 布局管理概述

9.2.1 边框布局

9.2.2 网格布局

9.3 文本输入

9.3.1 文本域

9.3.2 标签和标签组件

9.3.3 密码域

9.3.4 文本区

9.3.5 滚动窗格

9.4 选择组件

9.4.1 复选框

9.4.2 单选按钮

9.4.3 边框

9.4.4 组合框

9.4.5 滑动条

9.5 菜单

9.5.1 菜单创建

9.5.2 菜单项中的图标

9.5.3 复选框和单选按钮菜单项

9.5.4 弹出菜单

9.5.5 快捷键和加速器

9.5.6 启用和禁用菜单项

9.5.7 工具栏

9.5.8 工具提示

9.6 复杂的布局管理

9.6.1 网格组布局

9.6.2 组布局

9.6.3 不使用布局管理器

9.6.4 定制布局管理器

9.6.5 遍历顺序

9.7 对话框

9.7.1 选项对话框

9.7.2 创建对话框

9.7.3 数据交换

9.7.4 文件对话框

9.7.5 颜色选择器

第10章 部署应用程序和applet

10.1 JAR文件

10.1.1 清单文件

10.1.2 可运行JAR文件

10.1.3 资源



- 10.1.4 密封
- 10.2 Java Web Start
  - 10.2.1 沙箱
  - 10.2.2 签名代码
  - 10.2.3 JNLP API
- 10.3 applet
  - 10.3.1 一个简单的applet
  - 10.3.2 applet的HTML标记和属性
  - 10.3.3 object标记
  - 10.3.4 使用参数向applet传递信息
  - 10.3.5 访问图像和音频文件
  - 10.3.6 applet上下文
- 10.4 应用程序首选项存储
  - 10.4.1 属性映射
  - 10.4.2 Preferences API
- 第11章 异常、断言、日志和调试
  - 11.1 处理错误
    - 11.1.1 异常分类
    - 11.1.2 声明已检查异常
    - 11.1.3 如何抛出异常
    - 11.1.4 创建异常类
  - 11.2 捕获异常
    - 11.2.1 捕获多个异常
    - 11.2.2 再次抛出异常与异常链
    - 11.2.3 finally子句
    - 11.2.4 带资源的try语句
    - 11.2.5 分析堆栈跟踪元素
  - 11.3 使用异常机制的技巧
  - 11.4 使用断言
    - 11.4.1 启用和禁用断言
    - 11.4.2 使用断言完成参数检查
    - 11.4.3 为文档假设使用断言
  - 11.5 记录日志
    - 11.5.1 基本日志
    - 11.5.2 高级日志
    - 11.5.3 修改日志管理器配置
    - 11.5.4 本地化
    - 11.5.5 处理器
    - 11.5.6 过滤器
    - 11.5.7 格式化器
    - 11.5.8 日志记录说明
  - 11.6 调试技巧
  - 11.7 GUI程序排错技巧
  - 11.8 使用调试器
- 第12章 泛型程序设计
  - 12.1 为什么要使用泛型程序设计
  - 12.2 定义简单泛型类
  - 12.3 泛型方法
  - 12.4 类型变量的限定

- 12.5 泛型代码和虚拟机
  - 12.5.1 翻译泛型表达式
  - 12.5.2 翻译泛型方法
  - 12.5.3 调用遗留代码
- 12.6 约束与局限性
  - 12.6.1 不能用基本类型实例化类型参数
  - 12.6.2 运行时类型查询只适用于原始类型
  - 12.6.3 不能创建参数化类型的数组
  - 12.6.4 Varargs警告
  - 12.6.5 不能实例化类型变量
  - 12.6.6 泛型类的静态上下文中类型变量无效
  - 12.6.7 不能抛出或捕获泛型类的实例
  - 12.6.8 注意擦除后的冲突
- 12.7 泛型类型的继承规则
- 12.8 通配符类型
  - 12.8.1 通配符的超类型限定
  - 12.8.2 无限定通配符
  - 12.8.3 通配符捕获
- 12.9 反射和泛型
  - 12.9.1 使用Class参数进行类型匹配
  - 12.9.2 虚拟机中的泛型类型信息
- 第13章 集合
  - 13.1 集合接口
    - 13.1.1 将集合的接口与实现分离
    - 13.1.2 Java类库中的集合接口和迭代器接口
  - 13.2 具体的集合
    - 13.2.1 链表
    - 13.2.2 数组列表
    - 13.2.3 散列集
    - 13.2.4 树集
    - 13.2.5 对象的比较
    - 13.2.6 队列与双端队列
    - 13.2.7 优先级队列
    - 13.2.8 映射表
    - 13.2.9 专用集与映射表类
  - 13.3 集合框架
    - 13.3.1 视图与包装器
    - 13.3.2 批操作
    - 13.3.3 集合与数组之间的转换
  - 13.4 算法
    - 13.4.1 排序与混排
    - 13.4.2 二分查找
    - 13.4.3 简单算法
    - 13.4.4 编写自己的算法
  - 13.5 遗留的集合
    - 13.5.1 Hashtable类
    - 13.5.2 枚举
    - 13.5.3 属性映射表
    - 13.5.4 栈

## 13.5.5 位集

## 第14章 多线程

### 14.1 什么是线程

### 14.2 中断线程

### 14.3 线程状态

#### 14.3.1 新创建线程

#### 14.3.2 可运行线程

#### 14.3.3 被阻塞线程和等待线程

#### 14.3.4 被终止的线程

### 14.4 线程属性

#### 14.4.1 线程优先级

#### 14.4.2 守护线程

#### 14.4.3 未捕获异常处理器

### 14.5 同步

#### 14.5.1 竞争条件的一个例子

#### 14.5.2 竞争条件详解

#### 14.5.3 锁对象

#### 14.5.4 条件对象

#### 14.5.5 synchronized关键字

#### 14.5.6 同步阻塞

#### 14.5.7 监视器概念

#### 14.5.8 Volatile域

#### 14.5.9 final变量

#### 14.5.10 原子性

#### 14.5.11 死锁

#### 14.5.12 线程局部变量

#### 14.5.13 锁测试与超时

#### 14.5.14 读/写锁

#### 14.5.15 为什么弃用stop和suspend方法

### 14.6 阻塞队列

### 14.7 线程安全的集合

#### 14.7.1 高效的映射表、集合和队列

#### 14.7.2 写数组的拷贝

#### 14.7.3 较早的线程安全集合

### 14.8 Callable与Future

### 14.9 执行器

#### 14.9.1 线程池

#### 14.9.2 预定执行

#### 14.9.3 控制任务组

#### 14.9.4 Fork-Join框架

### 14.10 同步器

#### 14.10.1 信号量

#### 14.10.2 倒计时门栓

#### 14.10.3 障栅

#### 14.10.4 交换器

#### 14.10.5 同步队列

### 14.11 线程与Swing

#### 14.11.1 运行耗时的任务

#### 14.11.2 使用Swing工作线程

14.11.3 单一线程规则  
附录 Java关键字

## 精彩短评

- 1、比较全面的书
- 2、不评价
- 3、不错，清晰，但不够深入，入门很好。
- 4、 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，这是很有意义的，因为java与c++其实有很多类似之处，尤其是在面向对象的方面。同时书中也对Java与C++不同的地方做了一些解释，这对于学习过C++的朋友来说是很有助益的，更能理解两门语言的不同。  
我以为是写书评的，才写了这么多。。。。。。。
- 5、Java入门必备经典！
- 6、作者用更易理解的方式组织编写了这本Java API介绍的书，可以看到很多Java应用的细节，但Java思想涉及较少。
- 7、易懂而深刻，个别地方翻译的不好理解
- 8、覆盖面广，但不够深入
- 9、虽说全书语言不够简练，示例代码不够严谨，但是java是在是没有一本令人满意的入门书籍，矮子里面选将军，这本已经不错了。
- 10、太碎了，结构上不错。入门经典。时间要多花点，而且翻译会不通，要结合上下文理解。
- 11、还可以吧，适合初学者
- 12、oop编程的基础讲解，面向对象编程的通用模式就是设计模式
- 13、还行
- 14、并没觉得有多好，非gui例子也引用gui的类，云里雾里
- 15、有帮助
- 16、告诫自己，基础最重要，基础最重要。。。这本书对新手非常友好.....不想JAVA编程思想那么多抽象和概念，上来就介绍api，如何使用，非常的简单和直接，建议读完此书再看JAVA编程思想，收获会很多
- 17、看过c primer plus的同学路过 感觉这本书前面几章有点让人摸不着头脑，一本书应当是循序渐进，而作者一下子堆一些让人很难理解的东西 不说内容如何，编排结构就不好。
- 18、优点:全面！全面！还是全面。除此之外都一般！尤其是反人类的字体排版，看着难受！！！！！！！！！！把可视化部分删除或者压缩一下比较好，毕竟这门语言的精髓不在这里。前几章节放一大堆API文档也就算了，后面也是，完全是冒充页数！尤其是卷一，感觉全书就是内容的堆砌，缺乏条理。
- 19、Chap.7-9 GUI编程，可以跳过不看
- 20、用于复习java很不错
- 21、为了毕设
- 22、部分浅显易懂不深入，部分晦涩难懂没讲透
- 23、好书
- 24、写的挺全的，带有些地方一笔带过太省略，相对有些基础获悉更适合这本。
- 25、扎实基础的利器
- 26、后面多线程已晕，跟Java编程思想比起来这本真是入门级啊
- 27、语言很精练，很深刻，没有Thinking in Java那么啰嗦
- 28、初级进阶很棒！
- 29、过誉了，写的一般
- 30、挺好的入门书。但还是觉得入门的话，国内的某些书更适合我们。
- 31、不明白为什么这么多人推荐这本书。可能看原版好一点。翻译真的太垃圾了！很多话让人不明不

白，让人读不下去，我看了25%左右然后就开始速度浏览一遍了事。不推荐这本作为入门的教材，更像是一个词汇书或者工作经验总结。今后有时间会看看原版。

32、翻译感觉还是很别扭

33、前面还可以，多线程部分写得真是狗屎，我想不是我理解能力不行

34、挺不错的

35、看完真的后悔买了那本狗屁的《Java从入门到精通》。

36、好啊

37、卷一看完，收益颇丰，就是有几个章节没看，为何不把卷二的章节替换过来呢？准备看卷二。

38、书籍还是不错的，单要静下来读，有些翻译的专业术语还是要看英文

39、作为一本入门参考书还是很不错的

40、非常棒的入门书

41、JAVA入门强烈推荐！！

42、这书写得全面，可能是因为翻译原因，很多案例讲解得很散，一个案例分得东一片西一片，没有逻辑性

43、还在读，写的很不错，思路模块清楚

44、唯一比较坑爹的是，花了太多篇幅讲已经被淘汰的Swing GUI

45、什么翻译水平，make a lot of sense 都能翻译成“产生多种解释”

46、先来个短评：这本书内部类那块讲的很不好。既然定位基础知识，不应该包裹太多的东西

47、作为入门书籍还是不错的，尤其是学过C++后学

48、不怎么注重编程实践，但语法讲的很好，特别是对于刚学完《c++程序设计原理与实践》的，看到一语中的的c++注释感动哭了。

49、非常不错，这本书写得很新，很详细，而且和C++对比是特别适合以C++得基础入门Java

50、适合入门，java的各个技术点讲的很清楚。

### 精彩书评

1、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，这是很有意义的，因为java与c++其实有很多类似之处，尤其是在面向对象的方面。同时书中也对Java与C++不同的地方做了一些解释，这对于学习过C++的朋友来说是很有助益的，更能理解两门语言的不同。书中的代码实例也很多，同时也会对Java对象常用的方法进行解析，容易让初学者短时间内更容易熟悉Java常见对象的常见方法，对于掌握Java有很大帮助。但是，书中让人扣分的一点是，其对于Java图形化编程的大书特书。这，其实算是Java在实际应用中应用得最少的一部分吧。Java的主要应用都是在服务器端应用。不过，跳过这些章节也不影响阅读体验，这也是极好的。

## 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：[www.tushu000.com](http://www.tushu000.com)