Let's implement useless Python objects

役に立たない Python オブジェクトを作ろう

Hayao Suzuki

October 27, 2023

Share it

GitHub
- https://github.com/HayaoSuzuki/pyconapac2023

Hashtag
- #pyconapac #pyconapac2023 #pyconjp

Who am I ?（お前誰よ）

Name Hayao Suzuki（鈴木　駿）

X(Twitter) @CardinalXaro

Work Software Developer @ BeProud Inc.

- BeProud Inc.
  - connpass
  - PyQ
  - Tracery

Who am I ?

Translated Books

- TBA(O'Reilly Japan) New!

Supervised Translated Books

- Introducing Python 2nd ed.(O'Reilly Japan)
- Robust Python(O'Reilly Japan)

Who am I ?

Selected My Talks

- Symbolic Mathematics using SymPy(PyCon JP 2018)
- How to Use In-Memory Streams(PyCon JP 2020)
- Unknown Evolution of the Built-in Function pow(PyCon JP 2021)

Listed at `https://xaro.hatenablog.jp/` .

And now for something completely different...

Let's implement useless Python objects

What is it mean useless?

From LDOCE
1. not useful or effective in any way
2. (informal) unable or unwilling to do anything properly

Is the useless object really useless?

From Zhuangzi Ren-jian shi（莊子 人間世篇）

人皆知有用之用 而莫知無用之用也

Everyone knows the usefulness of the useful, but no one knows the usefulness of the useless.

Today's Theme

Let's implement useless Python objects

The useless objects are useless, but how to make a useless object is very useful.

Example: `ElasticSized`

```
>>> s = ElasticSized([1, 2, 3, 4, 5])
>>> for _ in range(3):
...     print(len(s), end=" ")
...
1 3 4
```

What is a useless Python object?

Example: ForgottenContainer

```
>>> c = ForgottenContainer([1, 2, 3, 4, 5])
>>> for _ in range(3):
...     print(3 in c, end=" ")
...
False True False
```

Example: `ShuffledIterable`

```
>>> it = ShuffledIterable([1, 2, 3, 4, 5])
>>> for _ in range(3):
...     for v in it:
...         print(v, end=" ")
...     print()
...
5 3 4 2 1
4 1 2 3 5
2 5 3 1 4
```

What is a useless Python object?

Definition of a useless Python object in this talk

A useless Python object behave Pythonic, but does not work as expected.

Pythonic Python objects

How to implement Pythonic Python objects

We need to understand object protocols.

Data Structures and Operations

Basic Data Structures of Python

List `[1, 2, 3, 4, 5]`

Tuple `("pen", "pineapple", "apple", "pen")`

Dictonary `{"Answer": 42}`

Set `{41, 43, 47, 53, 57, 59}`

Common Operations of Data Structure

`len()` Length of object

`in` Membership test

`for` Iteration