

Head First Ruby

(中文版)

重视大脑的学习指南

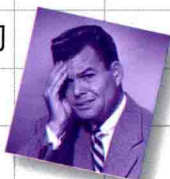


用更少的
代码做更
多的事情



用块轻松完成
繁重工作

避免难堪的
OO错误



掌握Ruby
标准库



用40余个Ruby
练习让大脑
开动起来

向世界提供
你的Web应用



Head First Ruby (中文版)

不是在做梦吧？一本关于Ruby的书居然没有抛出一大堆程序块、模块和异常？只是异想天开吧……



Jay McGavren 著

刘亮 朱涛江 等译

Beijing • Boston • Farnham • Sebastopol • Tokyo

O'REILLY®

图书在版编目 (CIP) 数据

深入浅出 Ruby: 中文版 / (美) 杰伊·格瑞恩 (McGavren, Jay) 著; 刘亮等译.

—北京: 中国电力出版社, 2017.3

书名原文: Head First Ruby

ISBN 978-7-5198-0057-4

I. ①深… II. ①杰… ②刘… III. ①计算机网络—程序设计 IV. ① TP393.09

中国版本图书馆 CIP 数据核字 (2016) 第 288489 号

北京市版权局著作权合同登记

图字: 01-2016-7465 号

©2015 by O'Reilly Media, Inc.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and China Electric Power Press, 2016.
Authorized translation of the English edition, 2015 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 2015。

简体中文版由中国电力出版社出版, 2017。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。

封面设计 /	Ellie Volkhausen, 张健
出版发行 /	中国电力出版社
地 址 /	北京市东城区北京站西街 19 号 (邮政编码 100005)
印 刷 /	航远印刷有限公司
开 本 /	850 毫米 × 980 毫米 16 开本 35.75 印张 771 千字
版 次 /	2017 年 3 月第 1 版 2017 年 3 月北京第 1 次印刷
印 数 /	0001 - 3000 册
定 价 /	98.00 元 (册)

敬告读者

本书如有印装质量问题, 我社发行部负责退换

版权专有 翻印必究

致世界各地的开源软件开发者们，
是你们让生活变得更美好。

Head First Ruby的作者



←
Jay McGavren

Jay McGavren 曾经为一家酒店服务公司开发自动化系统，一个同事向他推荐了《Perl语言编程》（就是大名鼎鼎的“大骆驼书”）。自此，他一发不可收拾，成为Perl的狂热支持者，因为他更喜欢自己动手编写代码，而不是等着一个10人的开发团队配置构建系统。这也让他有了一个想法，希望有朝一日能自己写一本技术书。

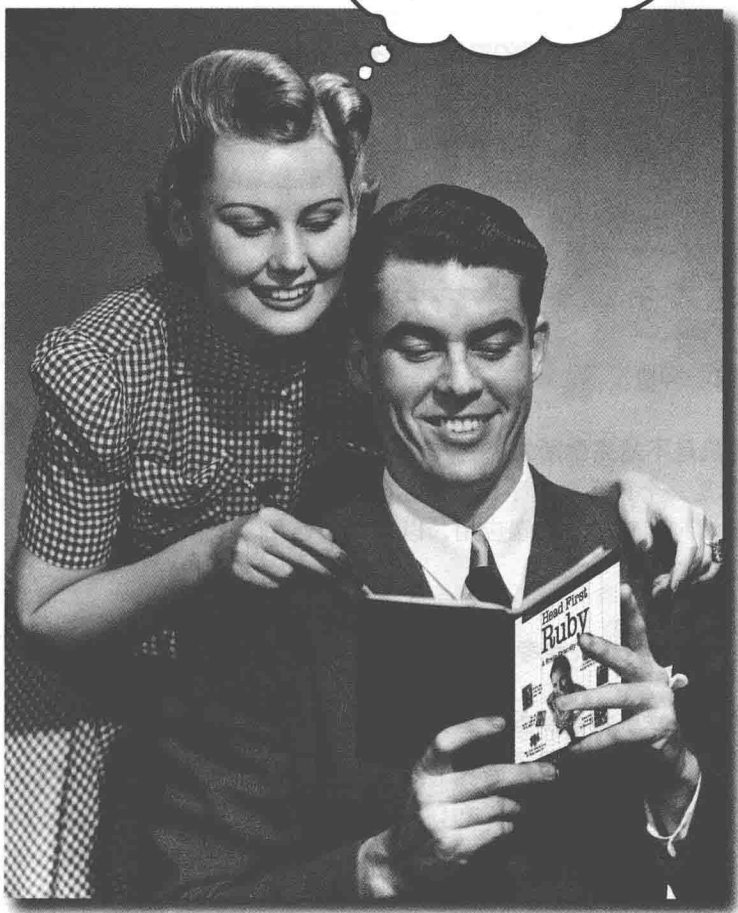
2007年，由于Perl的种种问题，Jay开始寻找新的解释语言。Ruby以其强大的面向对象特性、绝妙的库支持，以及无与伦比的灵活性，最终赢得了他的青睐。他已经使用Ruby完成了两个游戏库及一个生成艺术项目的开发，同时还是Ruby on Rails的自由作者。2011年以来他一直在从事在线开发人员教育领域的工作。

可以关注Jay的Twitter：<https://twitter.com/jaymcgavren>，或者访问他的个人网站<http://jay.mcgavren.com>。

如何使用这本书

引子

真是无法相信，这样一些东西也能放在一本Ruby书里！



有一个问题真是听得我们耳朵都磨出茧了：“到底为什么要把这些东西放在一本Ruby书里呢？”

谁适合看这本书？

如果能肯定地回答下面的所有问题：

- 1 你能用文本编辑器与计算机打交道吗？
- 2 你是不是想学一种编程语言让开发更加容易和高效？
- 3 你是不是更喜欢晚宴上轻松的交谈，而不是听别人长篇累牍的说教？

那么，这本书正是你需要的。

谁可能不适合看这本书？

如果满足下面任何一种情况：

- 1 你对计算机是不是完全陌生？
(不要求你是一个计算机高手，不过起码要了解文件夹和文件，知道如何打开一个终端应用，以及如何使用一个简单的文本编辑器。)
- 2 你是不是一个寻找有关参考书的高级开发人员？
- 3 你是不是对新鲜事物都畏首畏尾？是不是只喜欢简单的样式而不敢尝试把条纹和格子混在一起看看？你是不是觉得如果用猢猻来描述类继承的概念，这样的一本书肯定不是一本正经八百的技术书？

那么，这本书将不适合你。



[来自市场的声音：只要有信用卡，都可以拥有这本书。]

我们知道你在想什么

“这算一本正式的Ruby开发书吗？”

“这些图用来做什么？”

“我真能这样学吗？”

我们也知道你的大脑正在想什么

你的大脑总是渴求一些新奇的东西。它一直在搜寻、审视、期待着不寻常的事情发生。大脑的构造就是如此，正是这一点才让我们不至于墨守成规，能够与时俱进。

我们每天都会遇到许多按部就班的事情，这些事情很普通，对于这样一些例行的事情或者平常的东西，你的大脑又是怎么处理的呢？它的做法很简单，就是不让这些平常的东西妨碍大脑真正的工作。那么什么是大脑真正的工作呢？这就是记住那些确实重要的事情。它不会费心地去记乏味的东西，就好像大脑里有一个筛子，这个筛子会筛掉“显然不重要”的东西，如果遇到的事情枯燥乏味，这些东西就无法通过这个筛子。

那么你的大脑怎么知道到底哪些东西重要呢？打个比方，假如你某一天外出旅行，突然一只大老虎跳到你面前，此时此刻，你的大脑还有身体会做何反应？

神经元会“点火”，情绪爆发，释放出一些化学物质。

好了，这样你的大脑就会知道……

这肯定很重要！可不能忘记了！

不过，假如你正待在家里或者坐在图书馆里，这里很安全、很舒适，肯定没有老虎。你正在刻苦学习，准备应付考试。也可能想学一些比较难的技术，你的老板认为掌握这种技术需要一周时间，最多不超过十天。

这就存在一个问题。你的大脑很想给你帮忙。它会努力地把这些显然不太重要的内容赶走，保证这些东西不去侵占本不算充足的脑力资源。这些资源最好还是用来记住那些确实重要的事情，比如大老虎，遭遇火灾险情等。再比如，你的大脑会让你记住，绝对不能把聚会时狂欢的照片放在你的Facebook网页上。

没有一种简单的办法来告诉大脑：“嘿，大脑，真是谢谢你了，不过不管这本书多没意思，也不管现在我对它多么无动于衷，但我确实希望你能把这些东西记下来。”

你的大脑认为这真的很重要。



太好了，只有545页枯燥乏味的文字。

你的大脑认为，这些根本不值得记下来。



我们认为“Head First”的读者就是要学习的人。

那么，怎么学习呢？首先必须获得知识，然后保证自己确实不会忘记。这可不是填鸭式的硬塞。根据认知科学、神经生物学和教育心理学的最新研究，学习的途径相当丰富，绝非只是通过书本上的文字。我们很清楚怎么让你的大脑兴奋起来。

下面是一些Head First的学习原则：

看得到。与单纯的文字相比，图片更能让人记得住，通过图片，学习效率会更高（对于记忆和传递型的学习，甚至能有多达89%的效率提升）。而且图片更能让人看懂。以往总是把图片放在一页的最下面，甚至放在另外的一页上，与此不同，把文字放在与之相关的图片内部，或者在图片的周围写上相关文字，学习者的能力就能得到多至两倍的提高，从而能更好地解决有关的问题。

采用一种针对个人的交谈式风格。最新的研究表明，如果学习过程中采用一种第一人称的交谈方式直接向读者讲述有关内容，而不是用一种干巴巴的语调介绍，学生在学习之后的考试中成绩会提高40%。正确的做法是讲故事，而不是做报告。要用通俗的语言。另外不要太严肃。如果你面对着这样两个人，一个是你在餐会上认识的很有意思的朋友，另一个人学究气十足，喋喋不休地对你讲教，在这两个人中，你会更注意哪一个呢？

让学习的人想得更深。换句话说，除非你很积极地让神经元活动起来，否则你的头脑里什么也不会发生。必须引起读者的好奇，促进、要求并鼓励读者去解决问题、得出结论、产生新的知识。为此，需要发出挑战，留下练习题和拓宽思路的问题，并要求读者完成一些实践活动，让左右脑都开动起来，而且要利用多种思维。

引起读者的注意，而且要让他一直保持注意。我们可能都有过这样的体验，“我真的想把这个学会，不过看过一页后实在是让我昏昏欲睡。”你的大脑注意的是那些不一般、有意思、有些奇怪、抢眼的、意料之外的东西。学习一项有难度的新技术并不一定枯燥。如果学习过程不乏味，你的大脑很快就能学会。

影响读者的情绪。现在我们知道了，记忆能力很大程度上取决于所记的内容对我们的情绪有怎样的影响。如果是你关心的东西，就肯定记得住。如果让你感受到了什么，这些东西就会留在你的脑海中。不过，我们所说的可不是什么关于男孩与狗的伤心故事。这里所说的情绪是惊讶、好奇、觉得有趣、想知道“什么……”。还有一种就是自豪感，如果你解决了一个难题，学会了所有人都觉得很难的东西，或者发现你了解的一些知识竟是那些自以为无所不能的傲慢家伙所不知道的，此时就会有一种自豪感油然而生。

元认知：有关思考的思考

如果你真的想学，而且想学得更快、更深，就应该注意你怎样才会专注起来，考虑自己是怎样思考的，并了解你的学习方法。

我们中间大多数人长这么大可能都没有上过有关元认知或学习理论的课程。我们想学习，但是很少有人教我们怎么来学习。

不过，这里可以做一个假设，如果你手上有这本书，你想学如何用Ruby开发应用，而且可能不想花太多时间。如果你想把这本书中读到的知识真正用起来，就需要记住你读到的所有内容。为此，必须理解这些内容。要想最大程度地利用这本书或其他任何一本书，或者掌握学习经验，就要让你的大脑负起责来，要求它记住这些内容。

怎么做到呢？技巧就在于要让你的大脑认为你学习的新东西确实很重要，对你的生活有很大影响，就像老虎出现在面前一样。如若不然，你将陷入旷日持久的拉锯战中，虽然你很想记住所学的新内容，但是你的大脑却会竭尽全力地把它们拒之门外。

那么究竟怎样才能让你的大脑把编程看做是一只饥饿的老虎呢？

这两条路，一条比较慢，很乏味。另一条路不仅更快，还更有效。慢方法就是大量地重复。你肯定知道，如果反反复复地看到同一个东西，即便再没有意思，你也能学会并记住。如果做了足够的重复，你的大脑就会说，“尽管看上去这对他来说好像不重要，不过，既然他这样一而再、再而三地看同一个东西，所以我觉得这应该是重要的。”

更快的方法是尽一切可能让大脑活动起来，特别是开动大脑来完成不同类型的活动。如何做到这一点呢？上一页列出的学习原则正是一些主要的可取做法，而且经证实，它们确实有助于让你的大脑全力以赴。例如，研究表明，把文字放在所描述图片的中间（而不是放在这一页的别处，比如作为标题，或者放在正文中），这样会让你的大脑更多地考虑这些文字与图片之间有什么关系，而这就会让更多的神经元点火。让更多的神经元点火 = 你的大脑更有可能认为这些内容值得关注，而且很可能需要记下来。

交谈式风格也很有帮助，当人们意识到自己在与“别人”交谈时，往往会更专心，这是因为他们总想跟上谈话的思路，并能做出适当的发言。让人惊奇的是，大脑并不关心“交谈”的对象究竟是谁，即使你只是与一本书“交谈”，它也不会在乎！另一方面，如果写作风格很正统、干巴巴的，你的大脑就会觉得，这就像坐在一群人当中被动地听人做报告一样，很没意思，所以不必在意对方说的是什么，甚至可以打瞌睡。

不过，图片和交谈风格还只是开始而已，能做的还有很多。



我们是这么做的

我们用了**很多图**，因为你的大脑更能接受看得见的东西，而不是纯文字。对你的大脑来说，一幅图顶得上一千个字。如果既有文字又有图片，我们会把文字放在图片当中，因为文字处在所描述的图片中间时，大脑的工作效率更高，倘若把这些描述文字作为标题，或者“淹没”在别处的大段文字中，就达不到这种效果了。

我们采用了**重复手法**，会用不同方式，采用不同类型的媒体，运用多种思维手段来介绍同一个东西，目的是让有关内容更有可能储存在你的大脑中，而且在大脑中多个区域都有容身之地。

我们会用你**想不到的方式**运用概念和图片，因为你的大脑喜欢新鲜玩艺。在提供图和思想时，至少会含着一些情绪因素，因为如果能产生情绪反应，你的大脑就会投入更大的注意。而这会让你感觉到这些东西更有可能要被记住，其实这种感觉可能只是有点幽默，让人奇怪或者比较感兴趣而已。

我们采用了一种针对个人的**交谈式风格**，因为当你的大脑认为你在参与一个会谈，而不是被动地听一场演示汇报时，它就会更加关注。即使你实际上在读一本书，也就是说在与书“交谈”，而不是真正与人交谈，但这对你的大脑来说并没有什么分别。

在这本书里，我们加入了**实践活动**，因为与单纯的阅读相比，如果能实际做点什么，你的大脑会更乐于学习，更愿意去记。这些练习都是我们精心设计的，有一定的难度，但是确实能做出来，因为这是大多数人所希望的。

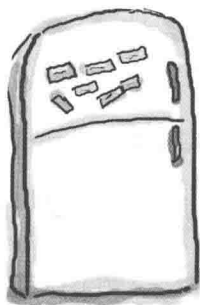
我们采用了**多种学习模式**，因为尽管你可能想循序渐进地学习，但是其他人可能希望先对整体有一个全面的认识，另外可能还有人只是想看看一个例子。不过，不管你想怎么学，要是同样的内容能以多种方式来表述，这对每一个人都会有好处。

这里的内容不只是单单涉及左脑，也不只是让右脑有所动作，我们会让你的**左右脑**都动起来，因为你的大脑参与得越多，你就越有可能学会并记住，而且能更长时间地保持注意力。如果只有一半大脑在工作，通常意味着另一半有机会休息，这样你就能更有效率地学习更长时间。

我们会**讲故事**，留练习，从**多种不同的角度**来看同一个问题，这是因为，如果要求大脑做一些评价和判断，它就能更深入地学习。

我们会给出一些练习，还会问一些**问题**，这些问题往往没有很直接的答案，通过克服这些**挑战**，你就能学得更好，因为让大脑真正做点什么的话，它就能学会并记住。想想吧，如果只是在体育馆里看着别人流汗，这对于保持你自己的身材肯定不会有什么帮助，正所谓临渊羡鱼，不如退而结网。不过另一方面，我们会竭尽所能不让你钻牛角尖，把劲用错了地方，而是能把功夫用在点子上。也就是说，你不会为搞定一个难懂的例子而耽搁，**也不会花太多时间去弄明白一段艰涩难懂而且通篇行话的文字**，我们的描述也不会太过简洁而让人无从下手。

我们用了**拟人手法**。在故事中，在例子中，还有在图中，你都会看到人的出现，这是因为你本身是一个人，不错，这就是原因。如果和人打交道，相对于某件东西而言，你的大脑会更为关注。



可以用下面的方法让你的大脑就范

好了，我们该做的已经做了，剩下的就要看你自己的了。以下提示可以作为一个起点：听一听你的大脑是怎么说的，弄清楚对你来说哪些做法可行，哪些做法不能奏效。要尝试新鲜事物。

把这一页撕下来，贴到你的冰箱上。

1 慢一点。你理解的越多，需要记的就越少。

不要光是看看就行了。停下来，好好想一想。书中提出问题的時候，你不要直接去翻答案。可以假想真的有人在问你这个问题。你让大脑想得越深入，就越有可能学会并记住它。

2 做练习，自己记笔记。

我们留了练习，但是如果这些练习的解答也由我们一手包办，那和有人替你完成工作有什么分别？不要只是坐在那里看着练习发呆。拿出笔来，写一写、画一画。大量研究都证实，学习过程中如果能实际动手，这将改善你的学习。

3 阅读“没有傻问题”部分。

顾名思义，这些问题不是可有可无的旁注，它们绝对是核心内容的一部分！千万不要跳过去不看。

4 上床睡觉之前不要再看别的书，至少不要看其他有难度的东西。

学习中有一部分是在你合上书之后完成的（特别是，要把学到的知识长久地记住，这往往无法在看书的过程中做到）。你的大脑也需要有自己的时间，这样才能再做一些处理。如果在这段处理时间内你又往大脑里灌输了新的知识，那么你刚才学的一些东西就会丢掉。

5 讲出来，而且要大声讲出来。

说话可以刺激大脑的另一部分。如果你想看懂什么，或者想更牢地记住它，就要大声地说出来。更好的办法是，大声地解释给别人听。这样你会学得更快，而且可能会有以前光看不说时不曾有的新发现。

6 要喝水，而且要多喝点水。

能提供充足的液体，你的大脑才能有最佳表现。如果缺水（可能在你感觉到口渴之前就已经缺水了），学习能力就会下降。

7 听听你的大脑怎么说。

注意一下你的大脑是不是负荷太重了。如果发现自己开始浮光掠影地翻看，或者刚看的东西就忘记了，这说明你该休息一会了。达到某个临界点时，如果还是一味地向大脑里塞，这对于加快学习速度根本没有帮助，甚至还可能影响正常的学习进程。

8 要有点感觉。

你的大脑需要知道这是很重要的东西。要真正融入到书中的故事里。为书里的照片加上你自己的图题。你可能觉得一个笑话很憋脚，不太让人满意，但这总比根本无动于衷要好。

9 编写大量软件！

要学习编程，没有别的办法，只能通过编写大量代码。这本书正是要这么做。编写代码是一种技巧，要想在这方面擅长，只能通过实践。我们会给你提供大量实践的机会：每一章都留有练习，提出问题让你解决。不要跳过这些练习，很多知识都是在完成这些练习的过程中学到的。我们为每个练习都提供了答案，如果你实在做不出来（很容易被一些小问题卡住），看看答案也无妨！不过在看答案之前，还是要尽力先自己解决问题。而且在读下一部分之前，一定要确确实实地掌握前面的内容。

重要说明

要把这看做是一个学习过程，而不要简单地把它看成是一本参考书。我们在安排内容的时候有意做了一些删减，只要是对有关内容的学习有妨碍，我们都毫不留情地把这些部分一律删掉。另外，第一次看这本书的时候，要从第一页看起，因为书中后面的部分会假定你已经看过而且学会了前面的内容。

如果用另外某种语言做过一些编程对你会有帮助。

大多数开发人员都是在学过另外某种编程语言之后才发现Ruby（他们通常还会寻求那种语言的帮助）。我们会介绍一些基础知识，即使完全是新手也能掌握，不过我们不打算深入探讨什么是变量，或者一个if语句是如何工作的。如果你之前至少做过一些编程，会更轻松一些。

我们不会介绍已创建的每一个类、库和方法。

Ruby提供了大量内置的类和方法。没错，它们都很有意思，不过我们没办法全部都介绍，否则这本书的篇幅就会是现在的两倍。我们的重点是像你这样的初学者关心的核心类和方法。这本书肯定能让你深入了解这些类和方法，而且很自信地知道该如何以及何时使用。无论如何，读完这本书之后，你完全可以拿起任意一本参考书，快速掌握这里没有谈到的类和方法。

书里的实践活动不是可有可无的。

这里的练习和实践活动不是可有可无的装饰和摆设。它们也是这本书核心内容的一部分。其中有些练习和活动有助于记忆，有些能够帮助你理解，还有一些对于如何应用所学的知识很有帮助。千万不要跳过、不做这些练习。

我们有意安排了许多重复，这些重复非常重要。

Head First系列的书有一个与众不同的地方，这就是，我们希望你确确实实地学会，另外希望在学完这本书之后你能记住学过了什么。大多数参考书都不太重视重复和回顾，但是由于这是一本有关学习的书，你会看到一些概念一而再、再而三地出现很多次。

代码示例尽可能简洁。

如果只是为了找出你想要理解的两行代码而要费劲地查看200行代码，这是很让人头疼的。这本书中的大多数例子都没有太多的上下文，从而保证真正需要你了解的部分清晰而简明。所以不要期望这里会给出健壮甚至完备的代码。这是你读完这本书之后要完成的作业。本书中的例子是为学习而编写的，不能保证提供完备的功能。

所有示例文件都放在网上可供你下载。下载地址为：<http://headfirstruby.com/>。

致谢

致Head First系列创始人：

首先衷心感谢Head First系列的创始人Kathy Sierra和Bert Bates。10年前第一次接触到这个系列时我就深深地为之着迷，不过绝对没有想到将来我也能写一本关于Head First的书。感谢你们开创了这种无与伦比的教学方式！

尤其要感谢Bert，感谢他对早期书稿的大量反馈，那时我还没有很好地掌握Head First之道。正是因为有了你，这本书才变得更好，也让我成为一个更好的作者！

致O'Reilly：

非常感谢我们的编辑Meghan Blanchette。她总是坚定不移地力求图书尽善尽美，要求我通过第2次（以及第3次）修订让这本书日臻完美。感谢Mike Loukides帮助我最早与O'Reilly接触，另外感谢Courtney Nash促成了这个项目。

还要感谢O'Reilly的每一位给予过帮助的人，特别是Rachel Monaghan、Melanie Yarbrough和制作团队的其余人员。

致技术审校人员：

每个人都会犯错，不过非常幸运我有Avdi Grimm、Sonda Sengupta、Edward Yue Shung Wong和Olivier Lacan这样一些非凡的技术审校帮我发现错误。你们永远也不会知道他们找出多少错误，因为我已经销毁了所有“罪证”。不过他们的帮助和反馈绝对不可缺少，在此深深表示感谢！

更多感谢：

感谢Ryan Benedetti，在我写这本书前几章时，感谢你让这个项目走出困境。感谢Deborah Robinson在排版上的帮助。感谢Janet McGavren和John McGavren的校对。感谢Lenny McGavren提供的摄影。还要感谢早期版本的读者，特别是Ed Fresco和John Larkin，感谢他们找出印刷错误和其他问题。最后要感谢Ruby Rogues Parley论坛成员对代码示例的反馈。

最重要的，我要感谢Christine、Courtney、Bryan、Lenny和Jeremy的耐心和支持。很高兴我又回到你们身边！

本书使用Ruby logo, Copyright © 2006, Yukihiro Matsumoto。得到Creative Commons Attribution-ShareAlike 2.5 License许可 (<http://creativecommons.org/licenses/by-sa/2.5/>)。

Safari®图书在线



Safari图书在线 (www.safaribooksonline.com) 是一个应需而变的数字图书馆, 通过图书和视频方式提供世界顶尖作者在技术

和商业领域积累的专家经验。技术专家、软件开发人员、Web设计人员和企业以及有创意的专业人员都使用Safari图书在线作为其主要资源来完成研究、解决问题、深入学习和资质培训。

Safari图书在线为机构、政府部门和个人提供了多种产品组合和定价程序。订阅者可以在一个可以快捷搜索的数据库中访问多家出版社提供的成千上万种图书、培训视频和正式出版前手稿, 如O'Reilly Media、Prentice Hall Professional、Addison-Wesley Professional、Microsoft Press、Sams、Que、Peachpit Press、Focal Press、Cisco Press、John Wiley & Sons、Syngress、Morgan Kaufmann、IBM Redbooks、Packt、Adobe Press、FT Press、Apress、Manning、New Riders、McGraw-Hill、Jones & Bartlett、Course Technology以及其他数十家出版公司。关于Safari图书在线的更多信息, 请访问我们的在线网站。

目录(概览)

引子	xxiii
1 少说多做：编写你想要的代码	1
2 方法与类：合理组织	35
3 继承：依靠你的双亲	75
4 初始化实例：从头开始	107
5 数组与块：比循环还要好	155
6 块返回值：我该如何处理？	193
7 散列：为数据加标签	225
8 引用：混乱的信号	257
9 mixin：混合	285
10 comparable与enumerable：现成的混入模块	311
11 文档：阅读手册	333
12 异常：处理意外事件	359
13 单元测试：代码质量保证	389
14 Web应用：提供HTML	421
15 保存与加载数据：保留下来	455
附录i 其他：十大主题（我们没有介绍的）	499
索引	515

目录

引子

让你的大脑学习Ruby。你想坐下来学点东西，可是你的大脑却总在帮倒忙，一直在告诉你学这些不重要。你的大脑在想，“还是把空间留给更重要的事情吧，比方说要躲开哪些野兽，还有光着身子滑雪不太好吧。”那么你该如何骗过大脑，让它认为如果你不学会Ruby开发就无法生活？

谁适合看这本书？	xxiv
我们知道你在想什么	xxv
我们也知道你的大脑正在想什么	xxv
元认知：有关思考的思考	xxvii
我们是这么做的	xxviii
重要说明	xxx
致谢	xxxix

少说多做

编写你想要的代码

你肯定想知道这个神奇的Ruby语言到底是什么。如果是这样，我们先来问问你：你想要高效率吗？是不是觉得其他语言中那些额外的编译器、库、类文件和按键可以让你得到更完备的产品，会让同事更仰慕你，让顾客更满意？想不想要一种语言来帮你考虑所有这些细节？如果有时候你不想再维护样板代码，想要处理你自己的问题，那么Ruby正是你需要的。Ruby允许你用更少的代码做更多的事情。

Ruby哲学	2
交互式使用Ruby	5
你的第一个Ruby表达式	6
数学运算和比较	6
字符串	6
变量	7
在对象上调用方法	8
输入、存储和输出	12
运行脚本	13
注释	14
“puts”和“print”	14
方法参数	15
“gets”	15
字符串内插	16
用“inspect”和“p”方法检查对象	18
字符串中的转义序列	19
在字符串对象上调用“chomp”	20
生成一个随机数	22
转换为字符串	23
将字符串转换为数字	25
条件语句	26
与“if”对应的是“unless”	29
循环	30
你的Ruby工具箱	34

