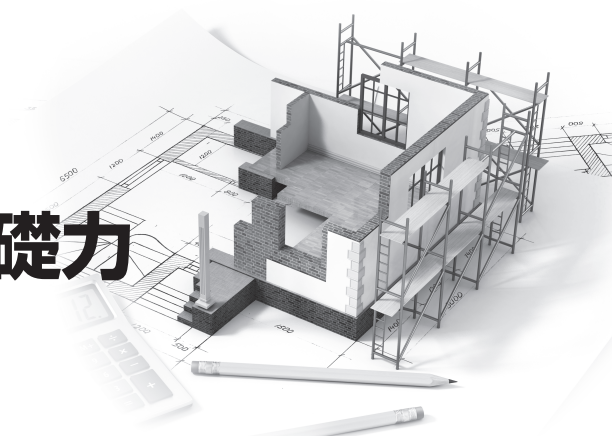


プロになるなら
身につけたい

プログラマの コーディング基礎力



CONTENTS

序	章	良いコードを書くために	1
---	---	-------------	---

第 1	章	今すぐ実践できる良いプログラムの書き方 [C、Java、C#、Ruby、JavaScript]	5
-----	---	--	---

1-1	[C 言語編] enum、配列、浮動小数点を駆使して差をつけよう 「より良いプログラム書きのヒント」	6
1-2	[Java 編] 良いコーディングのさいしょの一步	12
1-3	[C# 編] 言語機能の進化から学ぶ「良いコードの書き方」	21
1-4	[Ruby 編] お作法を意識して可読性や保守性を高めよう	32
1-5	[JavaScript + HTML + CSS 編] 再考！今どきの Web アプリ開発のベストプラクティス	42

第 2	章	今ふたたびの Java	51
-----	---	-------------	----

2-1	Java 5/6/7 の機能にみるリファクタリングの要点	52
2-2	業務アプリケーションにも使える Java 8 のラムダ式	65
2-3	Eclipse だけじゃない！今どきの統合開発環境	76
2-4	トラブル時に頼りになる JDK の解析ツール	85

第 3 章**手を動かして学ぼう 正規表現入門
～プログラミング／エディタ作業の効率大幅アップ****95**

3-0	正規表現は絶対にマスターしておけ！	96
3-1	さまざまな形式の電話番号を検索しよう	99
3-2	カタカナ語の揺れを許容して検索しよう	106
3-3	HTMLをCSVに置換しよう	109
3-4	スペースや改行などの空白文字を検索／置換しよう	115
3-5	正規表現のパフォーマンスを意識しよう	119

第 4 章**プログラマが知っておくべきTCP/IP
～C | JavaScript | PHP | Python | Rubyでコーディング、
Wiresharkでパケットキャプチャ****123**

4-0	プログラマがTCP/IPを知っておくべき理由	124
4-1	TCP/IPって何？～ネットワークの基礎の基礎	129
4-2	実践ネットワークプログラミング～C言語編～	141
4-3	実践ネットワークプログラミング～スクリプト言語編～	152
4-4	パケットキャプチャで学ぶTCP/IP	165

第 5 章**良いPHP、悪いPHP～すぐ効くWeb開発入門****173**

5-1	PHPのはじめ方と学び方～環境構築からコーディングまで	174
5-2	PHPのライブラリの選び方・使い方	182
5-3	PHPフレームワークの選び方～システムの目的から振り返る	190
5-4	PHPのユーザコミュニティ	196

Appendix**198**

「良いプログラム」のための「良いコメント」
～コードを読みやすくするための6つの書き方

《本書について》

本書は、月刊誌「Software Design」の人気記事のなかから、より良いコーディングをするための考え方や手法、基礎技術に関連する特集記事を必要に応じて更新、再編集した書籍です。

仕事としてコードを書くようになると必要になってくる考え方が、各著者の経験を元に紹介されています。ある程度プログラミングができるようになったら、プログラムの品質向上や保守に意識を向けることが、プロフェッショナルへの次の一歩です。

《本書のサポートサイト》

<http://gihyo.jp/book/2017/978-4-7741-8779-2>

《初出一覧》

序章	良いコードを書くために	(書き下ろし)
第1章	今すぐ実践できる良いプログラムの書き方	(Software Design 2016年4月号 第1特集)
第2章	今ふたたびのJava	(Software Design 2014年10月号 第1特集)
第3章	手を動かして学ぼう正規表現入門	(Software Design 2016年7月号 第2特集)
第4章	プログラマが知っておくべきTCP/IP	(Software Design 2016年7月号 第1特集)
第5章	良いPHP、悪いPHP	(Software Design 2016年9月号 第2特集)
Appendix	「良いプログラム」のための「良いコメント」	(Software Design 2016年9月号 一般記事)

《免責》

●本書をお読みにする前に

- ・本書に記載された内容は、情報の提供のみを目的としています。したがって、本書を用いた開発、製作、運用は、必ずお客様自身の責任と判断によって行ってください。これらの情報による開発、製作、運用の結果について、技術評論社および著者はいかなる責任も負いません。
- ・本書記載の情報は各記事の執筆時(再編集時の修正も含む)のもので、ご利用時には変更されている場合もあります。
- ・また、ソフトウェアに関する記述は、各記事に記載されているバージョンをもとにしています。ソフトウェアはバージョンアップされる場合があり、本書での説明とは機能内容や画面図などが異なってしまうこともあり得ます。本書ご購入の前に、必ずバージョン番号をご確認ください。

以上の注意事項をご承諾いただいたうえで、本書をご利用願います。これらの注意事項をお読みいただかずにお問い合わせいただいても、技術評論社および著者は対処しかねます。あらかじめ、ご承知おきください。

●商標、登録商標について

本書に登場する製品名などは、一般に各社の商標または登録商標です。なお、本書中に™、©、®などのマークは記載していません。

良いコードを書くために

Author 増田 亨(ますだ とおる) ギルドワークス(株)
URL <http://guildworks.jp/>

コードには良いコードと悪いコードがあります。良いコードとは何か。どうすれば良いコードを書けるようになるか。そのための考え方とやり方を紹介します。

三つの正しさを目指す

コードの正しさ

良いコードとは、次図の三つの正しさを満たすコードです。

正しく動く

正しく語る

正しく育つ

正しく動く

あたりまえですがプログラムは正しく動かなければいけません。プログラミング初心者は「正しく動く」を単純に考えがちです。良いコードとは、**正しく動く**ことを、さまざまな視点から考慮できているコードです。

正しく語る

コードは第一級的设计文書です。プログラムの意図をわかりやすく伝える手段です。良いコードとは、**読みやすくわかりやすい**コードです。

正しく育つ

正しく動き、正しく語るコードでも、修正や拡張がたいへんなコードがあります。どこに何を書いてあるかを見つけるのにあちこちを調べたり、変更したときに予想外の副作用が起きるコードです。

変更がやりやすいプログラムと、変更がやっかいで危険なプログラムとでは、コードの価値がまった

く異なります。良いコードとは、**修正や拡張がやりやすい**コードです。

正しく動くコード

プログラムを実行したときに問題が起きやすいのは次の場合です。

- 想定外のデータ
- 任意のデータ
- 上書き

これらの場合をきちんと考慮するのが正しく動く良いコードです。

想定外のデータ

プログラミング言語が扱うデータ範囲は制限があります。Javaのint型であれば、マイナス21億からプラス21億までの整数。String型であれば、21億文字まで。

しかしアプリケーションで扱う適切なデータの範囲はもっとずっと狭い範囲です。数量であれば1以上1000以内とか、予約の日付であれば、本日以降で3ヵ月以内などです。

正しく動くコードを書く第一歩は、扱うべきデータの範囲を適切に想定することです。そして、正しい範囲では正しく動作し、範囲外のデータでは、エラーメッセージを出力するなど、適切なエラー処理を行うことが必要です。

正しく動くことを確認するには、範囲外のデータを必ずテストしなければいけません。この範囲外のデータのテストを半ば無意識にできるようになれば初心者は卒業です。

正しく動くコードとは、範囲外のデータを適切に扱っているコードです。