



Scala

编程实战

O'REILLY®

机械工业出版社
China Machine Press

Alvin Alexander 著
马博文 张锦文 任晓君 译

Scala编程实战

Alvin Alexander 著

马博文 张锦文 任晓君 译

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

O'Reilly Media, Inc. 授权机械工业出版社出版

机械工业出版社

图书在版编目 (CIP) 数据

Scala编程实战/ (美) 亚历山大 (Alexander, A.) 著; 马博文, 张锦文, 任晓君译.
—北京: 机械工业出版社, 2016.2

(O'Reilly精品图书系列)

书名原文: Scala Cookbook

ISBN 978-7-111-52686-5

I. S… II. ①亚… ②马… ③张… ④任… III. Java语言—程序设计 IV. TP312
中国版本图书馆CIP数据核字 (2016) 第014624号

北京市版权局著作权合同登记

图字: 01-2014-2043号

©2013 Alvin Alexander. All rights reserved.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and China Machine Press, 2016. Authorized translation of the English edition, 2013 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由O'Reilly Media, Inc. 出版2013。

简体中文版由机械工业出版社出版 2016。英文原版的翻译得到O'Reilly Media, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

封底无防伪标均为盗版

本书法律顾问

北京大成律师事务所 韩光/邹晓东

书 名/ Scala编程实战

书 号/ ISBN 978-7-111-52686-5

责任编辑/ 陈佳媛

封面设计/ Karen Montgomery, 张健

出版发行/ 机械工业出版社

地 址/ 北京市西城区百万庄大街22号 (邮政编码 100037)

印 刷/ 北京市荣盛彩色印刷有限公司

开 本/ 178毫米×233毫米 16开本 41.25印张

版 次/ 2016年6月第1版 2016年6月第1次印刷

定 价/ 119.00元 (册)

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010)88379426; 88361066

购书热线: (010)68326294; 88379649; 68995259

投稿热线: (010)88379604

读者信箱: hzit@hzbook.com

O'Reilly Media, Inc.介绍

O'Reilly Media通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自1978年开始，O'Reilly一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了Make杂志，从而成为DIY革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly现在还将先锋专家的知识传递给普通的计算机用户。无论是通过书籍出版，在线服务或者面授课程，每一项O'Reilly的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar博客有口皆碑。”

——Wired

“O'Reilly凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——Business 2.0

“O'Reilly Conference是聚集关键思想领袖的绝对典范。”

——CRN

“一本O'Reilly的书就代表一个有用、有前途、需要学习的主题。”

——Irish Times

“Tim是位特立独行的商人，他不光放眼于最长远、最广阔的视野并且切实地按照Yogi Berra的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去Tim似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——Linux Journal

译者序

刚开始走入工作岗位的时候，当时公司多数的项目都在使用Ruby语言，接触不久便深深地爱上了Ruby的灵活和可读性。2013年时，客户决定拆分一个遗留系统，并且尝试用基于Scala语言的Play框架去搭建微服务。第一次接触的时候，感觉非常震惊，因为这门基于JVM的语言，除了保留了Java静态检查的好处，还兼有OOP和FP的能力，并且写起来让我找到了Ruby的感觉。我们从Scala中受益匪浅，后续的微服务，都是采用基于Scala语言的框架，Unfinagled和Unfilterd，都取得了成功。我有理由相信，Scala将成为整条业务线的技术选择的标准。

从大的范围来看，越来越多的公司也开始采用Scala，如Twitter、LinkedIn、Coursera等。我们耳熟能详的一些大数据处理平台，如Storm和Spark，也都是基于Scala语言去构建的。TypeSafe的官网 (<https://www.typesafe.com/resources/case-studies-and-stories>) 给出了很多用户案例。从这些例子我们可以看出，Scala的未来不可限量，值得我们投入时间去学习。

学习Scala语言，不仅仅意味着熟悉新的API，更重要的是一种思维方式的转变。从原有的面向对象编程（OO）到函数式编程（FP）的思想，这是我认为学习Scala最困难的地方，因为很多程序员还处在面向对象编程的学习阶段。和很多其他编程语言的学习一样，如果一开始有丰富的例子可以参考，可以帮助你解决上手时的问题，不管是对于项目的进度，以及进一步的深入学习，都是大有裨益的。本书就是你学习Scala的绝佳指南，它面向实际的使用场景，提供了大量的实例，同时，也给出底层的原理和相关的参考。对于Scala新手来说这是一本非常不错入门书，而对于老手来说也是一本夯实基础，检视自己所学知识的好书。

感谢作者Alvin Alexander为我们带来这本实用的Scala作品，感谢我的同事张锦文和任晓军对这次翻译的鼎力相助，我们都是第一次翻译英文技术书籍，而且是如此鸿篇巨著，这对我们来说是莫大的挑战。感谢我的家人和女友对我工作的支持和理解，没有他们我无法完成这个任务。

本书翻译过程中，译者已尽力将原著的意思还原到中文的语境，但毕竟能力和经验都尚浅，难免有不足和疏漏的地方。望各位读者批评斧正。

马博文

2015年6月29日

目录

前言	1
第1章 字符串	11
1.1 测试字符串的相等性	14
1.2 创建多个字符串	16
1.3 分隔字符串	17
1.4 字符串中的变量代换	19
1.5 挨个处理字符串中的字符	23
1.6 字符串中的查找模式	27
1.7 字符串中的替换模式	30
1.8 抽取String中模式匹配的部分	31
1.9 访问字符串中的一个字符	33
1.10 在String类中添加自定义的方法	34
第2章 数值	39
2.1 从字符串到数值	40
2.2 数值类型转换	44
2.3 重载默认数值类型	45
2.4 替代++和--	47
2.5 浮点数的比较	49
2.6 处理大数	51
2.7 生成随机数	53
2.8 创建一个数值区间、列表或者数组	55

2.9 格式化数值和金额.....	57
-------------------	----

第3章 控制结构 60

3.1 for和foreach循环	61
3.2 在for循环中使用多个计数器	67
3.3 在for循环中嵌入if语句（卫语句）	69
3.4 创建for表达式（for/yield组合）	70
3.5 实现break和continue.....	72
3.6 像三元运算符一样使用if.....	77
3.7 像switch语句一样使用匹配表达式.....	78
3.8 一条case语句匹配多个条件.....	83
3.9 将匹配表达式的结果赋值给变量	84
3.10 访问匹配表达式缺省case的值	84
3.11 在匹配表达式中使用模式匹配	85
3.12 在匹配表达式中使用Case 类.....	92
3.13 给Case语句添加if表达式（卫语句）	93
3.14 使用匹配表达式替换isInstanceOf.....	94
3.15 在匹配表达式中使用List	95
3.16 用try/catch 匹配一个或者更多的异常	97
3.17 在try/catch/finally块中使用变量前定义变量	98
3.18 创建自定义控制结构.....	100

第4章 类和属性 103

4.1 创建一个主构造函数.....	104
4.2 控制构造函数字段的可见性.....	108
4.3 定义辅助构造函数.....	111
4.4 定义私有的主构造函数	115
4.5 设置构造函数参数的默认值.....	116
4.6 覆写默认的和修改方法.....	118
4.7 阻止生成getter和setter方法	121
4.8 将代码块或者函数赋给字段.....	123
4.9 设置未初始化的var字段类型.....	125

4.10 在继承类时处理构造函数参数	127
4.11 调用父类的构造函数	129
4.12 何时使用抽象类.....	131
4.13 在抽象基类(或者特质) 里定义属性.....	133
4.14 用Case类生成模板代码	137
4.15 定义一个equals方法 (对象的相等性)	141
4.16 创建内部类.....	144
第5章 方法	147
5.1 控制方法作用域	148
5.2 调用父类的方法	152
5.3 方法参数默认值	154
5.4 使用参数名.....	157
5.5 定义一个返回多个值(Tuples)的方法.....	159
5.6 调用getter/setter方法时不使用括号	160
5.7 创建接受变参的方法	162
5.8 方法的异常声明	164
5.9 支持链式调用编码风格	166
解决办法	167
第6章 对象	170
6.1 对象的强制转换	171
6.2 Java.class的Scala等价类.....	173
6.3 确定对象所属的类.....	174
6.4 用Object启动一个应用	175
6.5 用object创建单例.....	177
6.6 用伴生类创建静态成员	178
6.7 将通用代码放入包对象	180
6.8 不使用new关键字创建对象实例.....	183
6.9 在Scala中用apply方法实现工厂方法.....	187
第7章 包和导入	190
7.1 花括号风格的包记号法	191

7.2 引入一个或多个成员	192
7.3 在导入时重命名成员	194
7.4 在引入时隐藏一个类	195
7.5 使用静态导入	196
7.6 在任意地方使用Import语句	197
第8章 特质	200
8.1 特质用作接口	200
8.2 使用特质中的抽象字段和实际字段	203
8.3 像抽象类一样使用特质	204
8.4 简单混入特质	205
8.5 通过继承来限制特质的使用范围	206
8.6 限定特质只可用于指定类型的子类	208
8.7 保证特质只能被添加到只有一个特定方法的类型	210
8.8 为对象实例中添加特质	211
8.9 像特质一样继承一个Java接口	212
第9章 函数式编程	214
9.1 使用函数字面量（匿名函数）	215
9.2 将函数作为变量	216
9.3 定义接受简单函数作为参数的方法	220
9.4 更多复杂的函数	222
9.5 使用闭包	226
9.6 使用部分应用函数	231
9.7 创建返回函数的函数	232
9.8 创建部分应用函数	235
9.9 一个真实的例子	239
第10章 集合	242
10.1 理解集合的层级结构	243
10.2 选择一个集合类	247
10.3 用一个集合方法解决一个问题	252

10.4 理解集合的性能.....	258
10.5 在创建集合时声明一个类型.....	260
10.6 理解可变变量与不可变集合.....	262
10.7 将Vector“变为”不可变序列.....	263
10.8 把ArrayBuffer作为可变序列的第一选择.....	265
10.9 用foreach遍历一个集合.....	267
10.10 用for循环遍历一个集合.....	269
10.11 使用zipWithIndex或者zip创建循环计数器.....	272
10.12 迭代器的使用.....	275
10.13 用for/yield实现集合间的转换.....	276
10.14 用map实现集合的变换.....	279
10.15 展平列表的列表与扁平化问题.....	281
10.16 map, flatten和flatMap的组合.....	283
10.17 用filter过滤一个集合.....	285
10.18 从集合中提取元素序列.....	287
10.19 序列的分割 (groupBy、partition等).....	289
10.20 用reduce和fold方法遍历集合.....	291
10.21 从序列中提取不重复的元素.....	296
10.22 合并序列集合.....	297
10.23 用zip将两个序列集合合并为一对.....	300
10.24 在集合上创建一个惰性视图.....	301
10.25 利用Range创建集合.....	305
10.26 创建使用枚举.....	307
10.27 当需要一堆元素时使用元组.....	308
10.28 集合排序.....	310
10.29 通过mkString将集合转换成字符串.....	313

第11章 列表、数组、映射、集及其他 315

11.1 创建和填充列表的不同方式.....	316
11.2 创建可变List.....	318
11.3 为List添加元素.....	319
11.4 从List (或者ListBuffer) 中删除元素.....	321

11.5 合并或连接列表.....	323
11.6 使用List的懒惰版本, Stream.....	324
11.7 创建和更新数组的不同方式.....	326
11.8 创建大小可变的数组 (ArrayBuffer)	328
11.9 删除Array和ArrayBuffer的元素.....	329
11.10 数组排序.....	331
11.11 创建多维数组.....	332
11.12 创建映射.....	334
11.13 选择一种Map实现.....	336
11.14 为可变映射添加、更新或删除元素.....	338
11.15 为不可变映射添加、更新或删除元素.....	340
11.16 访问映射的值.....	342
11.17 遍历映射.....	344
11.18 从映射中获得所有的键或者值.....	345
11.19 反转键值.....	346
11.20 测试映射中键/值的存在.....	347
11.21 过滤映射.....	348
11.22 根据键或者值对映射排序.....	350
11.23 找到映射中最大的键或者值.....	354
11.24 给集添加元素.....	355
11.25 从集中删除元素.....	357
11.26 使用可排序集.....	359
11.27 使用队列.....	360
11.28 使用栈.....	362
11.29 使用Range.....	365

第12章 文件和进程 367

12.1 如何打开和读取文本文件.....	367
12.2 写入文本文件.....	373
12.3 读写二进制文件.....	374
12.4 如何处理文本文件中的每个字符.....	375
12.5 如何处理CSV文件.....	376

12.6 将字符串伪装为文件	379
12.7 使用序列化	381
12.8 列出目录中的文件	382
12.9 列举目录之下的子目录	384
12.10 执行外部命令	385
12.11 执行外部命令并使用标准输出	388
12.12 处理外部命令的标准输出和标准错误输出	390
12.13 构建命令管道	392
12.14 重定向外部命令的标准输出和标准输入	393
12.15 在进程中使用AND (&&) 和OR ()	395
12.16 外部命令中处理通配符	396
12.17 如何在不同目录下运行进程	397
12.18 在运行命令时设置环境变量	398
12.19 执行外部命令的方法索引	399
第13章 Actors和并发	401
13.1 Actor简单入门	403
13.2 创建构造函数有参数的actor	407
13.3 Actor之间如何通信	409
13.4 理解Akka Actor生命周期方法	411
13.5 启动Actor	414
13.6 停止Actors	416
13.7 关闭Akka Actor系统	421
13.8 用watch监控Actor的死亡	422
13.9 Futures的简单并发	425
13.10 给Actor发消息并等待回复	433
13.11 用become切换不同状态	434
13.12 使用并发集合	436
第14章 命令行任务	440
14.1 Scala REPL初体验	441
14.2 往REPL中粘贴和加载代码块	445

14.3 在REPL的Classpath中添加JAR文件和类	448
14.4 在REPL中运行Shell命令	449
14.5 用scalac命令编译, scala命令运行	451
14.6 反汇编和反编译Scala代码	452
14.7 寻找Scala类库	457
14.8 用scaladoc生成文档	458
14.9 更快的命令行编译器fsc	463
14.10 将Scala作为脚本语言使用	464
14.11 通过脚本访问命令行参数	468
14.12 在Scala shell脚本中输入提示	469
14.13 让Scala脚本运行得更快	473
第15章 Web服务	475
15.1 从Scala对象中创建JSON字符串	475
15.2 从包含集合的类中生成JSON字符串	479
15.3 从JSON字符串生成简单Scala对象	484
15.4 将JSON数据解析成为对象的数组	485
15.5 用Scalatra创建Web服务	487
15.6 用Scalatra挂载替换XML Servlet映射	490
15.7 访问Scalatra Web服务的GET参数	492
15.8 用Scalatra处理POST请求数据	494
15.9 创建一个简单的GET请求客户端	497
15.10 向POST链接发送JSON数据	501
15.11 获取URL Headers	502
15.12 发送请求时设置链接的Headers	503
15.13 用Play框架创建一个GET请求的Web服务	504
15.14 向Play框架的web服务POST JSON数据	507
第16章 数据库和持久化	510
16.1 用JDBC连接MySQL	511
16.2 用Spring Framework连接数据库	513
16.3 连接MongoDB并且插入数据	516

16.4 用insert, save或者+=把文档插入到MongoDB	519
16.5 搜索一个MongoDB集合	521
16.6 更新MongoDB集合中的文档.....	525
16.7 获取MongoDB文档ID.....	526
16.8 删除MongoDB集合中的文档.....	527
16.9 快速浏览Slick.....	529
第17章 与Java交互	531
17.1 与Java集合的转换	531
17.2 给Scala方法添加异常注解以便与Java交互	536
17.3 使用@SerialVersionUID和其他注解.....	538
17.4 使用Spring框架	539
17.5 注解变长参数方法	542
17.6 当Java的代码需要JavaBeans	543
17.7 包装特质的实现.....	547
第18章 简单构建工具 (SBT)	551
18.1 用SBT创建一个项目目录结构.....	551
18.2 用SBT编译、运行和打包一个Scala项目	555
18.3 用SBT和ScalaTest运行测试.....	560
18.4 用SBT管理依赖	562
18.5 控制所使用的可管理依赖的版本	565
18.6 创建有子项目的项目	567
18.7 在Eclipse中使用SBT.....	569
18.8 生成项目API文档	571
18.9 指定一个Main类运行.....	572
18.10 使用GitHub项目作为项目依赖	574
18.11 告诉SBT如何找到一个仓库 (使用Resolvers)	575
18.12 通过SBT堆栈跟踪解决问题.....	576
18.13 设置SBT日志等级	577
18.14 部署一个单独的、可执行的JAR文件.....	578
18.15 发布类库.....	581

18.16 用Build.scala取代build.sbt.....	582
18.17 在SBT中使用Maven仓库的类库.....	584
18.18 用Ant构建Scala项目	586
第19章 类型	589
19.1 创建使用泛型的类.....	591
19.2 创建一个接收简单泛型的方法	595
19.3 使用鸭子类型（结构化类型）	596
19.4 让可变集合非变.....	598
19.5 让不可变集合协变.....	600
19.6 创建所有元素都是基本类型的集合	602
19.7 给封闭模型选择性添加新行为	605
19.8 用类型构建功能.....	608
第20章 惯用法	612
20.1 创建没有副作用的方法（纯函数）	613
20.2 选择不可变对象.....	620
20.3 思考“面向表达式编程”	624
20.4 使用匹配表达式和模式匹配.....	627
20.5 从代码中根除null值.....	630
20.6 使用Option/Some/None 模式	634

前言

Scala是我用过的最有趣的编程语言，这本书是用Scala解决问题的cookbook。本书囊括了超过250个问题的解决办法，逾700个范例（我没有数过，但是我觉得是准确的）。

本书有如下几个特点：

- 作为一本cookbook，本书旨在通过对最常见的问题提供解决办法，以节省你学习的时间。
- 几乎所有的例子都在Scala解释器中演示。这样一来，不论你是在电脑旁、飞机上，或者在最喜爱的摇椅中阅读，都可以看到它们确切的输出结果。（好处是你经常会发现，“啊，原来是这么工作的。”）
- 本书不仅涵盖Scala语言，而且花了大量的篇幅介绍Scala的工具和类库，包括SBT、actor、集合类库和JSON处理。

本书发布前进行了更新，覆盖Scala 2.10.x和SBT 0.12.3。

Scala语言

我在推广Scala的演讲中说它是Ruby和Java的孩子：它具有Ruby的轻量，简洁和可读的特性，但它将编译好的class文件打包成Jar文件并且在JVM上运行；它使用特性（traits）和混入（mixins），感觉很动态，但它是静态类型的。它使用Actor模型去简化并发编程，更好地利用多核处理器。Scala这个名字来自单词scalable，顾名思义，它为世界上最繁忙的网站，包括Twitter、Netflix、Tumblr、LinkedIn、Foursquare等的运行提供动力。

在我看来，Scala并不适于编程的初级课程。相反，它是为专业程序员定制的强力语言。