

# 『みんなの Python』メモ

ぱろすけ

2013 年 4 月 7 日

## Chapter 1.

- `#!/usr/bin/python` より `#!/usr/bin/env python` を使いましょう

## Chapter 2.

- クォートを 3 つ重ねると改行可能な長い文字列を表すことができる
- `ord()` と `chr()`
- `del` でリストの要素を削除できる。 `del a[2]` など。
- タプルは辞書のキーとして利用可能

## Chapter 3.

とくになし。

## Chapter 4.

- 関数名はなるべく小文字だけで付けること。

## Chapter 5.

- `hex()`
- raw 文字列
- `S.rjust()`, `S.ljust()`, `S.center()` で幅を揃えられる
- 数値のフォーマット ex. `%+010.2f`
- 辞書によるフォーマット ex. `"%(key)s" % ("key": "text")`
- set 型の xor は `A^B`
- 辞書では `get()` を使うとデフォルト値が指定できる
- 空のリストや辞書は `False`

- `enumerate()` や `zip()` をきちんと使おう
- 関数の引数に `*` をつけるとタプルで、`**` をつけると辞書で受け取る

## Chapter 6.

- ファイルオブジェクトはイテレータ

## Chapter 7.

- リスト内包表記
- ジェネレータ

## Chapter 8.

- `Decimal` というクラスがある
- カプセル化したい変数にはアンダースコアをひとつかふたつ付ける

## Chapter 9.

- `object` を継承すると新スタイルクラスになる
- `__slots__` に操作可能なアトリビュートを指定できる
- `property()` でセッターとゲッターを利用可能
- `super(class_name, self)` でスーパークラスを取得可能
- 特殊メソッドの項は必要になったら参照する

## Chapter 10.

- `traceback` モジュールを使うと例外の様子を取り出せる

## Chapter 11.

とくになし

## Chapter 12.

- `ininstance()` である変数がある型であるか調べられる

## Chapter 13.

- StringIO: 文字列のファイル風操作
- csv: csv ファイルの操作
- shelve: 辞書をファイルに保存しつつ用いる
- calendar: 便利なかなこれ
- 他いろいろ書いてありますが、細部は知らずとも存在は知っているものばかりだったので良しとします

## Chapter 14.

- 応用。GUI を作りたくなったら読む。

## Chapter 15.

- Python 3 について。必要になったら読みましょうという感じ。