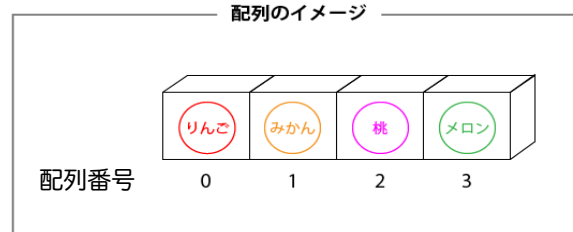


1 配列について復習しよう。

配列とは・・・同じ種類のデータを複数含んだ変数のこと

例) 果物が 4 個入った配列 a がある場合、3 個目に入っている果物は、
a[2] のようにして参照する。この場合の要素数は 4。

**超重要**

配列番号を添字といいます！問題の指示は主に 2 つです。
必ず読んで確認してください

①配列の添字は 0 から始まるものとする

②配列の添字は 1 から始まるものとする

2 線形探索について知ろう。

(① 線形探索)・・・先頭から順番に目的のデータを探す方法

例) 3 を探す

1 番目 →	④	1	3	5
2 番目 →	4	①	3	5
3 番目 →	4	1	③	5

練習 1 配列を書いていこう。

```
data=[43,24,18,77,55]
```

```
print(data[2])
```


配列の書き方について

- ・ data[] の中に書いていきます。
- ・ a は配列名なので何でもかまいません。

練習 2 練習 1 を順番に表示されるようにしよう

```
data=[43,24,18,77,55]
```

```
for i in range(0,5):  
    print(data[i])
```



このプログラムについて

- ・ 配列番号が 0 から始まります。
- ・ 終わりが配列番号 4 の 55

練習 3 外部からの入力ができるように設定します

```
data=[43,24,18,77,55]
```

```
a=input("探している数字は?")  
atai=int(a)
```

```
for i in range(0,5):  
    print(data[i])
```

このプログラムについて

- ・ 探している数字が atai の箱に入るようにします

練習 4 探している数字が見つかった場合とそうでない場合を設定します

```
data=[43,24,18,77,55]
```

```
a=input("探している数字は?")  
atai=int(a)
```

```
kekka=0
```



```
for i in range(0,5):  
    if data[i]==atai:  
  
if kekka==0:  
    print("見つかりませんでした")  
else:  
    print(kekka,"番目で発見")
```

このプログラムについて

- ・ kekka=0 を初期値に設定
- ・ 見つければ kekka の変数に i+1 の値をいれる
- ・ kekka が初期値のまま 0 だと見つからない状態

● 上手いこと動いた人は問題集→p.94～p.95（線形探索）