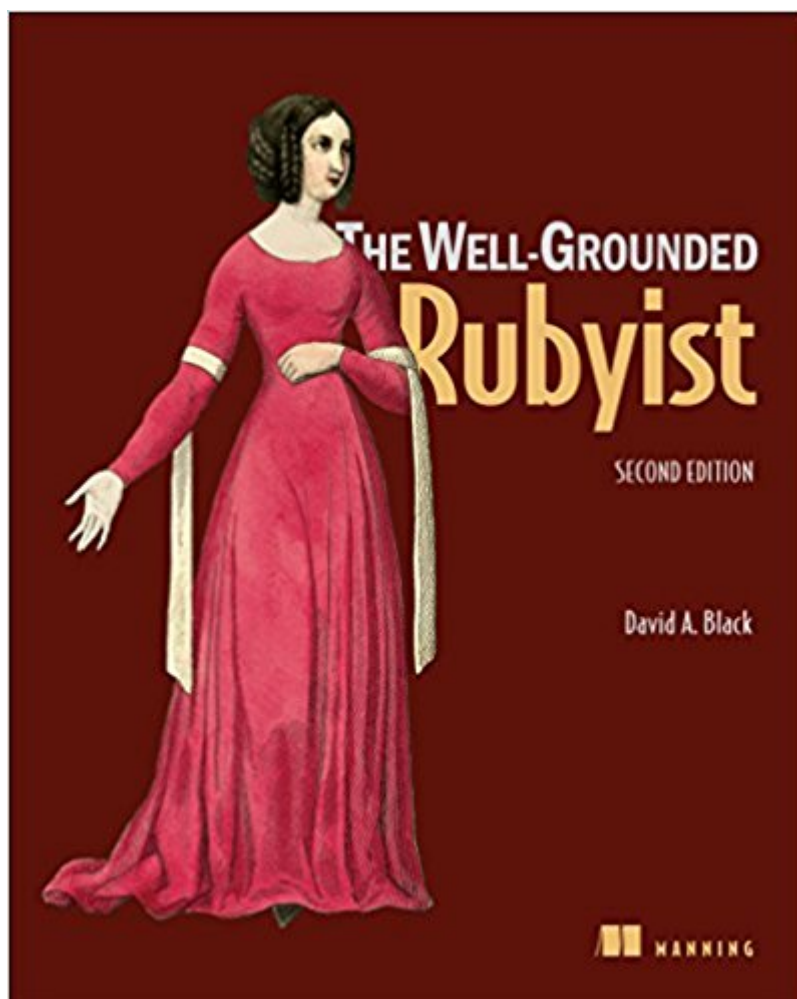


The book was found

The Well-Grounded Rubyist



Synopsis

SummaryThe Well-Grounded Rubyist, Second Edition addresses both newcomers to Ruby as well as Ruby programmers who want to deepen their understanding of the language. This beautifully written and totally revised second edition includes coverage of features that are new in Ruby 2.1, as well as expanded and updated coverage of aspects of the language that have changed. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

About the TechnologyThis is a good time for Ruby! It's powerful like Java or C++, and has dynamic features that let your code react gracefully to changes at runtime. And it's elegant, so creating applications, development tools, and administrative scripts is easier and more straightforward. With the long-awaited Ruby 2, an active development community, and countless libraries and productivity tools, Ruby has come into its own.

About the BookThe Well-Grounded Rubyist, Second Edition is a beautifully written tutorial that begins with your first Ruby program and goes on to explore sophisticated topics like callable objects, reflection, and threading. The book concentrates on the language, preparing you to use Ruby in any way you choose. This second edition includes coverage of new Ruby features such as keyword arguments, lazy enumerators, and `Module#prepend`, along with updated information on new and changed core classes and methods.

What's InsideClear explanations of Ruby concepts
Numerous simple examples
Updated for Ruby 2.1
Prepares you to use Ruby anywhere for any purpose

About the AuthorDavid A. Black is an internationally-known software developer, author, trainer, speaker, and event organizer. He is a Ruby standard library contributor and one of the founders of Ruby Central, Inc., the parent organization of the official international Ruby and Ruby on Rails conferences. David works at 2U, Inc. in New York City.

Table of Contents
PART 1 RUBY FOUNDATIONS
Bootstrapping your Ruby literacy
Objects, methods, and local variables
Organizing objects with classes
Modules and program organization
The default object (self), scope, and visibility
Control-flow techniques
PART 2 BUILT-IN CLASSES AND MODULES
Built-in essentials
Strings, symbols, and other scalar objects
Collection and container objects
Collections central: Enumerable and Enumerator
Regular expressions and regexp-based string operations
File and I/O operations
PART 3 RUBY DYNAMICS
Object individuation
Callable and runnable objects
Callbacks, hooks, and runtime introspection

Book Information

Paperback: 536 pages

Publisher: Manning Publications; 2 edition (July 4, 2014)

Language: English

ISBN-10: 1617291692

ISBN-13: 978-1617291692

Product Dimensions: 1.2 x 7.5 x 9.2 inches

Shipping Weight: 2 pounds (View shipping rates and policies)

Average Customer Review: 4.8 out of 5 stars [See all reviews](#) (39 customer reviews)

Best Sellers Rank: #62,270 in Books (See Top 100 in Books) #8 in [Books > Computers & Technology > Programming > Languages & Tools > Ruby](#) #29 in [Books > Textbooks > Computer Science > Object-Oriented Software Design](#) #62 in [Books > Textbooks > Computer Science > Software Design & Engineering](#)

Customer Reviews

I reviewed the book (courtesy of Manning). As a Ruby developer I found this book very accurate and serious. It covers pretty well the Ruby language and shows deeply a lot of aspects. As Ruby and the standard library is quite a big subject, the book cannot cover everything but the author was able to select the part of the standard library that is fundamental and that is not going to disappear in next Ruby releases (for instance the "curses" library was removed and is available as a gem as from Ruby 2.1). The intended audience is definitely programmers knowing object oriented programming who either want to refresh their Ruby knowledge, either want to learn the language as a new language. But as the book states, it does not teach programming. But we feel that the author really wants to be precise and comprehensive, he struggles with the need to explain things that some people might not know, while many people already know... For Ruby programmers the book is useful as a reference for parts of the language he/she does not use on a daily basis. The book covers the regular expressions and can be useful for the use of regexp beyond its usage with the Ruby language. But the book has some weaknesses. The chapter 6 is not precise enough about exceptions, "can i catch more than 1 exception?", "what if I write rescue A, then B, if B derives from A, is B ever going to be caught?". In chapter 7, the "Bid" example presents the `operator` (object comparison) but is not clear enough that comparing objects does not necessarily involve the class relationship but is "duck-typing" (the nature of the Ruby language).

First, a brief introduction: I'm a freelance developer, and I've worked several times with Ruby. As I have started to read, I have to say that I was fascinated by the style of writing. The writer composes expressions very well and is able to hold your attention. Even having used it some

hundreds of hours and went through the documentation few times, already since the first pages I can find some arguments and concepts that before were absolutely obscure to me. Keeping a practical framework, it densely packs the theory that every rubyist should know, comprehensive of conventions and suggestions on how to use its features. Maybe it falls sometimes on excessive exposure - like the info about the configuration at the beginning, that may be liked or not, depending on the reader. It starts with fundamentals and syntax, going on with the built-in features, and finishing with the special Ruby OOP features, more advanced concepts. Although it may seem only for beginners, an intermediate can find many useful things, especially if you didn't still have attended a complete course on the features of the language (those things that you were supposed to know but you always find as new on StackOverflow..). It will change your mindset about how this language should be used. The style of writing is very near to the voice of a cautious grandfather who advises you on what to do or not. If you want to get better in Ruby and get some solid foundations of the language It will deepen also your understanding of the language, its internal mechanisms and why some features have been introduced (e.g., why ! or ? are at the end of methods, how objects work internally, etc.).

[Download to continue reading...](#)

The Well-Grounded Rubyist Well Grounded: Using Local Land Use Authority to Achieve Smart Growth (Environmental Law Institute) The Rapid Rubyist Cancelled, Delayed, Grounded: Law for the Frustrated Air Traveler The Discovery of Grounded Theory: Strategies for Qualitative Research Constructing Grounded Theory (Introducing Qualitative Methods series) Grounded! Amazing Classroom Demonstrations in Soil Mechanics Thanksgiving: How to Cook It Well: How to Cook It Well American Cancer Society Complete Guide to Nutrition for Cancer Survivors: Eating Well, Staying Well During and After Cancer Breathe Well and Live Well with COPD: A 28-Day Breathing Exercise Plan Obsessive-Compulsive Disorders: A Complete Guide to Getting Well and Staying Well The Well-Armored Teen: Easy Tools Protect Your Teen and Tween From Sexual Abuse, Bullying, and Exploitation (The Well-Armored Child Library Book 1) Healthy at Home: Get Well and Stay Well Without Prescriptions All's Well That Ends Well (Folger Shakespeare Library) Tired of Trying to Measure Up: Getting Free from the Demands, Expectations, and Intimidation of Well-Meaning People Why Should I Eat Well? (Why Should I? Books) What's So Yummy?: All About Eating Well and Feeling Good Python: Learn Python in One Day and Learn It Well. Python for Beginners with Hands-on Project. (Learn Coding Fast with Hands-On Project Book 1) All Shall Be Well Living Well and Dying Faithfully: Christian Practices for End-of-Life Care

[Dmca](#)