

Pythonらしい書き方

文責：@tellusium



Pythonらしい書き方

- ループ
- if文
- 辞書型

ループその1

C

```
int i;  
int array[4] = {98, 37, 48, 85};  
for (i = 0; i < 4; i++) {  
    printf("%d¥n", array[i]);  
}
```

Py

```
array = [98, 37, 48, 85]  
for i in range(len(array)):  
    print(array[i])
```



ループその1

オブジェクトを直接ループできる

```
Py array = [98, 37, 48, 85]  
for obj in array:  
    print(obj)
```

ループその2

インデックス（要素番号）も取得したい時

```
Py array = [98, 37, 48, 85]
   for i, obj in enumerate(array):
       print(i, obj)
```

結果

```
0, 98
1, 37
2, 48
3, 85
```

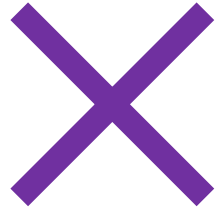
ループその3

配列の要素に操作を加えたい

例：全要素を2倍したい

Py

```
array = [1, 2, 3, 4]
doubled_array = []
for obj in array:
    doubled_array.append(obj * 2)
print doubled_array
```



結果 [2, 4, 6, 8]

ループその3

そんな時はリスト内包表現

例：全要素を2倍したい

Py

```
array = [1, 2, 3, 4]  
doubled_array = ¥  
    [x*2 for x in array]  
print doubled_array
```

結果 [2, 4, 6, 8]

ループその4

リスト内包表現その2

例：奇数の要素を2倍したい

Py

```
array = [1, 2, 3, 4, 5, 6]  
doubled_array = ¥  
    [x*2 for x in array if x%2 != 0]  
print doubled_array
```

結果 [2, 4, 6, 8]

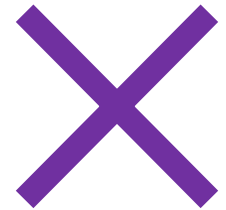
if文

配列にある要素があるかチェックしたい

例："apple"の要素があるか確かめたい

Py

```
array = ["apple", "lemon", "grape"]  
for obj in array:  
    if obj == "apple":  
        print("found!")
```



if文

in 判定を使う

例："apple"の要素があるか確かめたい

Py

```
array = ["apple", "lemon", "grape"]  
if "apple" in array:  
    print("found!")
```

タプル(tuple)

複数の値のセットを表現

$$x, y = 1, 2$$
$$\begin{aligned} x &= 1 \\ y &= 2 \end{aligned}$$

上2つは等価

タプル(tuple)

代入は同時

```
x, y = 1, 2  
x, y = y, x  
print(x, y)
```

結果 2 1

入れ替えもtmp変数なしで可能

辞書型

文字列をkeyに，値を格納する

Py

```
dict = { "apple": 2,  
         "lemon": 3,  
         "grape": 4}  
  
print(dict["apple"])
```

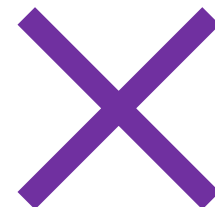
結果 2

辞書型のループ

keyでループ

Py

```
dict = { "apple": 1,  
         "lemon": 2,  
         "grape": 3}  
  
for key in dict:  
    print(key, dict[key])
```



結果

```
grape 3  
apple 1  
lemon 2
```

辞書型のループ

items()を用いてkeyとvalueでループ可能

Py

```
dict = { "apple": 1,  
         "lemon": 2,  
         "grape": 3}  
for key,value in dict.items():  
    print(key, value)
```

結果

```
grape 3  
apple 1  
lemon 2
```

その他

- 参考文献

- <http://kesin.hatenablog.com/entry/2013/05/12/004541>
- <http://nonbiri-tereka.hatenablog.com/entry/2016/01/25/073521>
- <http://qiita.com/stsn/items/ff6535c73f49ba3a01e7>