

# Python 基礎

List - リスト内包表記( List  
Comprehensions ) - 1

# 基本的な書き方：

リスト・コンプリヘンションは、リスト内包表記とも呼ばれます。

これは、ループを利用してイテラブルオブジェクトから一つの要素を取り出し、その要素に対して処理を行って算出してから、リストへ追加する構文です。

新たなリスト = [要素を算出する式 for x in 旧リスト(または、イテラブルオブジェクト)]

# 等価なfor文例：

等価なfor文：

```
新たなリスト = []  
for x in 旧リスト(または、イテラブルオブジェクト):  
    新たなリスト.append(something)
```

Example

：

```
1 list(range(9))  
2 # [0, 1, 2, 3, 4, 5, 6, 7, 8]  
3 [i for i in range(9)]  
4 # [0, 1, 2, 3, 4, 5, 6, 7, 8]
```

## Example -

1～10の平方数を生成して下さい(1)

```
[1x1, 2x2, 3x3, ..., 10x10]
```

ループを使います

```
1 array = []  
2 for x in range(1, 11):  
3     array.append(x * x)  
4  
5 print(array)
```

## Example -

## 1～10の平方数を生成して下さい(2)

リスト・コンプリヘンション

```
1 array = [x * x for x in range(1, 11)]  
2 print(array)
```

## Example - 文字列を文字配列にします

['H', 'e', 'l', 'l', 'o']

```
1 | s = "Hello"  
2 | [c for c in s] # list(s)と同義です
```

Example -

ダブルループですべての組み合わせを生成します

['a1', 'a2', 'a3', 'b1', 'b2', 'b3', 'c1', 'c2',  
'c3']

```
1 | [m + n for m in 'abc' for n in '123']
```