

いちやさしい Python の裏側

2017 Oct 5 / BPStyle #86
Takanori Suzuki

いちばんやさしいPythonの教本

- 272ページ / 2,376円(税込)
- 著者: 鈴木 たかのり、杉谷 弥月
- 2017年8月10日発売
- <https://book.impress.co.jp/books/1116101151>

いちばんやさしいPythonの教本



目次

1. Pythonを学ぶ準備をしよう
2. コマンドプロンプトに慣れよう
3. 基礎を学びながらプログラムを作成しよう
4. 繰り返しと条件分岐を学ぼう
5. 辞書とファイルの扱いを学ぼう
6. 会話botを作ろう
7. ライブラリを使いこなそう
8. サードパーティ製パッケージを使いこなそう
9. Webアプリケーションを作成しよう
10. さらに知識を身に付けるための学び方

目次

1. インストール
2. コマンドプロンプト、ターミナル
3. データ型、演算
4. for、if
5. 辞書、ファイルIO
6. in演算子、関数、組み込み関数
7. モジュール、datetime、スライス
8. pip、venv、Requests
9. Bottle
10. 書籍、コミュニティ

いちばんやさしい教本

- <https://book.impress.co.jp/category/series/easybook/>
- 「絶対に挫折しない」をコンセプトとした実用書
- 第一線で活躍する講師のセミナーを受けているように学べる



いちばんやさしいデジタルマーケティングの教本 人気講師が教えるメディアと販促の新しい基礎

ISBN : 9784295002307
発売日 : 2017/9/15
¥1,980+税



いちばんやさしいJimdoの教本 人気講師が教える本格ホームページの作り方

ISBN : 9784295002260
発売日 : 2017/8/25
¥2,200+税



いちばんやさしいPythonの教本 人気講師が教える基礎からサーバサイド開発まで

ISBN : 9784295002086
発売日 : 2017/8/10
¥2,200+税



いちばんやさしいPHPの教本 第2版 人気講師が教える実践Webプログラミング

ISBN : 9784295001249
発売日 : 2017/5/25
¥1,980+税



ISBN : 9784295002277
発売日 : 2017/11/15
¥1,980+税



ISBN : 9784295002093
発売日 : 2017/11/15
¥1,980+税



ISBN : 9784295001256
発売日 : 2017/11/15
¥1,980+税



ISBN : 9784295001256
発売日 : 2017/11/15
¥1,980+税

はなすこと

- チーム
- スケジュール
- 執筆の進め方
- 執筆ツール
- 感想
- おまけ

チーム

チーム

- takanory: 著者リーダー(1、6～10章)
- omega: 著者(2～5章)
- shimizukawa: 相談役
- haru: レビュワー
- kameko: レビュワー
- hirokiky: コラミスト
- 大津さん: 編集プロダクション(リブロワークス)
- 柳沼さん: インプレス

スケジュール

スケジュール(当初)

- 12月5日: インプレス初回打ち合わせ
- 12月16日: インプレス2回目打ち合わせ→Go
- 12月末: 構成まとめ
- 1月: 執筆開始
 - 2月8日: キックオフ飲み
- 2月27日: 写真撮影
- 4月末: 初校完成
- 5月後半: 再校の確認
- 5月末: 印刷所入稿
- 6月: 刊行

スケジュール(実際)

- 4月末: 7章まで初校完成(omega離脱)
- 5月末: 8章(pip)初校。4章を4、5章に分割
- 6月5日: 1、3～8章初校PDF
- 6月6日: 10章、コラム初校
- 6月9日: 9章(bottle)初校
- 6月18日: 5章(ファイルIO)を再構成
- 6月22日: 2章(コマンドライン)を追加
 - 初校ほぼ完成(2ヶ月遅れ)

スケジュール(実際)

- 6月29日: 再校PDF確認
- 7月10日: はじめにを追加
- 7月19日: 四校PDF確認
- 7月24日: 印刷所入稿
- 8月10日: 刊行
- 8月16日: 打ち上げ

ざっくり2ヶ月遅れた

- スラスラわかるPythonも、遅れたっぽい
- 結果的にほぼ同時発売



理由

- shimizukawaは執筆入れなそう
- omega途中で離脱
- 僕はまだ本気出してないだけ

対策

- PDFのレビューはharuにほぼまかせた
 - 残件と指摘対応に集中(takanory)
- コラムをhirokikyに書いてもらった
 - テキスト量の調整のみ実施(takanory)

執筆の進め方

構成まとめ

- 教えていく順番をまとめる
- コード例をまとめる
 - 前に教えてない要素を使わない
- 説明するテキストを書いていく

執筆のすすめ方

- ざっくり書く
- →週に2回、30分のレビュー会
- →直す

レビュー会

- takanory, shimizukawa, omega
 - ぶっちゃけomegaの執筆がメイン
 - あとから本気出す(takanory)
- 観点
 - ストーリーはどうか
 - 説明の過不足
 - いいまわしはどうか

原稿フォーマット

- Markdown + 独自記法
- mdpreview
 - <https://github.com/lwohtsu/mdpreview>
 - Electron製のプレビューアプリ
 - 大津さん(リブロワークス)製

「セクションヘッダー」————【サブタイ】 [数値計算]
コンピューターに計算をさせてみよう
前のレッスンでは、プログラムを作るときの考え方を学びました。次はいよいよPyt
「セクションヘッダー」

コンピューターで計算するには？

【二段】

1 + 2 - 7

- 数値 (値)

- 演算子

【/構文】

★★編集コメント：「値」とは何かという説明を軽くここでしておくかないと後で苦し

-->

-->

1+2-7

HTMLプレビュー（高速） 改ページプレビュー Vivliostyleプレビュー

/Users/takanori/Project/ichiyasa-python/pages/Chapter3.md

Lesson 00 [数値計算] コンピューターに計算をさせてみよう

 このレッスンのポイント

前のレッスンでは、プログラムを作るときの考え方を学びました。次はいよいよPythonを使ってプログラムを書いていきます。このレッスンで、簡単な例題をもとにコンピューターの得意なことの1つである「数値計算」をする方法を学んでいきましょう。

① コンピューターで計算するには？

プログラムで計算をするには、「式」を書きます。
式は数値や演算子を組み合わせたものです。以下の例は、一番簡単な計算式です。算数で習う数式とそ
う変わりません。また、ここに登場するような数値やデータ（文字列や論理値等）、式の計算結果を「値」といいます。

▶計算する式の例

1 + 2 - 7

数値（値）演算子

プログラムは基本的に半角英数字で入力しましょう。全角で入力してしまうと、プログラムが動かなくなったり、予期しない結果になる可能性があります。



② 算数とはちょっと違う！計算記号の「演算子」

演算子とは、計算するときの記号のことです。足し算のプラス記号(+)、引き算のマイナス記号(-)などが演算子にあたります。プログラムの世界では、算数の計算方法のように線や図を書いて計算はできませんから、すべての計算方法を演算子で指定します。

例えば、割った余りを出したいときは、「%」という演算子を使います。「17 % 4」の計算結果は、余りの数である「1」となります。余りの数以外にも、掛け算の演算子は「×」ではなく「*」と書かなければいけないなどの独特的のルールがあります。

▶Pythonの主な演算子と実行例

| 演算子 | 演算子の意味 | 計算の例 |
|--------|---------------------|---------------|
| a + b | 足し算 | 1 + 2 → 3 |
| a - b | 引き算 | 3 - 1 → 2 |
| a * b | 掛け算 | 2 * 7 → 14 |
| a / b | 割り算 | 17 / 4 → 4.25 |
| a // b | 割り算の結果から小数点以下を切り捨てる | 17 // 4 → 4 |
| a % b | aをbで割った余り | 17 % 4 → 1 |
| a ** b | aのb乗 | 2 ** 3 → 8 |

③ 値と演算子の間は半角スペースを入れる

プログラムの内容は、一般に「書く時間」より「読まれる時間」のほうが多いといわれます。Pythonでは、PEP8と呼ばれる可読性の高いプログラムの書き方についてのまとまった指標があります。例えば、計算式を書くときには数値と演算子の間に半角スペースを1つ置くというルールがあります。半角スペースを置くことで、数式が読みやすくなるのです。もし、数値と演算子が密着した長い数式があったら、とても読みづらいプログラムになってしまいます。こういった書き方は、プログラムのミスにもつながります。ルールにしたがった書き方を心がけましょう。

▶可読性の悪い式 可読性のよい式

1+2-7

1 + 2 - 7

これ、どちらもPEP8違反にならないんだよなあ...(たかのり)

Lesson 13

[数値の計算]

コンピューターに 計算をさせてみましょう



このレッスンの
ポイント

前のLessonでは、プログラムを作るときの考え方を学びました。次はいよいよPythonを使ってプログラムを書いていきます。このLessonで、簡単な例題をもとにコンピューターの得意なことの1つである「数値計算」をする方法を学んでいきましょう。

→ プログラムで計算をするには？

プログラムで計算をするには、「式」を書きます。式は数値や演算子を組み合わせたものです。以下の例は、一番簡単な計算式です。算数で習う数式と

そう変わりはありません。また、ここに登場するような数値やデータなどの文字列や真偽値(74ページを参照)、式の計算結果のことを「値」といいます。

▶ 計算する式の例



プログラムは基本的には半角英数字で入力しましょう。全角で入力してしまうと、プログラムが動かなかったり、予期しない結果になる可能性があります。



→ 算数とはちょっと違う！ 計算記号の「演算子」

演算子とは、計算するときの記号のことです。足し算のプラス記号「+」、引き算のマイナス記号「-」などが演算子にあたります。プログラムの世界では、算数の計算方法のように線や図を書いて計算はできませんから、すべての計算方法を演算子で指定し

ます。例えば、割った余りを出したいときは、「%」という演算子を使います。「 $17 \% 4$ 」の計算結果は、余りの数である「1」となります。余りの数以外にも、掛け算の演算子は「×」ではなく「*」と書くなどの独特的ルールがあります。

▶ Pythonの主な演算子と実行例

| 演算子 | 演算子の意味 | 計算の例 |
|----------|---------------------|---------------------------|
| $a + b$ | 足し算 | $1 + 2 \rightarrow 3$ |
| $a - b$ | 引き算 | $3 - 1 \rightarrow 2$ |
| $a * b$ | 掛け算 | $2 * 7 \rightarrow 14$ |
| a / b | 割り算 | $17 / 4 \rightarrow 4.25$ |
| $a // b$ | 割り算の結果から小数点以下を切り捨てる | $17 // 4 \rightarrow 4$ |
| $a \% b$ | a を b で割った余り | $17 \% 4 \rightarrow 1$ |
| $a ** b$ | a の b 乗 | $2 ** 3 \rightarrow 8$ |

→ ワンポイント プログラムの適切な位置に半角スペースを入れる

プログラムの内容は、一般に「書く時間」より「読まれる時間」のほうが多いといわれます。Pythonでは、PEP8(ペップエイト)と呼ばれる可読性の高いプログラムの書き方についてのまとまった規約があります。変数の代入(Lesson 15参照)の等号(=)の前後には半角スペースを

1つ置くというルールがあります。半角スペースを入れることでプログラムを読みやすくしています。スペースの有無がルールと異なると、複数人で開発する場合、不統一で読みにくいプログラムとなってしまいます。ルールにしたがった書き方を心がけましょう。

▶ 読みにくい式

`i = i + 1`

▶ 読みやすい式

`i = i + 1`

▶ PEP 8 -- Style Guide for Python Code(公式)

<https://www.python.org/dev/peps/pep-0008/>

日本語訳

<https://pep8-jp.readthedocs.io/ja/latest/>

だいたいのレイアウトがわかる
めちゃくちゃ助かる

執筆ツール

執筆ツール

- GitHub
- エディタ
- mdpreview
- Google Drive
- Windows PC
- Dropbox

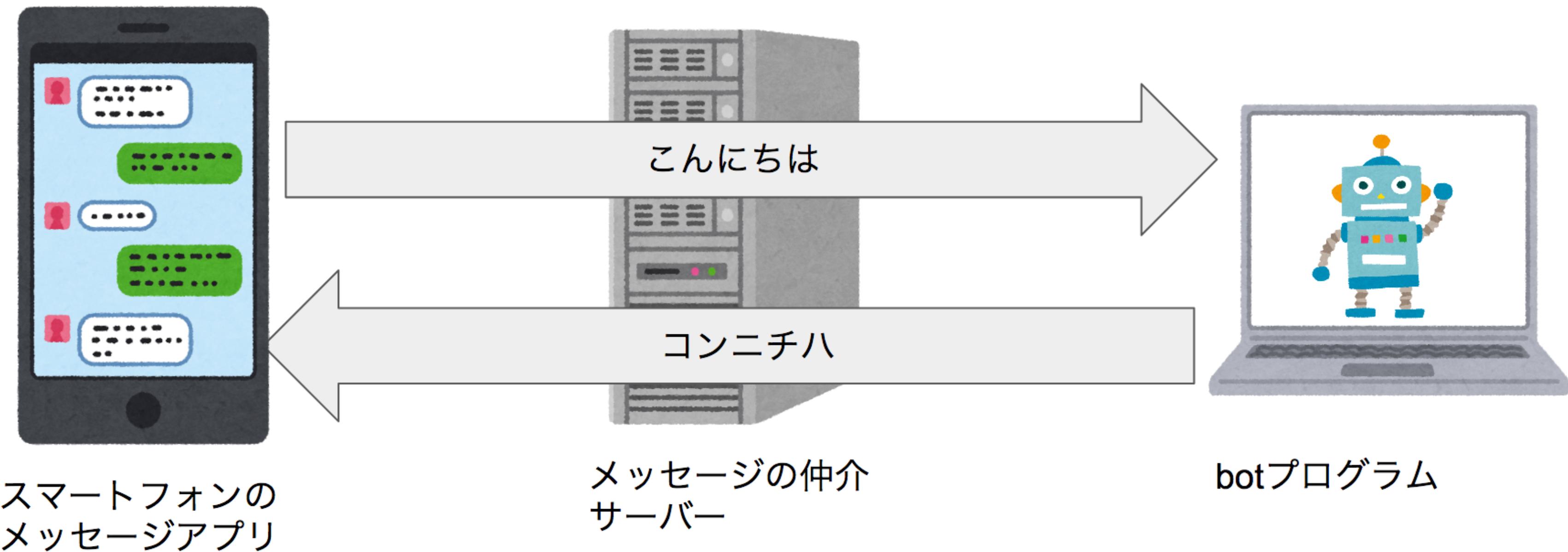
GitHub

- <https://github.com/beproud/ichiyasa-python/>
- 章ごとにissue, branch, pull request
 - pull requestにコメント

| <input type="checkbox"/> | 0 Open ✓ 28 Closed | Author ▾ | Labels ▾ | Projects ▾ | Milestones ▾ | Reviews ▾ | Assignee ▾ | Sort ▾ |
|--------------------------|---|----------|----------|------------|--------------|-----------|------------|------------------|
| <input type="checkbox"/> | sampleコードの作成 ✘ #6 by OMEGA001 was merged on Jan 17 • Changes requested | | | | | | | 92 |
| <input type="checkbox"/> | Chapter2-3の日本語ありバージョンを書く ✘ #8 by OMEGA001 was merged on Mar 10 | | | | | | | 190 |
| <input type="checkbox"/> | [WIP] #12 chapter1 を書く ✘ #13 by takanory was merged on Mar 10 | | | | | | | |
| <input type="checkbox"/> | #10 chapter 4 のbotのコード ✘ #15 by takanory was merged on Feb 23 | | | | | | | 16 |

Google Drive

- Googleスライドを図の作成に使用(takanory)
- 図の中のテキストが抜き出せる
- いらすとや++



Windows PC

- インフラPCを利用
- Windowsのキャプチャに使用
- [Screenpresso](#)でキャプチャ
- ファイルサーバー経由でMacとファイル共有
- キーボードが狭くて、とてもだるかった

Dropbox

- 世界最強のPDFレビューツール
- これなしで書籍のレビューはありえない

おもちゃってなんなのにやん! 15:01

+ ☺ ▶

◀ ○ □

やりとりする形
返信...

HS やりとりする形

リ (API)

スマートフォンのメッセージアプリとbotプログラムが会話をするには、通常は上記の図のようにサーバーを介してメッセージのやりとりをします。

この書籍ではプログラムを単純にするために、サーバーを介さずに直接人とbotプログラムが

やりとりを行う形で話を進めますが、実際にbotプログラムでスマートフォンをやりとりする場合には、各メッセージアプリ（LINE、Facebook Messengerなど）が公開しているやりとり方法（APIといいます）にしたがってプログラムを作成します。

49 やりとりする形で話を進めますが、50 実際にbotプログラムでスマートフォンをやりとりする場合には、各メッセージアプリ（LINE、Facebook Messengerなど）が41 公開しているやりとり方法（APIといいます）にしたがってプログラムを作成します。

- きる命令)
- HS コメント箇所を表示
- HS やりとりする形
- 50 コメント箇所を表示
- HS "実際にbotプログラムでスマートフォンをやりとりする"とは？
- 「実際にbotプログラムがスマートフォンとやりとりするには、」

感想

takanory

- 初心者向けの本にはチャレンジしてみたかったので、やれてよかったです
 - 研修でも役に立っている感じがある
- 出張での研修講師と作業のピークがカズってだいぶキツかった

takanory

- 初心者向けの本にはチャレンジしてみたかったので、やれてよかったです
 - 研修でも役に立っている感じがある
- 出張での研修講師と作業のピークがカブってだいぶキツかった
- Twitterにアレを流したtell-kは許さない

omega

- コードを書くより心身ともに疲れましたが、色々勉強になりました
- 他のPython本と一緒に並んでて嬉しい、この本に適する人に選んでもらえたら良いなあと思います

shimizukawa

- 見開きで情報を納めるのが重要な本だったので、オンデマンドビューアーは超便利だった。あれがなかったらこの本の執筆はもっと大変だったと思う
- 「この書き方でプログラミング初心者が分かってくれるかなあ」の塩梅が難しかった。端的に説明するよりイメージを膨らませる説明が重要な本は書くのが大変そう
- たかのりさん大変そうだった。彼はコンテンツになったのだ

haru

- 編集者の大津さんが頑張っていたので、自分もレビュー頑張ろうと思えた
- 最後の品質を上げるところを、ぎりぎりまで頑張って読みやすい書籍になったとおもう。PyPro3もこれくらい読みやすい書籍になれば良さそう
- DropBoxの機能でPDF上で直接指摘を出来たのは楽だった
- hirokikyの緊急コラミストをやってくれて終盤でボリュームと品質がガッとあがった
- takanory、最後の追い込み、産みの苦しみおつかれさまでした

kameko

- 教えにくい環境設定、ターミナルなどをわかりやすく書いてあり、プログラミング初心者への説明を行っている立場から参考になりました
- レビューに参加させていただいてありがとうございました

hirokiy

- takanoryさんがすごく頑張っていた
- いっちょ噛みできて楽しかった

大津さん

- Pythonの特性に合わせたリスト→for in→if→条件式→whileという構成がすばらしかったです
- slackでのやりとりのスピード感が勉強になりました
- 終盤のチェックで多くの方に協力していただけて助かりました

おまけ

あのイラスト

- 「人気講師が教える」体→顔出し必須
- BPオフィス(bar)で撮影
- プロのカメラマン
- 一人30分
- いろんなポーズ
- 写真をそのままイラストにするこだわりの職人



モデルさんってすごい



Takanori Suzuki

@takanory



素材として強く生きていこう

♡ 8 5:19 - 2017年10月2日



[Takanori Suzukiさんの他のツイートを見る](#)



ありがとうございました