

# BOOK REVIEWS

BY JAMES PYLES

## Ruby by Example: Concepts and Code

Do you want to learn to program in Ruby? Do you want to learn to program in Ruby using real-life examples rather than contrived “textbook” code? Baird’s *Ruby by Example* turns out to be a really good reference guide and general learning tool for picking up Ruby in a practical way. Let me explain.

First, the author must really love Ruby. He’s a Ph.D., and as you know, they have to write a pesky little thing called a “dissertation” to use those initials behind their name.

Turns out that Baird originally wrote his dissertation in Python (which is fine and good), but after finding Ruby, he rewrote the whole thing.

If dedication to a programming language means anything, then you should believe the author’s focus in in the right place for writing such a book.

Baird says he wrote the book for the Ruby newcomer but not the programming newcomer. That means this book won’t teach you beginning programming principles. You need to acquire those skill sets first and then come back and get this book.

For those of you who know how to program and have an interest in picking up Ruby, read on.

You may now choose your weapons – or in this case, your development platform. The book begins by showing you how to install Ruby on Windows or

Unix/Unix-like (Linux, BSD, or Mac) platforms. However, the “What is Ruby” (and how do I lay my hands on it?) section is quite brief, so don’t blink. Before you know it, you are learning the language proper, albeit at the “Hello World!” level in Chapter 1.

Kevin Baird writes like a Ph.D., and that’s not necessarily a compliment. I don’t think his writing style will prevent a person with a lesser degree or a non-degreed but well-experienced programmer from understanding or learning from this book, but the general rule for writing (at least from my perspective) is: Don’t use any more words than you need to get your point across.

Maybe Baird’s dissertation experience spilled over into this text.

I really liked the little gray “Who should read this chapter?” box on page 1, which was used to open up Chapter 1. I wish the other chapters had similar introduction boxes because it provided a very nice, succinct message about what you were about

to learn. It also would have been a really good way to tell the reader who likes to skip ahead whether they needed to review any of the prior chapters (or any other resources for that matter) first.

By now, you’re probably convinced that I didn’t like the book. However, that’s not true.

I especially liked the “by example” approach the book takes (although that’s as much a reflection of this No Starch series as it is of the writer) in dissecting a series of small scripts in bite-sized chapters to teach the fundamentals of Ruby.

You might be tempted to think of this as a “hacks” book because each of the code chunks are numbered in the table of contents, but that’s just for the sake of organization. The hacks in a such a book aren’t necessarily associated with one another, and the book as a whole doesn’t necessarily present a unified purpose or goal (beyond presenting a list of hacks).

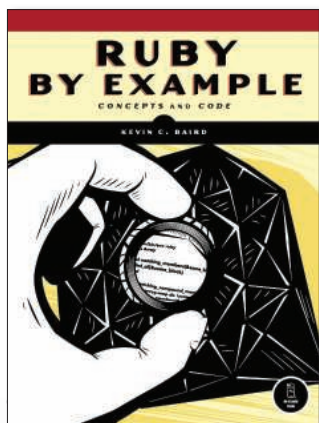
Kevin Baird’s *Ruby by Example* exists to instruct the audience in learning and using Ruby, culminating with “A Simple Rails Project” in Chapter 13.

One of my favorite languages is Python, and the similarities between Python and Ruby presented a compelling hook from the first few pages.

Regardless of your leanings or preferences, though, if you believe learning Ruby is in your future, pick up a copy of this book. You’ll start acquiring hands-on experience from the very beginning.

To download source code or the Ruby by Example Rails Application, or to read Chapter 6: Functionalism with Blocks and Procs, visit: <http://nostarch.com/ruby.htm>.

**Kevin C. Baird**  
**Paperback, 326 Pages**  
**No Starch Press, 2007**  
**ISBN-10: 1593271484**  
**ISBN-13: 978-1593271480**  
**£ 16.41, US\$ 29.95, EUR 20.42**



## The Productive Programmer

The structure of the *The Productive Programmer* isn't particularly important relative to learning the content, so you can read it cover to cover or just drill down into the chapters that interest you, according to author Neal Ford. That said, the book is broken into two major parts: Mechanics and Practice. It also starts with an Introduction, so I don't particularly buy the idea that you can just jump in anywhere.

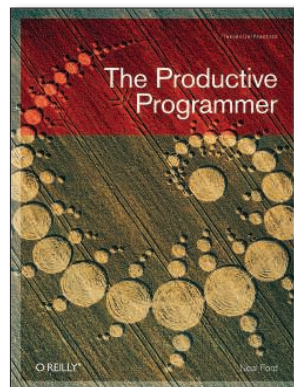
Part One, Mechanics, addresses managing or tweaking your development platform to get the most out of it. Ford isn't kidding when he says the book is platform independent. The first few pages of this section mention Windows, Mac, and Linux.

He also wasn't kidding when he said he wrote the book for programmers and that the book assumes you have a coder's general level of knowledge.

If you start reading this book and get that deer-in-the-headlights feeling, you

probably need more programming experience.

Assuming you do have the experience and are reading the book in a linear fashion, you'll scream through Part One in no time (it's fewer than 100 pages). You probably will want to slow down, though, to absorb all of the details that are presented. Part Two, Practice, is about specific tasks relative to coding, although the first chapter in this section is about testing code rather than coding as such. The owner of this book will enjoy the treasures it contains, such as the knowledgeable presentation of analytical tools. I was a little disappointed that Ford seemed to bias this set of information more for Windows and Mac users. My personal platform bias is Linux and more recommendations in that direction would have been appreciated. I'd also recommend visiting the errata pages at



O'Reilly (after all, no one is perfect). A few notable design "issues" pop up within the book's pages.

Aside from that, I think that most programmers would benefit from owning and reading Neal Ford's book.

He does a good job of leveraging both his programming and training background in producing a sleek, streamlined guide that will help you automate tasks and pump up your code.

**Neal Ford**  
**Paperback, 222 Pages**  
**O'Reilly Media Inc., 2008**  
**ISBN-10: 0596519788**  
**ISBN-13: 978-0596519780**  
**£ 21.98, US\$ 39.99, EUR 27.26**

# WANT TO KNOW WHAT'S UP NEXT?



**SUBSCRIBE TO LINUX  
MAGAZINE PREVIEW,  
OUR FREE MONTHLY  
EMAIL NEWSLETTER!**

**[WWW.LINUX-MAGAZINE.COM/NEWSLETTER](http://WWW.LINUX-MAGAZINE.COM/NEWSLETTER)**