

# 配列を使ったプログラミングをしてみよう

```
(array) PS C:\Users\小西 翔\python\python4> python array_Input.py
何人分のデータを入力しますか？ : 3
--- 1人目 ---
名前を入力してください : 佐藤
年齢を入力してください : 24
--- 2人目 ---
名前を入力してください : 鈴木
年齢を入力してください : 29
--- 3人目 ---
名前を入力してください : 山本
年齢を入力してください : 22

入力されたデータ :
名前 : 佐藤 , 年齢 : 24
名前 : 鈴木 , 年齢 : 29
名前 : 山本 , 年齢 : 22
```

# この教材でできること

- Pythonについて理解できる
- ターミナルを触れる
- 配列について理解できる

## どんな教材？

データの代入を配列を使って行う

# 目次

- ① 配列を使って文字を出力してみよう
- ② for文を使って出力してみよう
- ③ 文字の入力で配列に代入をしてみよう

# 目次

① 配列を使って文字を出力してみよう

② for文を使って出力してみよう

③ 文字の入力や配列に代入をしてみよう

# ① 配列を使って文字を出力してみよう

このページでは配列とは何か？配列を使ってどのようなコードを書いていけばいいかわかるよ

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py
15
3
[1, 2, 3, 4, 5]
[5, 4, 3, 2, 1]
(array) PS C:\Users\小西 翔\python\python4>
```

# ① 配列を使って文字を出力してみよう

配列とは同じような種類のデータを格納するデータ構造

例)

```
weights = [24, 40, 33]
```

体重が24, 40, 33の人が3人いるような例だよ

# ① 配列を使って文字を出力してみよう

実際にコードを書いていって理解を深めていこう

# ① 配列を使って文字を出力してみよう

仮想環境を作ってVscodeを開こう

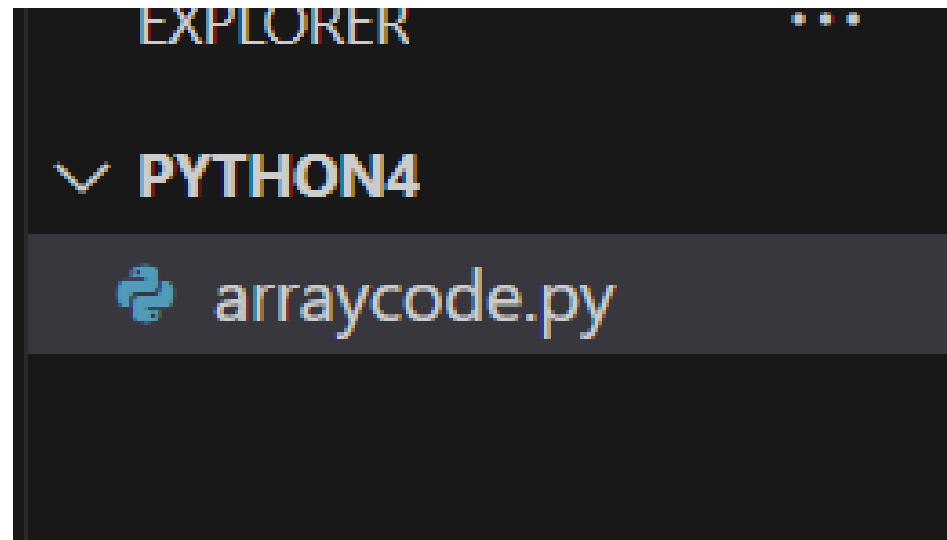
```
# To deactivate an active environment, use
#
#     $ conda deactivate

(base) PS C:\Users\小西 翔\python\python4> conda activate array
(array) PS C:\Users\小西 翔\python\python4> code .
(array) PS C:\Users\小西 翔\python\python4>
```



# ① 配列を使って文字を出力してみよう

Pythonのファイルを作ろう



# ① 配列を使って文字を出力してみよう

numbersという名前の配列を作って3行書いてみよう

```
arraycode.py > ...  
1  numbers = [1, 2, 3, 4, 5]  
2  total = sum(numbers)  
3  print(total)  
4
```

# ① 配列を使って文字を出力してみよう

実行してどのような表示結果が出たか確認してみよう

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py  
15
```

# 質問

出力結果が15になったね！なんで15になったかコードを見ながら考えてみよう

※考えてから次のスライドに進んでね

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py  
15
```

# 答え

total = sum(numbers)が原因だね！

total = 1 + 2 + 3 + 4 + 5

つまりsumは配列の値の合計値を変数に代入するよ

```
arraycode.py > ...  
1  numbers = [1, 2, 3, 4, 5]  
2  total = sum(numbers)  
3  print(total)  
4
```

# ① 配列を使って文字を出力してみよう

次は普通に配列の値を一つ出してみよう！

```
2 total = sum(numbers)
3 print(total)
4
5 print(numbers[2])
6
7
```

# ① 配列を使って文字を出力してみよう

実行してどのような表示結果が出たか確認してみよう

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py  
15  
3  
(array) PS C:\Users\小西 翔\python\python4>
```

# ① 配列を使って文字を出力してみよう

配列は先頭の番号から0からスタートするからnumbers[2]は3番目の数字が表示されるようになるよ

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py
15
3
(array) PS C:\Users\小西 翔\python\python4>
```

```
arraycode.py > ...
1 #           0 1 2 3 4
2 numbers = [1, 2, 3, 4, 5]
3 total = sum(numbers)
4 print(total)
5
6 print(numbers[2])
7
```



# ① 配列を使って文字を出力してみよう

次は配列の中の順番をソートして表示してみよう

```
1 #           0 1 2 3 4
2 numbers = [1, 2, 3, 4, 5]
3 total = sum(numbers)
4 print(total)
5
6 print(numbers[2])
7
8 # リストの要素をソートする
9 numbers.sort()
10 print(numbers)
11
```

# 説明

「ソート」:数字を小さい順に並び替えたり、大きい順に並び替えたりすることを表す言葉

# ① 配列を使って文字を出力してみよう

実行してみたら最後に配列がソートされて表示されるようになったかな？

※今回は元からソートしてたからソートするコードを書かなくても表示結果は同じになってたよ

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py
15
3
[1, 2, 3, 4, 5]
(array) PS C:\Users\小西 翔\python\python4>
```

# ① 配列を使って文字を出力してみよう

最後に逆順にするコードを書いて表示するようにしてみよう

```
9   numbers.sort()
10  print(numbers)
11
12  # リストの要素を逆順にする
13  numbers.reverse()
14  print(numbers)
15
```

# ① 配列を使って文字を出力してみよう

実行してみたら逆順の配列も表示出来たかな？

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py  
15  
3  
[1, 2, 3, 4, 5]  
[5, 4, 3, 2, 1]  
(array) PS C:\Users\小西 翔\python\python4>
```

# 目次

① 配列を使って文字を出力してみよう

② for文を使って出力してみよう

③ 文字の入力や配列に代入をしてみよう

## ②for文を使って出力してみよう

このページではfor文を使って繰り返し処理をしていくよ

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py
15
3
[1, 2, 3, 4, 5]
[5, 4, 3, 2, 1]
8
matrix[0][0]の列:1
matrix[0][1]の列:2
matrix[0][2]の列:3
matrix[1][0]の列:4
matrix[1][1]の列:5
matrix[1][2]の列:6
matrix[2][0]の列:7
matrix[2][1]の列:8
matrix[2][2]の列:9
```

## ②for文を使って出力してみよう

まずは多次元配列を書こう！今回は2次元の配列にしたよ

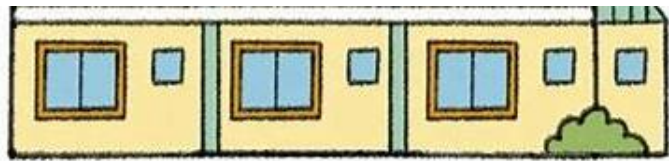
```
13 numbers.reverse()  
14 print(numbers)  
15  
16 matrix = [  
17     [1, 2, 3],  
18     [4, 5, 6],  
19     [7, 8, 9]  
20 ]
```



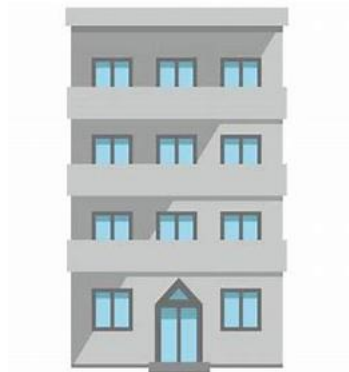
# 説明

「多次元配列」:配列の中に配列を書いていくこと  
(配列のイメージ)

1次元配列  
配列[3]



2次元配列  
配列[4][3]



3次元配列  
配列[3][3][4]



## ②for文を使って出力してみよう

多次元配列の特定の文字を出力したい場合は22行目のように書くよ

```
15
16     matrix = [
17         [1, 2, 3],
18         [4, 5, 6],
19         [7, 8, 9]
20     ]
21
22     print(matrix[2][1])
23
```

# 質問

matrix[2][1]はどの数字が表示されるかな？  
実行してみて確認してみよう  
※考えてから次のスライドに進んでね

```
15
16     matrix = [
17         [1, 2, 3],
18         [4, 5, 6],
19         [7, 8, 9]
20     ]
21
22     print(matrix[2][1])
23
```

# 答え

8が出力されるよ

```
15
16 matrix = [
17     [1, 2, 3], #[0][]
18     [4, 5, 6], #[1][]
19     [7, 8, 9] #[2][]
20 ] #0 1 2
21
22 print(matrix[2][1])
23
```

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py
15
3
[1, 2, 3, 4, 5]
[5, 4, 3, 2, 1]
8
```

## ②for文を使って出力してみよう

for文を使って配列を順番に表示していこう

```
20 ] #0 1 2
21
22 print(matrix[2][1])
23
24 i = 0
25
26 for row in matrix:
27     print(f"matrix[{i}][]の列:{row}")
28     i+=1
```

## ②for文を使って出力してみよう

下に多次元配列のデータが順番に表示されるようになったかな？

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py
15
3
[1, 2, 3, 4, 5]
[5, 4, 3, 2, 1]
matrix[0][]の列:[1, 2, 3]
matrix[1][]の列:[4, 5, 6]
matrix[2][]の列:[7, 8, 9]
(array) PS C:\Users\小西 翔\python\python4>
```

## ②for文を使って出力してみよう

for文で2次元配列を1次元配列にしたらfor文内は1次元配列と同じ書き方でコードを書けるよ

```
24     i = 0
25
26     for row in matrix:
27         print(f"matrix[{i}][0]の列:{row[0]}")
28         print(f"matrix[{i}][1]の列:{row[1]}")
29         print(f"matrix[{i}][2]の列:{row[2]}")
30     i+=1
```

## ②for文を使って出力してみよう

ちなみに実行したらこのようになるよ

```
(array) PS C:\Users\小西 翔\python\python4> python arraycode.py
15
3
[1, 2, 3, 4, 5]
[5, 4, 3, 2, 1]
8
matrix[0][0]の列:1
matrix[0][1]の列:2
matrix[0][2]の列:3
matrix[1][0]の列:4
matrix[1][1]の列:5
matrix[1][2]の列:6
matrix[2][0]の列:7
matrix[2][1]の列:8
matrix[2][2]の列:9
```



# 質問

□のprintが繰り返し同じことを書いているね！もう少し簡単にコードを書き換えてみよう

※考えてから次のスライドに進んでね

ヒント:26行目のfor文の中にもう1つfor文を追加

```
24     i = 0
25
26     for row in matrix:
27         print(f"matrix[{i}][0]の列:{row[0]}")
28         print(f"matrix[{i}][1]の列:{row[1]}")
29         print(f"matrix[{i}][2]の列:{row[2]}")
30         i+=1
```

# 答え

for文の中にfor文を書いてprintの同じようなコードをを書くことを防いでいるよ

今回は合計のコードの行数は増えるけどこの考え方は必ず使うから覚えておいてね

```
23
24     i = 0
25
26     for row in matrix:
27         j = 0
28         for element in row:
29             print(f"matrix[{i}][{j}]の列:{element}")
30             j+=1
31     i+=1
```

# 目次

① 配列を使って文字を出力してみよう

② for文を使って出力してみよう

③ 文字の入力で配列に代入をしてみよう

## ③文字の入力で配列に代入を試してみよう

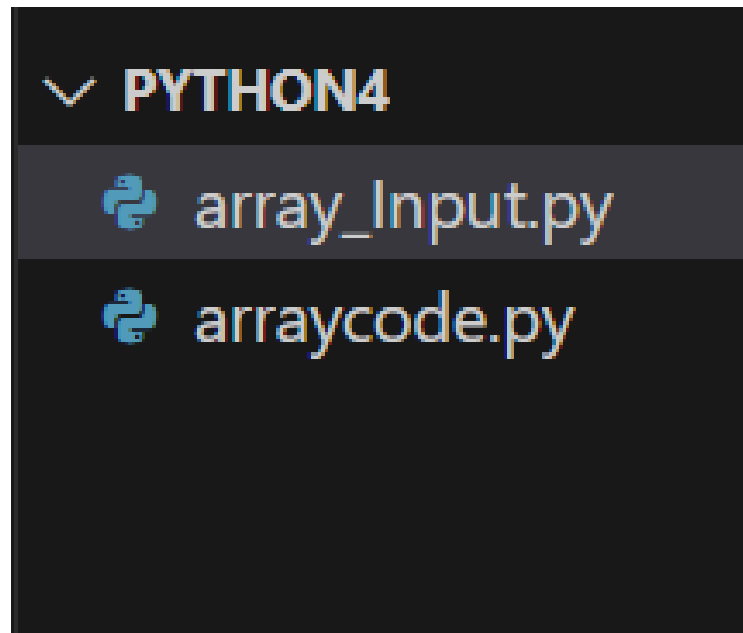
このページでは文字や数字を入力して配列に代入していくよ  
今回は人数分の名前と年齢を入力して配列を使って表示する  
ようにするよ

```
(array) PS C:\Users\小西 翔\python\python4> python array_Input.py
何人分のデータを入力しますか? : 3
--- 1人目 ---
名前を入力してください : 佐藤
年齢を入力してください : 24
--- 2人目 ---
名前を入力してください : 鈴木
年齢を入力してください : 29
--- 3人目 ---
名前を入力してください : 山本
年齢を入力してください : 22

入力されたデータ :
名前 : 佐藤 , 年齢 : 24
名前 : 鈴木 , 年齢 : 29
名前 : 山本 , 年齢 : 22
```

# ③文字の入力で配列に代入を試してみよう

新しいファイルを作ろう



## ③文字の入力で配列に代入を試してみよう

人数の入力をできるようにしよう

ここでちゃんと例外処理のコードも書いていこう

```
array_input.py > ...
1  try:
2      num_people = int(input("何人分のデータを入力しますか?: "))
3
4  except ValueError:
5      print("エラー: 人数は整数で入力してください。")
6  except Exception as e:
7      print(f"予期せぬエラーが発生しました: {e}")
```

## ③文字の入力で配列に代入を試してみよう

for文を使って人数分のデータを入力する準備を行おう

```
2     num_people = int(input("何人分のデータを入力しますか?: "))
3
4     for i in range(num_people):
5         print(f"--- {i+1}人目 ---")
6
7     except ValueError:
8         print("エラー: 人数は整数で入力してください。")
```

## ③文字の入力で配列に代入を試してみよう

while文を使って名前を入力できるようにしよう

```
4     for i in range(num_people):
5         print(f"--- {i+1}人目 ---")
6
7         while True:
8             name = input("名前を入力してください: ")
9             if name.strip(): # 空白でない場合のみ受け付ける
10                break
11            else:
12                print("名前が空白です。再度入力してください。")
13
```



# 説明

「while文」:特定の条件が真(True)である限り、ブロック内の文を繰り返し実行する制御構造

例)iの値が10のとき



```
while i > 5:  
    処理実行
```



```
while i < 5:  
    処理実行
```

## ③文字の入力で配列に代入を試してみよう

年齢の入力もコードを書いていこう

```
6
7     while True:
8         name = input("名前を入力してください: ")
9         if name.strip(): # 空白でない場合のみ受け付ける
10            break
11        else:
12            print("名前が空白です。再度入力してください。")
13
14        while True:
15            try:
16                age = int(input("年齢を入力してください: "))
17                break
18            except ValueError:
19                print("年齢は数字で入力してください。")
20
21    except ValueError:
22        print("エラー: 人数は整数で入力してください。")
```

## ③文字の入力で配列に代入を試してみよう

ここまで出来たら実行してみよう

```
(array) PS C:\Users\小西 翔\python\python4> python array_Input.py
何人分のデータを入力しますか? : 3
--- 1人目 ---
名前を入力してください : 佐藤
年齢を入力してください : 24
--- 2人目 ---
名前を入力してください : 山本
年齢を入力してください : 29
--- 3人目 ---
名前を入力してください : 鈴木
年齢を入力してください : 22
(array) PS C:\Users\小西 翔\python\python4>
```

## ③文字の入力で配列に代入を試してみよう

次に配列に代入するようにしてみよう

まずは配列の宣言を試してみよう

```
1  people = []
2
3  try:
4      num_people = int(input("何人分のデータを入力しますか?: "))
5
6      for i in range(num_people):
7          print(f"--- {i+1}人目 ---")
8
9          while True:
10             name = input("名前を入力してください: ")
```

## ③文字の入力で配列に代入を試してみよう

入力したものに名前を付けよう

今回はnameとageという名前にするよ

```
14         print("名前が空白です。再度入力してください。")
15
16     while True:
17         try:
18             age = int(input("年齢を入力してください: "))
19             break
20         except ValueError:
21             print("年齢は数字で入力してください。")
22
23     person = {"name": name, "age": age}
24     people.append(person)
25
```

## ③文字の入力で配列に代入を試してみよう

名前を付けると配列の宣言で

`array[2]`でもいいし、`array[name]`のように`[]`の中に名前を入力してもデータを取り出せるようにできるよ

# 説明

「配列.append(i):配列の中にiのデータを追加  
例)

```
array = [1,2,3]
```

```
array.append(10)
```

arrayの配列が[1,2,3,10]になるよ

## ③文字の入力で配列に代入を試してみよう

最後に入力したデータを出力してみよう

```
13         else:
14             print("名前が空白です。再度入力してください。")
15
16     while True:
17         try:
18             age = int(input("年齢を入力してください: "))
19             break
20         except ValueError:
21             print("年齢は数字で入力してください。")
22
23     person = {"name": name, "age": age}
24     people.append(person)
25
26     print("\n入力されたデータ:")
27     for person in people:
28         print(f"名前: {person['name']}, 年齢: {person['age']}")
29
```



## ③文字の入力で配列に代入を試してみよう

実行して入力したデータが最後にまとめて表示されるようになったかな？

```
(array) PS C:\Users\小西 翔\python\python4> python array_Input.py
何人分のデータを入力しますか？ : 3
--- 1人目 ---
名前を入力してください : 佐藤
年齢を入力してください : 24
--- 2人目 ---
名前を入力してください : 鈴木
年齢を入力してください : 29
--- 3人目 ---
名前を入力してください : 山本
年齢を入力してください : 22

入力されたデータ :
名前 : 佐藤 , 年齢 : 24
名前 : 鈴木 , 年齢 : 29
名前 : 山本 , 年齢 : 22
```

# お疲れさまでした

テキストは終了です。  
あとは自分なりにアレンジを付け加えていこう！