

まえがき

本書を手にとってください、どうもありがとうございます。本書はプログラミング言語Rubyの言語仕様や開発の現場で役立つRubyの知識を説明した本です。

20年以上前に生まれたRubyはもはや新しい言語ではありません。さらに、Rubyを使ったWebアプリケーションフレームワークであるRuby on Rails (以下、Rails) が非常に有名になったこともあり、Rubyは歴史が長く、人気も高いプログラミング言語の1つと言えるでしょう。それゆえ、すでに多くの入門本が出版されています。また、本だけではなく、ネット上にもオンラインの学習コンテンツがたくさん提供されています。これだけ選択肢があると、自分にぴったり合った本や学習コンテンツを見つけるのはなかなか大変ですよね。きっと本書もその選択肢の1つになると思いますが、本書がみなさんのニーズに合っているかどうか、最初にチェックさせてください。

- Q1. すでにほかのプログラミング言語で3年以上の開発経験がある。もしくは「プログラミング歴=Ruby歴」で、Rubyを始めてから半年以上経っている。(Yes/No)
- Q2. Rubyを学習する動機はRailsアプリケーションの開発に役立つためだ。(Yes/No)
- Q3. すでに仕事でRubyを使っている。もしくはこれからRubyを使った仕事に就こうとしている。(Yes/No)

みなさんはいくつYesになりましたか？ 3つともYesになった方は、本書がみなさんのニーズを満たしている可能性が高いです！ 反対に、3つともNoだった方は残念ながら本書がお役に立つ部分は少ないかもしれません。Yesが1つ、または2つだった方は、もしかするとみなさんのニーズにマッチする部分があるかもしれないので、もう少しじっくり本書の内容を吟味してください。

■本書はどんな人に向いているか？

ではここで、先ほどの質問の背景をそれぞれ簡単に説明しておきましょう。

最初の質問は、本書がある程度のプログラミング経験を持っている人を対象にしていることを意味しています。プログラミングをまったくやったことがない、という人は読み進めるのがちょっと難しいかもしれません。本書は「今までほかの言語をやっていたが、これからRubyを始めてみたい人」や「Rubyプログラミングを始めてしばらく経ったが、まだまだ自信が持てない人」を想定して書かれています。

また、本書はあくまでRubyの入門本であり、Railsの入門本ではありません。ですが、本書はRailsの存在を強く意識しながら書かれています。筆者自身も普段はRailsアプリケーションの開発をメインでやっています。Railsに特化した知識は説明しないものの、筆者の経験上、「Railsアプリを開発するなら、ここは必ず押さえておきたい」というトピックは必ずカバーするようにしました。これが2つめの質問の背景です。もちろん、Rails以外のRubyプログラミングでも本書は役立ちますが、その場合はもしかすると「こんなことが知りたい」という内容が多少不足しているかもしれません。目次にざっと目を通して、自分の知りたいトピックがカバーされているかどうかを確認してください。

最後の質問は、読者のみなさんがどれくらい真剣にRubyに取り組む必要があるのかを確認したくて尋ねた質問です。本書はなかなかのボリュームです。最初から最後まで読み進めるためにはそれなりの時間がかかると思います。ですが、開発の現場で本格的なRubyプログラムを書くためには、最低でもこれぐらいの知識が必要になります。本気でRubyをやっていくつもりなら、本書を手にとってRubyをしっかり勉強しましょう。逆に、

「ちょっとRubyをかじってみたい」という程度のニーズであれば、本書は少し網羅的すぎるかもしれません。

■ 読んで終わり、ではなく手と頭を動かそう

ところで、みなさんはこれまでどんなふうにプログラミングの勉強をしてきましたか？ 筆者もこれまでいろんな方法で勉強してきましたが、その中で「一番失敗したなあ」と思うのは、「本を読むだけでわかった気になってしまうこと」です。本を読んでその内容を理解すると、それだけで「わかったつもり」になりがちです。しかし、いざ自分でプログラムを書こうとするとまったく手が動かず、何をどうすればいいんだと途方に暮れてしまう……そんな経験が筆者にはありました。

その経験から筆者が学んだのは「自分の手と頭を動かさなければ、プログラミングは学べない」ということです。そこで、本書では読者のみなさんに手と頭を動かしてもらうために、簡単なプログラミング問題を各章に用意しています。このプログラミング問題はぜひ手元のマシンを使って実際にプログラムを組んでみてください。「本を読んで終わり」にするよりも、ずっと高い学習効果が得られるはずです。

■ 分厚い本は難しい？

最後に、本書はとても分厚い本ですが、「分厚い本＝難しい本」だと思わないでください。いや、難しい内容が多少登場することは否定しませんが、分厚くなった理由はそれだけではありません。本書のボリュームが大きくなったのは以下のような理由からです。

- ・ 文章による説明だけでなく、サンプルコードをふんだんに用いて説明している。
- ・ 難しい内容もできるだけわかりやすくなるよう、筆者が自分の言葉でできるだけかみ砕いて説明している。
- ・ 高度な話題ばかりでなく、Rubyをまったく知らない人でも無理なく読み進められるよう、初歩的な話題から順番に説明している。

つまり、筆者が目指したのは「分厚くて難しい技術書」ではなく、「分厚いが非常にわかりやすい技術書」です。

また、本書は技術的に簡単な内容ほど前半に、高度な話題ほど後半に配置しています。これは本書全体を通しての章構成もそうですし、各章ごとの説明内容もそうになっています。それゆえ、ほかのRubyの入門書と比較すると少し変わった構成になっているかもしれません。しかし、前から順番に読んでいけば、前半で学んだ基礎知識のおかげで、後半の高度な知識もすんなり頭に入ってくるはずです。

■ これから本書を読むみなさんに向けて

もちろん、このまえがきで説明した内容が本当かどうかは、みなさんの感想次第です。ですが、このまえがきを読んで「なんかおもしろそう」「これは自分に向いているかも」と思った人は、ぜひこの続きを読み進めてほしいと思います。この本を最後まで読み終わったら、まえがきの内容が合っていたかどうか、あとがきのページでみなさんと答え合わせをしましょう。

「本書を読む前に比べると全然コードが違って見える！」

「Rubyの言語機能を使って、こんなにシンプルで読みやすいコードが書けるようになった！」

本書を読み終えたみなさんから、そんな声が聞こえてくることを願っています。