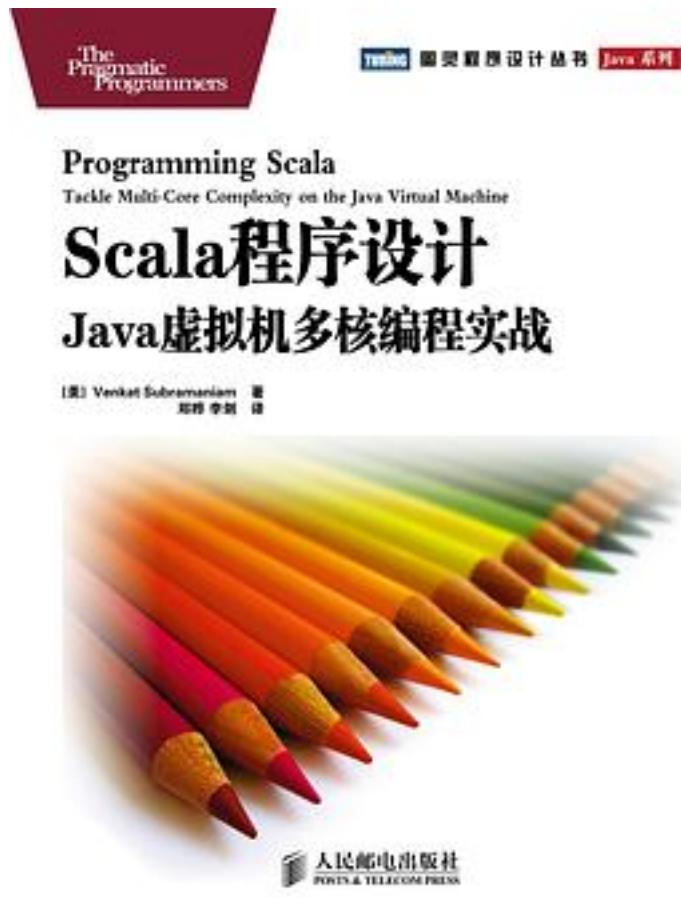


Scala程序设计



[Scala程序设计_下载链接1](#)

著者:[美] Venkat Subramaniam

出版者:人民邮电出版社

出版时间:2010-8

装帧:

isbn:9787115232953

Scala是一种针对 JVM 的通用的编程语言，用来以简明、优雅、类型安全的方式表示常见的编程模式。它无缝集成了面向对象语言和函数式语言的特性，可以大大提高JAVA程序员的编程效率。Twitter、SAP等知名企业已经开始在一些应用里使用Scala，很多新的程序员也开始学习它。

。

本书循序渐进地介绍了Scala的函数式编程基础，虽然篇幅短小，却切中要害。读者可以学会使用Scala静态语言的强大功能创建简洁、可扩展、高度可并行的代码。对于多核时代JVM上的并发编程，Scala是绝好的工具，而本书是你必不可少的向导。

作者介绍：

Venkat Subramaniam, Agile Developer创办人，培训并指导了美国、加拿大、印度和欧洲多国的上千名软件开发人员，并多次应邀在各种大会上发表演讲。他是《高效程序员的45个习惯：敏捷开发修炼之道》（2007年Jolt大奖作品）的第一作者，还著有.NET Gotchas (O'Reilly)、Programming Groovy (Pragmatic) 等书。

目录: 第1章 简介 1

1.1 为何选择Scala 1

1.2 何为Scala 3

1.3 函数式编程 7

1.4 本书的内容 9

1.5 本书面向的读者 11

1.6 致谢 11

第2章 起步 13

2.1 下载Scala 13

2.2 安装Scala 13

2.2.1 在Windows上安装Scala 14

2.2.2 在类UNIX系统上安装Scala 14

2.3 让Scala跑起来 15

2.4 命令行上的Scala 16

2.5 把Scala代码当作脚本运行 17

2.5.1 在类UNIX系统上作为脚本运行 17

2.5.2 在Windows上作为脚本运行 18

2.6 在IDE里面运行Scala 18

2.7 编译Scala 19

第3章 Scala步入正轨 20

3.1 把Scala当作简洁的Java 20

3.2 Java基本类型对应的Scala类 23

3.3 元组与多重赋值 23

3.4 字符串与多行原始字符串 25

3.5 自适应的默认做法 26

3.6 运算符重载 27

3.7 Scala带给Java程序员的惊奇 29

3.7.1 赋值的结果 29

3.7.2 Scala的== 30

3.7.3 分号是半可选的 31

3.7.4 默认的访问修饰符 32

3.7.5 默认的访问修饰符以及如何修改 32

3.7.6 Scala的Protected 33

3.7.7 细粒度访问控制 34

3.7.8 避免显式return 35

第4章 Scala的类 37

4.1 创建类 37

4.2 定义字段、方法和构造函数 38

4.3 类继承 41
4.4 单例对象 42
4.5 独立对象和伴生对象 43
4.6 Scala中的static 44
第5章 自适应类型 46
5.1 容器和类型推演 47
5.2 Any类型 49
5.3 关于Nothing的更多情况 50
5.4 Option类型 50
5.5 方法返回类型推演 51
5.6 传递变参 52
5.7 参数化类型的可变性 53
第6章 函数值和闭包 57
6.1 从普通函数迈向高阶函数 57
6.2 函数值 58
6.3 具有多参数的函数值 59
6.4 Curry化 61
6.5 重用函数值 62
6.6 参数的位置记法 64
6.7 Execute Around Method模式 65
6.8 偏应用函数 67
6.9 闭包 68
第7章 Trait和类型转换 70
7.1 Trait 70
7.2 选择性混入 72
7.3 以trait进行装饰 74
7.4 Trait方法的延迟绑定 75
7.5 隐式类型转换 77
第8章 使用容器 81
8.1 常见的Scala容器 81
8.2 使用Set 82
8.3 使用Map 83
8.4 使用List 85
8.5 for表达式 90
第9章 模式匹配和正则表达式 93
9.1 匹配字面量和常量 93
9.2 匹配通配符 94
9.3 匹配元组和列表 94
9.4 类型和卫述句的匹配 96
9.5 case表达式里的模式变量和常量 96
9.6 对XML片段进行模式匹配 98
9.7 使用case类进行模式匹配 98
9.8 使用提取器进行匹配 100
9.9 正则表达式 103
9.10 把正则表达式当做提取器 104
第10章 并发编程 106
10.1 促进不变性 106
10.2 使用Actor的并发 107
10.3 消息传递 110
10.4 Actor类 113
10.5 actor方法 115
10.6 receive和receiveWithin方法 117
10.7 react和reactWithin方法 120
10.8 loop和loopWhile 124

10.9 控制线程执行	125
10.10 在各种接收方法中选择	127
第11章 与Java互操作	128
11.1 在Scala里使用Scala类	128
11.2 在Scala里使用Java类	130
11.3 在Java里使用Scala类	132
11.3.1 有普通函数和高阶函数的Scala类	132
11.3.2 同trait一起工作	134
11.3.3 单例对象和伴生对象	134
11.4 继承类	136
第12章 用Scala做单元测试	138
12.1 使用JUnit	138
12.2 使用ScalaTest	139
12.3 以Canary测试开始	140
12.4 使用Runner	140
12.5 Asserts	142
12.6 异常测试	144
12.7 在测试间共享代码	146
12.7.1 用BeforeAndAfter共享代码	146
12.7.2 用闭包共享代码	147
12.8 FunSuite的函数式风格	148
12.9 用JUnit运行ScalaTest	149
第13章 异常处理	152
13.1 异常处理	152
13.2 注意catch顺序	154
第14章 使用Scala	156
14.1 净资产应用实例	156
14.2 获得用户输入	156
14.3 读写文件	157
14.4 XML, 作为一等公民	159
14.5 读写XML	161
14.6 从Web获取股票价格	164
14.7 让净资产应用并发	167
14.8 为净资产应用增加GUI	168
附录A Web资源	178
• • • • • (收起)	

[Scala程序设计 下载链接1](#)

标签

scala

Java

函数式编程

多核编程

编程

计算机

编程语言

程序设计

评论

作为一本入门书籍还是不错的，而且翻译也还不错吧。特别是这本书是站在一个java程序员的角度去介绍scala的，当然，对于scala自己的特性，书中介绍的就比较详细了

不学Scala的两个理由：1.Scala朝着错误的方向发展，它要做到大而全，导致语言越来越复杂，学习成本高。2.Scala是java平台的语言，必然用到jdk，设计糟糕的jdk，api设计得很差；实际工作中，涉及到业务系统时，用到的语言和框架，请远离java平台！

并发，tuple，语法糖等好多便利的语法，很是喜欢啊

讲的太简单了，只能用于快速了解Scala。

快速入门。。。写出java式的scala，不着急用的童鞋可以略过这本，直接看programming in scala

书略旧（讲的还是 Scala 2.7），火车上翻完的，细节没认真看，但还是注意到一些亮点：1. trait 2. 函数调用的 .

(dot) 和括号可选 3. 略变态的 pattern matching。评分低的原因：讲编程语言的书不附习题都是偷懒 ==

有点（副）标题党 ...

想看 akka，无奈还是觉得学 scala 生产效率才能提升，一直没怎么实践过，这本书入门不错～

魔化太多

因为 Spark 里用了 Scala 才看的这本书，UCB 的牛就是猛。很多概念，如闭包、actor 以前也接触过，也就没有什么感觉，不太喜欢 Scala（包括 Ruby）里面的语法糖，书很薄，内容太少。

泛泛的讲解 Scala 的一些很基础的知识，作为入门是可以

还行

适合有一点 scala 经验的人复习下… 版本太老是硬伤…

先看了七周七语言的 scala 部分 这本一两周的空闲时间就可以大致读完了
很适合 java 程序员看

副标题和内容严重不符

口袋书，枕边书，如厕书，入门必备

仅仅是学习是不够的，还需要多做项目。另外，关于Scala的书籍好像还是少了点。

这本书原来看过一遍 大概是大四的时候 当时没有理解函数编程的含义 现在略有所悟
此书内容深刻 容量很大 文字流畅 毫不做作 比较了Java与Scala的异同 句句切中要害
难得的好书

以前看到Scala代码的时候觉得语法莫名其妙难以理解，看完这本书开始喜欢上Scala了。
◦

内容凝练也易懂。scala会是我通向jvm的钥匙么？

[Scala程序设计 下载链接1](#)

书评

大师们都推荐，每个人至少学第二门编程语言。在看scala之前，学C#的同学一直诟病Java的语法：一潭死水、不吸取新的特色。当时，我只能用C#问世比Java晚借鉴Java的长处弥补不足为条件力辩--后来人依葫芦画瓢然后避免前人犯的一些错误就Ok，而且，你不能期待微软模拟出来...

这两个关键词：1、适合Java开发者 2、快速入门Scala
概念清楚，条理清晰。很多在Java眼里很晦涩的概念解释的非常合理和透彻。看完这本之后再看更深的书会容易的多。

译者的辛苦努力首先要得到承认。但不得不说这本书的内容有点鸡肋。很多东西都没写清楚，行文不太流畅，有个别错误。作为一名Java程序员如果打算了解Scala，IBM

developer有几篇不错的文章。如果打算全面了解Scala，这本薄薄的书的内容实在太少。总之，买这本书，有点亏得慌！

目标读者定位很明确，就是帮助Java程序员熟悉Scala语言。
可能是由于Scala的确有很多新颖(或者说不太适应)的东西，加上一些概念还没有标准化翻译，所以感觉还是稍欠打磨。
不过这本书还是推荐作为Scala入门书籍，对初接触Scala的人来说，还是很不错。
另外，我觉得译作加上...

Scala中文方面的资料很少，这本虽然老一点不过相当不错，至少比官方文档好理解得多，把很多坑跟Java的异同点都说得很清楚（这一点相当重要）。由于Scala是基于JVM的，本书也提到很多Java的东西所以最好还是有点Java基础，不过话说回来说不定不被Java影响可能学得更好？以下是...

本来是想借助它入门，结果，也仅仅是入了个门，对很多东西描述的太少。
是很轻薄，但同时好多例子也不好理解，解释的太粗。
所以，我的观点：不管是入门，还是想深入研究，强烈推荐另一本《scala编程》，scala的作者写的，不会错的。

很薄而有精炼的一本书，通过ipad在上班的地铁上把这本书E文版看完了。一个Java程序员，看完一遍，基本上可以比较顺利的使用Scala了写程序了。有了Scala，你还会期待慢腾腾的Java
7,8,9,10..的新特性么？打算在一些小的项目、工具和机器学习的实验代码中使用并熟悉scala语言。对...

本书主要讲了学习、使用 Scala 必备的知识点（编程要点、技巧），
略去了太过“偏”的细节，更加容易理解！

快速入门书，没看完呢，不过觉得译者有点刻意想让读者知道自己看懂了原文的感觉，有点别扭，而且经常是通过无意义的脚注来说明他们看懂了，不喜欢这种做法。

[Scala程序设计](#) [下载链接1](#)