

みんなのPython勉強会 #53

1/16, 2020

Python本の著者が Pythonの本を読み漁ってみた

辻 真吾 (@tsjshg)

Start Python Club

おまえ、誰よ？

❖ 辻真吾（つじしんご）

❖ 東京大学先端科学技術研究センターに所属

❖ コンピュータと数学が好き（数学は下手の横好き）

❖ バイオインフォマティクスから応用データサイエンスへ

❖ 単著に「Pythonスタートブック増補改訂版」（技術評論社）、
「Pythonで学ぶアルゴリズムとデータ構造」（講談社）、共著
に「Python3スキルアップ教科書」（技術評論社）など

宣伝：医療業界がデータ解析できる人を求めています

- ❖ 2020年3月27～29日に開催される国際的なワークショップ
- ❖ 東京医科歯科大学（東京・お茶の水）で開催される
- ❖ PythonやRを使って実際の医療現場で得られるデータ解析を体験できる
- ❖ 非医療関係者の参加歓迎です



3rd
Big Data Machine Learning in Healthcare
in Japan@TMDU

TMDU-MIT-NUS-ANZICS-JSICM Critical Data Workshops and Datathon 2020

- HOME
- WELCOME MESSAGE
- EVENT PROFILE
- PROGRAM
- Steering Committee
Speakers & Facilitators
- REGISTRATION
- HOTEL
ACCOMMODATIONS
- SPONSORSHIP
- Photo Album



March 27-29, 2020 / Tokyo Medical and Dental University (TMDU)
Co-organized by : MIT Critical Data / NUS / ANZICS / JSICM / Datathon-Japan
Chair: Hidenobu Shigemitsu
Institute of Global Affairs / Department of Intensive Care Medicine, TMDU

もくじ

- ❖ Pythonの入門書と関連書籍、合計8冊の紹介
- ❖ プログラミング入門書3冊
- ❖ 変数は箱か？
- ❖ プログラミングを何かと一緒に学ぶ3冊
- ❖ コンピュータショナル・シンキングとは？

ことの経緯

- ❖ 前回のStapy (2019/12/11) の打ち上げで次回はPythonの本の話など、どうやって勉強するかという話題に決定
- ❖ 昨年末、大きめの本屋さん (MARUZEN & ジュンク堂書店 渋谷店) へ
- ❖ 知ってはいたが、Pythonの本がありすぎる・・・
- ❖ 自分はもちろん、会ったことがある人の本を外そう！
- ❖ かなり候補が絞られ、良さそうな本を自費で購入



- Python によるプログラミング入門**
森畑明昌 (Akira Morihata) 著
東京大学出版会
- プログラミング入門講座**
The Best Way to Learn Computer Programming.
米田昌悟 著
- Head First Python 第2版**
頭でかたで覚える Pythonの基礎
Paul Barry 著
鳥田健哉 監訳
オト哲世 訳
O'REILLY
- Learn PYTHON 3 the HARD WAY**
書くと覚える Python入門
ZED SHAW 著
堂阪真司 訳
丸
- コンピュータシミュレーション**
磯辺秀司・小泉英介・静谷啓樹・早川美徳 著
高橋麻奈 (Mana Takahashi) 著
シリーズ 100万部突破!!
SB Creative
- やさしい Python**
Machine or human? Be the winner.
Whatever the future will be.
黒川雄英 著
Sriji Devadas 著
O'REILLY
- 問題解決のPythonプログラミング**
数学パズルと解けるPythonのプログラミング
柴田望洋 著
SB Creative
- 新・明解 Python入門**
柴田望洋 著
SB Creative

プログラミング入門書を選ぶ指針

❖ 実践/リファレンス

- ❖ とにかくやってみて、最低限できるようにすることを目指す実践タイプ

- ❖ 言語に関する知識を余すところなく網羅するリファレンスタイプ

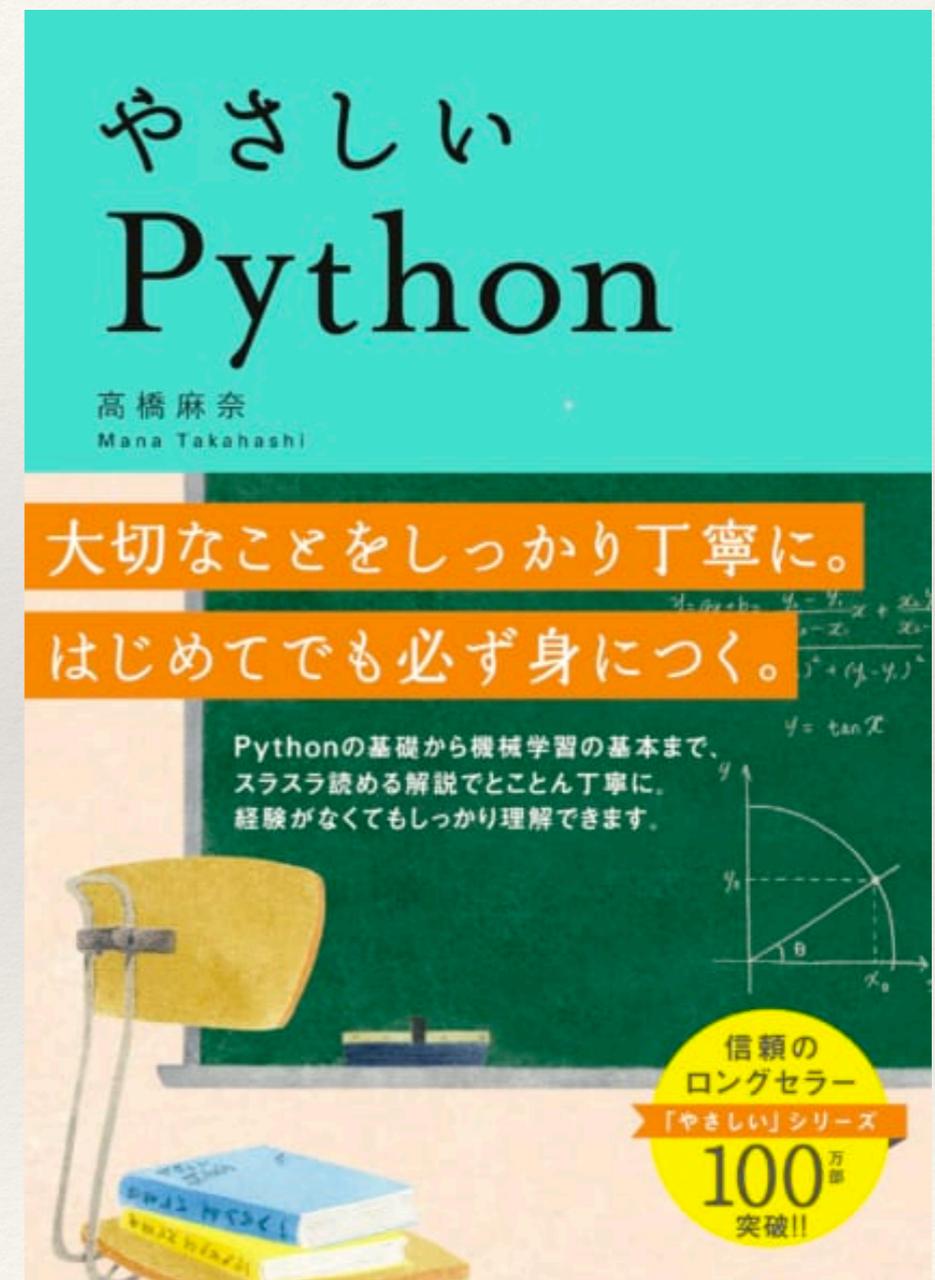
❖ 言語だけ学ぶ/何か一緒に学ぶ

- ❖ 純粹にプログラミング言語の説明に特化する言語だけタイプ

- ❖ Webアプリやアルゴリズムなど、他の項目を学びながらプログラミングに関する知識を解説する何か一緒にタイプ

- ❖ 「やさしいC」、「やさしいJava」などで有名な著者
- ❖ まったくクセがない解説
- ❖ 内包表記や関数デコレータ、ジェネレータなど入門書ではあまり見ない内容も入っている
- ❖ SQLを含むデータベースの話と機械学習の基本まで解説
- ❖ 記述量は少ないので、応用の1つの例といった位置付け

実践度とリファレンス度のバランス
言語だけではなく、後半では機械学習なども少し



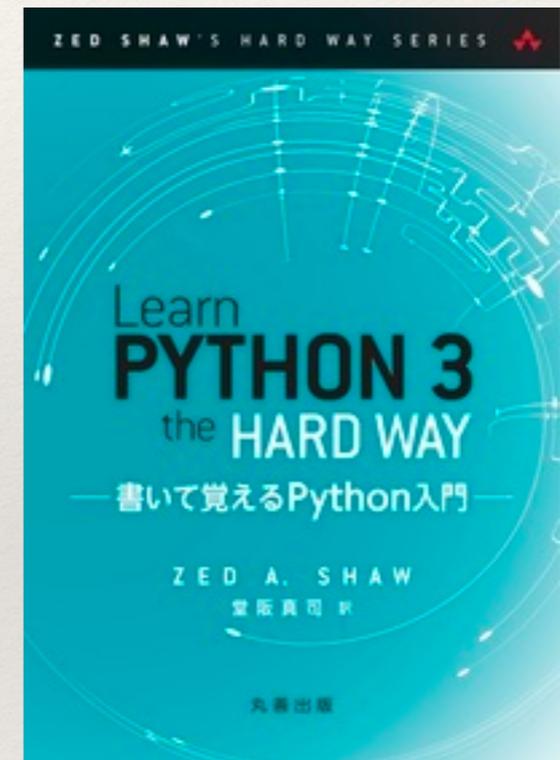
- ❖ プログラミング言語教育界の巨匠（本の帯より）CやC++、Javaなど多くの解説書を出版
- ❖ 細かいところまでしっかり記述されていて網羅性が高い
- ❖ 「Pythonは、見かけはシンプルですが、奥が深くて複雑なプログラミング言語です。習得は、決して容易ではありません。」本書の構成より引用
- ❖ 他のプログラミング言語との比較でPythonを説明している箇所が多くあり、他言語の経験者にはとくにおすすめ

リファレンス度がかなり高い
言語の解説に特化



- ❖ 書いて覚えることを主眼に置く
 - ❖ コピペ禁止
 - ❖ インターネットで調べる
- ❖ 基本的にはテキストエディタで書いて、その都度pythonコマンドで実行
- ❖ 全体が52のエクササイズに分かれているので、段階的に進められる
- ❖ 普通の人目線からの記述
 - ❖ ifの中のif、関数を呼ぶ関数、最初はわかるわけない

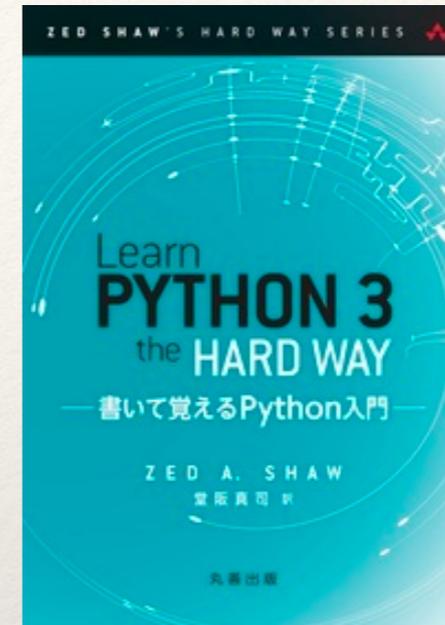
実践度が高い
言語の習得が中心



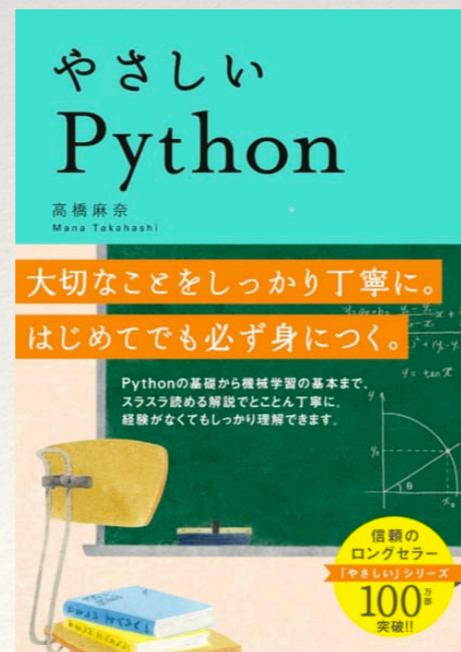
類書は多いので自分にあった方向性を



言語の全体を網羅的に理解



とにかく書いて実践する方向



バランス重視の入門書

変数は箱？

- ❖ プログラミング入門書では、変数を初めて説明するとき「箱」と例え、データの入れ物と説明することが多い
- ❖ 箱という例えは、プログラミングの学習を進めて行くと混乱のもとになる可能性もある（と思う）
- ❖ 賛否両論あるかと思います

1	a = 20
1	a
20	
1	a = b = [1, 2, 3]
2	a
[1, 2, 3]	
1	a[0] = 10
2	b
[10, 2, 3]	

入門書によっても記述はそれぞれ違う



それまで箱のようなものと言っておきながらP114で「**真っ赤な嘘**」として参照先を示す名前と説明

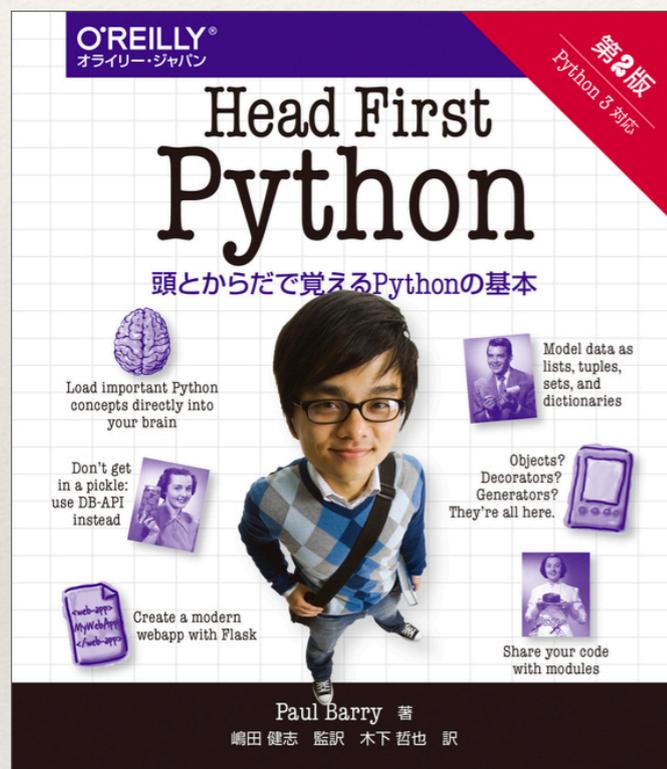


変数が最初にでてくるP23で「名前でしかない」と言い切り、実際のコードをすぐ紹介する



初出のP30で「変数というハコの中に値を入れる」とし、以後ハコのイメージで統一

ここから何か一緒に学ぶ系3冊



Flaskを使った
Webアプリケーション



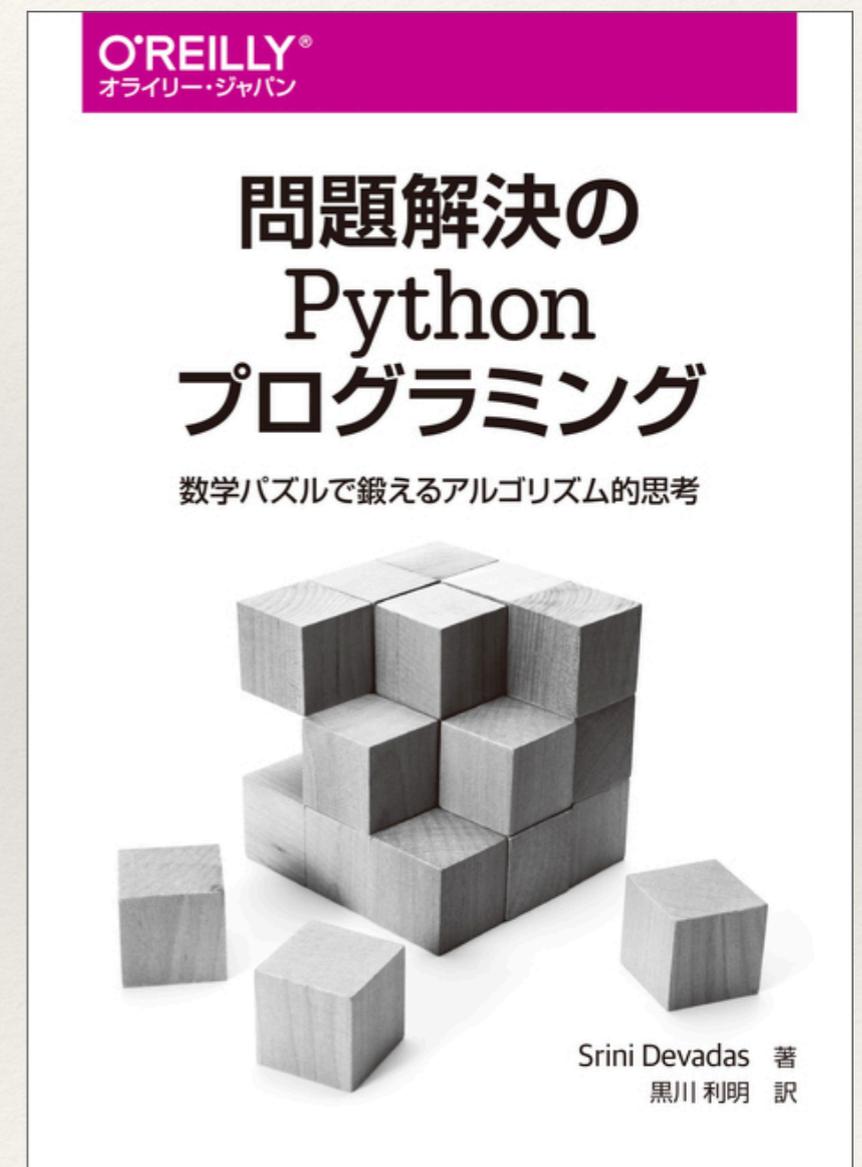
パズルを題材にした
簡単なアルゴリズム



少し高度なアルゴリズムと
微分方程式などの数学

- ❖ タイルの配置やハノイの塔、数独など実際のパズルを解くコードを書くことでプログラミングを学ぶ本
- ❖ 基本文法を知ったあと何か作りたいと思うパズル好きにおすすめ
- ❖ 内容的には結構難しいので、真剣に読むと時間がかかると思われる
 - ❖ わざと深くインデントするコードを見せてそれを改良するなどの工夫もある
- ❖ 関数や変数の名前がスネークケースになっていないなど、文法的には気になるところも

言語の説明はほぼなし
パズルを通じてアルゴリズム
に入門できる



- ❖ 「新しいタイプ」と銘打っているだけあって、類書はほとんどないと思われる
- ❖ 放物線のシミュレーション、拡散方程式、回帰分析からゲノム解析まで高度で多様な応用事例
- ❖ 東京大学教養学部の教科書ということで、周辺知識は講義で教えてもらいながら学ぶ前提か？
- ❖ コラムなど随所に著者の情報科学に対する造詣の深さが反映されている

実践度が高い

言語の解説はそれほどなく、
数理的な応用事例が多数



- ❖ 脳を刺激するをテーマに置いているちょっと変わった入門書
 - ❖ 他言語の経験を前提としている
 - ❖ 600ページ弱のボリューム
- ❖ いたるところに吹き出しや挿絵、矢印を使った注釈など、ページ全体がガヤガヤした印象
- ❖ 読んでいるとこれもありか？（と思えた）
- ❖ 前半は言語の解説、後半はFlaskでWebアプリを作る
 - ❖ 127.0.0.1:8888がループバックアドレスとポートだという解説があったり、周辺知識も手に入る

実践を重要視

言語の解説に加えWebアプリについても学べる



番外編

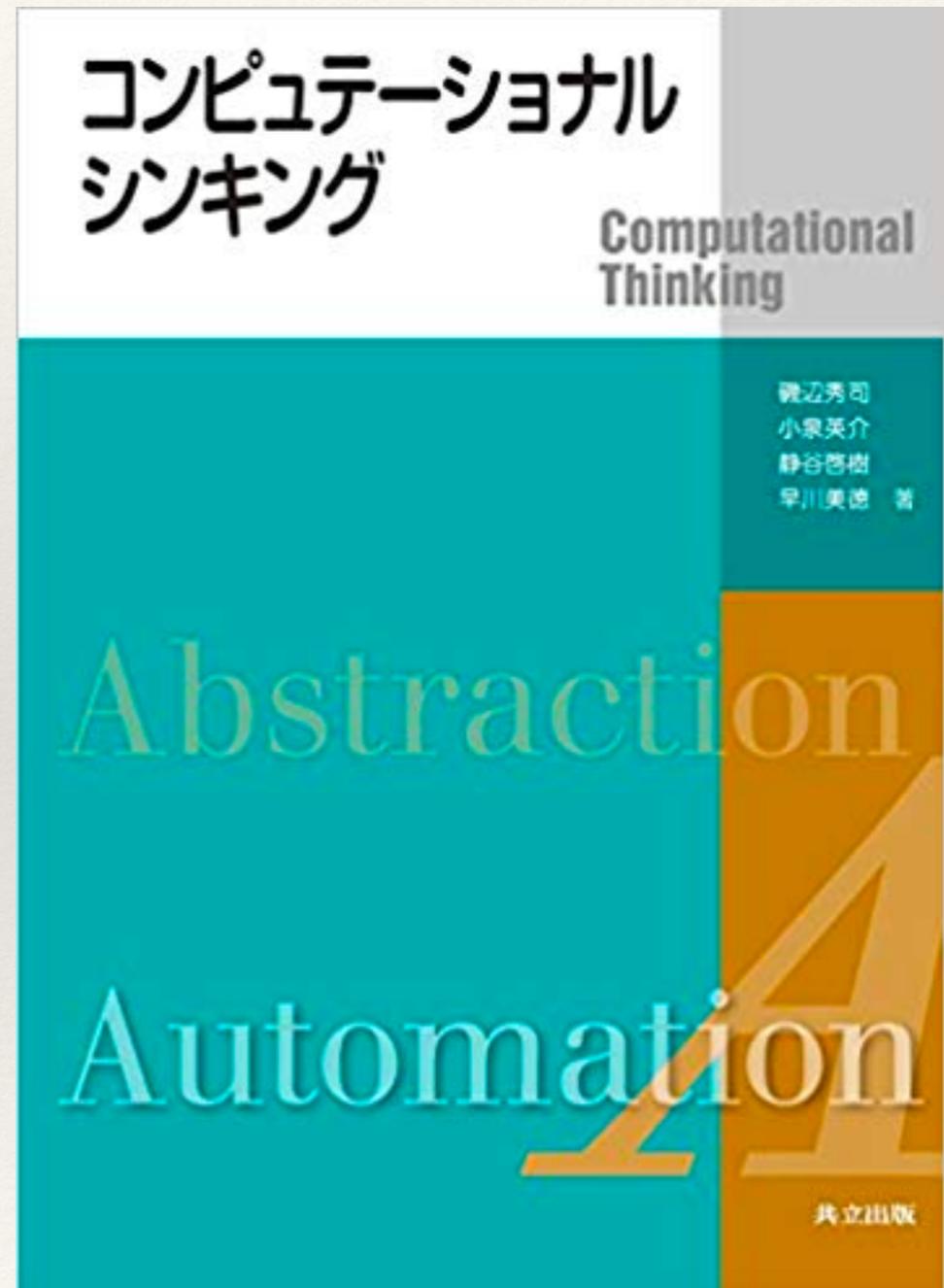
- ❖ バラク・オバマ、スティーブ・ジョブズ、マーク・ザッカーバーグらの「プログラミングを学べ」という引用から始まる一般向け入門書
- ❖ Hour of Code, lightbot, Scratch, CodeMonkeyなど子供向けツールを紹介
- ❖ lightbotは楽しくプログラミングの基本を理解できそう



<https://www.sbcr.jp/product/4797383102/>

Computational Thinking

- ❖ 20世紀中頃アルゴリズム・シンキングと呼ばれていたものが、コンピューテーショナル・シンキングとして再び注目されている
- ❖ 「計算機科学の流儀で考えて問題を解決すること」 (引用)
- ❖ さまざまな問題を解くアルゴリズムを紹介した本



例題 2.11. 次のようなプログラムを記述せよ

- 自然数 n を1つ入力する。
- n を逆から読んだ自然数を入力する。例えば、入力が $n=31522$ ならば、出力は22513である。

本の回答は疑似言語なのでPythonで実装したもの

```
1 def rev_n(n):
2     one = 0
3     ans = 0
4     while n > 0:
5         one = n % 10
6         ans = ans * 10 + one
7         n = (n - one) / 10
8     return int(ans)
```

```
1 rev_n(31522)
```

22513

Pythonならこんなコードも書ける

```
1 def my_rev_n(n):
2     str_n = str(n)
3     rev_str = str_n[::-1]
4     return int(rev_str)
```

```
1 my_rev_n(31522)
```

22513

まとめ

- ❖ 実践/リファレンス、言語だけ/何かと一緒にという切り口で6冊の書籍を紹介
- ❖ 変数が箱かどうかは、著者によって対応が分かれる
- ❖ 興味がある分野と一緒にPythonを学ぶという選択肢もある
- ❖ 計算機科学の流儀でものを考えるということは、利用するプログラミング言語とも関係が深いと思われる

ご清聴ありがとうございました