

# Notebook\_chapitre\_4\_Numpy

February 1, 2023

## 1 2. Les tableaux NumPy

### 1.1 2.1. Créer un ndarray

#### 1.1.1 2.1.1. Créer un ndarray à partir de listes

```
[1]: import numpy as np
notre_tableau=np.array([1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20])
print(notre_tableau)
type(notre_tableau)
```

```
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20]
```

```
[1]: numpy.ndarray
```

```
[2]: import numpy as np
mon_array_bidimensionnel=np.array([[1,2,3],[4,5,6],[7,8,9]])
print(mon_array_bidimensionnel)
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```

```
[3]: #mon_array_bidimensionnel[0,0]="texte"
```

```
[4]: mon_array_bidimensionnel.dtype
```

```
[4]: dtype('int64')
```

```
[5]: mon_array_bidimensionnel=np.array([[1.5,2,3],[4,5.2,6],[7,8,9.1]], dtype =_
↪float)
print(mon_array_bidimensionnel)
mon_array_bidimensionnel.dtype
```

```
[[1.5 2.  3. ]
 [4.  5.2 6. ]
 [7.  8.  9.1]]
```

```
[5]: dtype('float64')
```

### 1.1.2 2.1.2. Créer un ndarray grâce à des fonctions NumPy

```
[6]: import numpy as np
mon_array_zeros=np.zeros((4,6))
print(mon_array_zeros)
```

```
[[0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0.]]
```

```
[7]: import numpy as np
mon_array_zeros=np.zeros((4,6), dtype=int)
print(mon_array_zeros)
```

```
[[0 0 0 0 0 0]
 [0 0 0 0 0 0]
 [0 0 0 0 0 0]
 [0 0 0 0 0 0]]
```

```
[8]: import numpy as np
mon_array_ones=np.ones((4,6), dtype=int)
print(mon_array_ones)
```

```
[[1 1 1 1 1 1]
 [1 1 1 1 1 1]
 [1 1 1 1 1 1]
 [1 1 1 1 1 1]]
```

```
[9]: import numpy as np
mon_array_empty=np.empty((4,6), dtype=int)
print(mon_array_empty)
```

```
[[ 94561617505136          0 140361944713456 140361944713456
 140361956566192 140361956567152]
 [140361956567152 140361957854496 140361944713520 140361957854496
 140361957854496 140361957982272]
 [140361957982272          0          0          0
          0          0]
 [          0          0          0          0
          0          0]]
```

```
[10]: import numpy as np
from numpy import random
np.random.random((4,2))
```

```
[10]: array([[0.51710019, 0.20398919],
            [0.14923524, 0.60108345],
```

```
[0.10947357, 0.84807995],  
[0.94266619, 0.71952252]])
```

### 1.1.3 2.1.3. Créer un ndarray à partir d'un fichier

```
[11]: import numpy as np  
iris=np.genfromtxt("../datasets/iris.csv",  
↳delimiter=",",skip_header=True,encoding="latin-1")#, dtype="unicode")  
print(iris)
```

```
[[5.1 3.5 1.4 0.2 1. ]  
 [4.9 3.  1.4 0.2 1. ]  
 [4.7 3.2 1.3 0.2 1. ]  
 [4.6 3.1 1.5 0.2 1. ]  
 [5.  3.6 1.4 0.2 1. ]  
 [5.4 3.9 1.7 0.4 1. ]  
 [4.6 3.4 1.4 0.3 1. ]  
 [5.  3.4 1.5 0.2 1. ]  
 [4.4 2.9 1.4 0.2 1. ]  
 [4.9 3.1 1.5 0.1 1. ]  
 [5.4 3.7 1.5 0.2 1. ]  
 [4.8 3.4 1.6 0.2 1. ]  
 [4.8 3.  1.4 0.1 1. ]  
 [4.3 3.  1.1 0.1 1. ]  
 [5.8 4.  1.2 0.2 1. ]  
 [5.7 4.4 1.5 0.4 1. ]  
 [5.4 3.9 1.3 0.4 1. ]  
 [5.1 3.5 1.4 0.3 1. ]  
 [5.7 3.8 1.7 0.3 1. ]  
 [5.1 3.8 1.5 0.3 1. ]  
 [5.4 3.4 1.7 0.2 1. ]  
 [5.1 3.7 1.5 0.4 1. ]  
 [4.6 3.6 1.  0.2 1. ]  
 [5.1 3.3 1.7 0.5 1. ]  
 [4.8 3.4 1.9 0.2 1. ]  
 [5.  3.  1.6 0.2 1. ]  
 [5.  3.4 1.6 0.4 1. ]  
 [5.2 3.5 1.5 0.2 1. ]  
 [5.2 3.4 1.4 0.2 1. ]  
 [4.7 3.2 1.6 0.2 1. ]  
 [4.8 3.1 1.6 0.2 1. ]  
 [5.4 3.4 1.5 0.4 1. ]  
 [5.2 4.1 1.5 0.1 1. ]  
 [5.5 4.2 1.4 0.2 1. ]  
 [4.9 3.1 1.5 0.2 1. ]  
 [5.  3.2 1.2 0.2 1. ]  
 [5.5 3.5 1.3 0.2 1. ]
```

[4.9 3.6 1.4 0.1 1. ]  
[4.4 3. 1.3 0.2 1. ]  
[5.1 3.4 1.5 0.2 1. ]  
[5. 3.5 1.3 0.3 1. ]  
[4.5 2.3 1.3 0.3 1. ]  
[4.4 3.2 1.3 0.2 1. ]  
[5. 3.5 1.6 0.6 1. ]  
[5.1 3.8 1.9 0.4 1. ]  
[4.8 3. 1.4 0.3 1. ]  
[5.1 3.8 1.6 0.2 1. ]  
[4.6 3.2 1.4 0.2 1. ]  
[5.3 3.7 1.5 0.2 1. ]  
[5. 3.3 1.4 0.2 1. ]  
[7. 3.2 4.7 1.4 2. ]  
[6.4 3.2 4.5 1.5 2. ]  
[6.9 3.1 4.9 1.5 2. ]  
[5.5 2.3 4. 1.3 2. ]  
[6.5 2.8 4.6 1.5 2. ]  
[5.7 2.8 4.5 1.3 2. ]  
[6.3 3.3 4.7 1.6 2. ]  
[4.9 2.4 3.3 1. 2. ]  
[6.6 2.9 4.6 1.3 2. ]  
[5.2 2.7 3.9 1.4 2. ]  
[5. 2. 3.5 1. 2. ]  
[5.9 3. 4.2 1.5 2. ]  
[6. 2.2 4. 1. 2. ]  
[6.1 2.9 4.7 1.4 2. ]  
[5.6 2.9 3.6 1.3 2. ]  
[6.7 3.1 4.4 1.4 2. ]  
[5.6 3. 4.5 1.5 2. ]  
[5.8 2.7 4.1 1. 2. ]  
[6.2 2.2 4.5 1.5 2. ]  
[5.6 2.5 3.9 1.1 2. ]  
[5.9 3.2 4.8 1.8 2. ]  
[6.1 2.8 4. 1.3 2. ]  
[6.3 2.5 4.9 1.5 2. ]  
[6.1 2.8 4.7 1.2 2. ]  
[6.4 2.9 4.3 1.3 2. ]  
[6.6 3. 4.4 1.4 2. ]  
[6.8 2.8 4.8 1.4 2. ]  
[6.7 3. 5. 1.7 2. ]  
[6. 2.9 4.5 1.5 2. ]  
[5.7 2.6 3.5 1. 2. ]  
[5.5 2.4 3.8 1.1 2. ]  
[5.5 2.4 3.7 1. 2. ]  
[5.8 2.7 3.9 1.2 2. ]  
[6. 2.7 5.1 1.6 2. ]  
[5.4 3. 4.5 1.5 2. ]

[6. 3.4 4.5 1.6 2. ]  
[6.7 3.1 4.7 1.5 2. ]  
[6.3 2.3 4.4 1.3 2. ]  
[5.6 3. 4.1 1.3 2. ]  
[5.5 2.5 4. 1.3 2. ]  
[5.5 2.6 4.4 1.2 2. ]  
[6.1 3. 4.6 1.4 2. ]  
[5.8 2.6 4. 1.2 2. ]  
[5. 2.3 3.3 1. 2. ]  
[5.6 2.7 4.2 1.3 2. ]  
[5.7 3. 4.2 1.2 2. ]  
[5.7 2.9 4.2 1.3 2. ]  
[6.2 2.9 4.3 1.3 2. ]  
[5.1 2.5 3. 1.1 2. ]  
[5.7 2.8 4.1 1.3 2. ]  
[6.3 3.3 6. 2.5 3. ]  
[5.8 2.7 5.1 1.9 3. ]  
[7.1 3. 5.9 2.1 3. ]  
[6.3 2.9 5.6 1.8 3. ]  
[6.5 3. 5.8 2.2 3. ]  
[7.6 3. 6.6 2.1 3. ]  
[4.9 2.5 4.5 1.7 3. ]  
[7.3 2.9 6.3 1.8 3. ]  
[6.7 2.5 5.8 1.8 3. ]  
[7.2 3.6 6.1 2.5 3. ]  
[6.5 3.2 5.1 2. 3. ]  
[6.4 2.7 5.3 1.9 3. ]  
[6.8 3. 5.5 2.1 3. ]  
[5.7 2.5 5. 2. 3. ]  
[5.8 2.8 5.1 2.4 3. ]  
[6.4 3.2 5.3 2.3 3. ]  
[6.5 3. 5.5 1.8 3. ]  
[7.7 3.8 6.7 2.2 3. ]  
[7.7 2.6 6.9 2.3 3. ]  
[6. 2.2 5. 1.5 3. ]  
[6.9 3.2 5.7 2.3 3. ]  
[5.6 2.8 4.9 2. 3. ]  
[7.7 2.8 6.7 2. 3. ]  
[6.3 2.7 4.9 1.8 3. ]  
[6.7 3.3 5.7 2.1 3. ]  
[7.2 3.2 6. 1.8 3. ]  
[6.2 2.8 4.8 1.8 3. ]  
[6.1 3. 4.9 1.8 3. ]  
[6.4 2.8 5.6 2.1 3. ]  
[7.2 3. 5.8 1.6 3. ]  
[7.4 2.8 6.1 1.9 3. ]  
[7.9 3.8 6.4 2. 3. ]  
[6.4 2.8 5.6 2.2 3. ]

```
[6.3 2.8 5.1 1.5 3. ]
[6.1 2.6 5.6 1.4 3. ]
[7.7 3.  6.1 2.3 3. ]
[6.3 3.4 5.6 2.4 3. ]
[6.4 3.1 5.5 1.8 3. ]
[6.  3.  4.8 1.8 3. ]
[6.9 3.1 5.4 2.1 3. ]
[6.7 3.1 5.6 2.4 3. ]
[6.9 3.1 5.1 2.3 3. ]
[5.8 2.7 5.1 1.9 3. ]
[6.8 3.2 5.9 2.3 3. ]
[6.7 3.3 5.7 2.5 3. ]
[6.7 3.  5.2 2.3 3. ]
[6.3 2.5 5.  1.9 3. ]
[6.5 3.  5.2 2.  3. ]
[6.2 3.4 5.4 2.3 3. ]
[5.9 3.  5.1 1.8 3. ]]
```

```
[12]: import numpy as np
      np.savetxt("iris_2.csv", iris, delimiter=",")
```

## 1.2 2.2. Indexation

### 1.2.1 2.2.1. Indexation simple

```
[13]: import numpy as np
      iris=np.genfromtxt("../datasets/iris.csv",
      ↪delimiter=",",skip_header=True,encoding="latin-1")
      iris[2,1]
```

[13]: 3.2

```
[14]: iris[2,1]=4
      iris[2,1]
```

[14]: 4.0

```
[15]: iris[2,1]=3.2
      iris[2,1]
```

[15]: 3.2

### 1.2.2 2.2.2. Indexation booléenne

```
[16]: iris[iris >2]
```

```
[16]: array([5.1, 3.5, 4.9, 3. , 4.7, 3.2, 4.6, 3.1, 5. , 3.6, 5.4, 3.9, 4.6,
          3.4, 5. , 3.4, 4.4, 2.9, 4.9, 3.1, 5.4, 3.7, 4.8, 3.4, 4.8, 3. ,
          4.3, 3. , 5.8, 4. , 5.7, 4.4, 5.4, 3.9, 5.1, 3.5, 5.7, 3.8, 5.1,
          3.8, 5.4, 3.4, 5.1, 3.7, 4.6, 3.6, 5.1, 3.3, 4.8, 3.4, 5. , 3. ,
          5. , 3.4, 5.2, 3.5, 5.2, 3.4, 4.7, 3.2, 4.8, 3.1, 5.4, 3.4, 5.2,
          4.1, 5.5, 4.2, 4.9, 3.1, 5. , 3.2, 5.5, 3.5, 4.9, 3.6, 4.4, 3. ,
          5.1, 3.4, 5. , 3.5, 4.5, 2.3, 4.4, 3.2, 5. , 3.5, 5.1, 3.8, 4.8,
          3. , 5.1, 3.8, 4.6, 3.2, 5.3, 3.7, 5. , 3.3, 7. , 3.2, 4.7, 6.4,
          3.2, 4.5, 6.9, 3.1, 4.9, 5.5, 2.3, 4. , 6.5, 2.8, 4.6, 5.7, 2.8,
          4.5, 6.3, 3.3, 4.7, 4.9, 2.4, 3.3, 6.6, 2.9, 4.6, 5.2, 2.7, 3.9,
          5. , 3.5, 5.9, 3. , 4.2, 6. , 2.2, 4. , 6.1, 2.9, 4.7, 5.6, 2.9,
          3.6, 6.7, 3.1, 4.4, 5.6, 3. , 4.5, 5.8, 2.7, 4.1, 6.2, 2.2, 4.5,
          5.6, 2.5, 3.9, 5.9, 3.2, 4.8, 6.1, 2.8, 4. , 6.3, 2.5, 4.9, 6.1,
          2.8, 4.7, 6.4, 2.9, 4.3, 6.6, 3. , 4.4, 6.8, 2.8, 4.8, 6.7, 3. ,
          5. , 6. , 2.9, 4.5, 5.7, 2.6, 3.5, 5.5, 2.4, 3.8, 5.5, 2.4, 3.7,
          5.8, 2.7, 3.9, 6. , 2.7, 5.1, 5.4, 3. , 4.5, 6. , 3.4, 4.5, 6.7,
          3.1, 4.7, 6.3, 2.3, 4.4, 5.6, 3. , 4.1, 5.5, 2.5, 4. , 5.5, 2.6,
          4.4, 6.1, 3. , 4.6, 5.8, 2.6, 4. , 5. , 2.3, 3.3, 5.6, 2.7, 4.2,
          5.7, 3. , 4.2, 5.7, 2.9, 4.2, 6.2, 2.9, 4.3, 5.1, 2.5, 3. , 5.7,
          2.8, 4.1, 6.3, 3.3, 6. , 2.5, 3. , 5.8, 2.7, 5.1, 3. , 7.1, 3. ,
          5.9, 2.1, 3. , 6.3, 2.9, 5.6, 3. , 6.5, 3. , 5.8, 2.2, 3. , 7.6,
          3. , 6.6, 2.1, 3. , 4.9, 2.5, 4.5, 3. , 7.3, 2.9, 6.3, 3. , 6.7,
          2.5, 5.8, 3. , 7.2, 3.6, 6.1, 2.5, 3. , 6.5, 3.2, 5.1, 3. , 6.4,
          2.7, 5.3, 3. , 6.8, 3. , 5.5, 2.1, 3. , 5.7, 2.5, 5. , 3. , 5.8,
          2.8, 5.1, 2.4, 3. , 6.4, 3.2, 5.3, 2.3, 3. , 6.5, 3. , 5.5, 3. ,
          7.7, 3.8, 6.7, 2.2, 3. , 7.7, 2.6, 6.9, 2.3, 3. , 6. , 2.2, 5. ,
          3. , 6.9, 3.2, 5.7, 2.3, 3. , 5.6, 2.8, 4.9, 3. , 7.7, 2.8, 6.7,
          3. , 6.3, 2.7, 4.9, 3. , 6.7, 3.3, 5.7, 2.1, 3. , 7.2, 3.2, 6. ,
          3. , 6.2, 2.8, 4.8, 3. , 6.1, 3. , 4.9, 3. , 6.4, 2.8, 5.6, 2.1,
          3. , 7.2, 3. , 5.8, 3. , 7.4, 2.8, 6.1, 3. , 7.9, 3.8, 6.4, 3. ,
          6.4, 2.8, 5.6, 2.2, 3. , 6.3, 2.8, 5.1, 3. , 6.1, 2.6, 5.6, 3. ,
          7.7, 3. , 6.1, 2.3, 3. , 6.3, 3.4, 5.6, 2.4, 3. , 6.4, 3.1, 5.5,
          3. , 6. , 3. , 4.8, 3. , 6.9, 3.1, 5.4, 2.1, 3. , 6.7, 3.1, 5.6,
          2.4, 3. , 6.9, 3.1, 5.1, 2.3, 3. , 5.8, 2.7, 5.1, 3. , 6.8, 3.2,
          5.9, 2.3, 3. , 6.7, 3.3, 5.7, 2.5, 3. , 6.7, 3. , 5.2, 2.3, 3. ,
          6.3, 2.5, 5. , 3. , 6.5, 3. , 5.2, 3. , 6.2, 3.4, 5.4, 2.3, 3. ,
          5.9, 3. , 5.1, 3. ])
```

```
[17]: iris[iris >2].shape
```

```
[17]: (472,)
```

```
[18]: iris >2
```

```
[18]: array([[ True,  True, False, False, False],
          [ True,  True, False, False, False],
          [ True,  True, False, False, False],
```







```
[ True,  True,  True, False, False],
[ True,  True,  True, False, False],
[ True,  True,  True, False, False],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
```

```
[ True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True, False,  True],
[ True,  True,  True,  True,  True],
[ True,  True,  True, False,  True]])
```

```
[19]: iris[iris[:,0] >5]
```

```
[19]: array([[5.1, 3.5, 1.4, 0.2, 1. ],
             [5.4, 3.9, 1.7, 0.4, 1. ],
             [5.4, 3.7, 1.5, 0.2, 1. ],
             [5.8, 4. , 1.2, 0.2, 1. ],
             [5.7, 4.4, 1.5, 0.4, 1. ],
             [5.4, 3.9, 1.3, 0.4, 1. ],
             [5.1, 3.5, 1.4, 0.3, 1. ],
             [5.7, 3.8, 1.7, 0.3, 1. ],
             [5.1, 3.8, 1.5, 0.3, 1. ],
             [5.4, 3.4, 1.7, 0.2, 1. ],
             [5.1, 3.7, 1.5, 0.4, 1. ],
             [5.1, 3.3, 1.7, 0.5, 1. ],
             [5.2, 3.5, 1.5, 0.2, 1. ],
             [5.2, 3.4, 1.4, 0.2, 1. ],
             [5.4, 3.4, 1.5, 0.4, 1. ],
             [5.2, 4.1, 1.5, 0.1, 1. ],
             [5.5, 4.2, 1.4, 0.2, 1. ],
             [5.5, 3.5, 1.3, 0.2, 1. ],
             [5.1, 3.4, 1.5, 0.2, 1. ],
             [5.1, 3.8, 1.9, 0.4, 1. ],
             [5.1, 3.8, 1.6, 0.2, 1. ],
             [5.3, 3.7, 1.5, 0.2, 1. ],
             [7. , 3.2, 4.7, 1.4, 2. ],
             [6.4, 3.2, 4.5, 1.5, 2. ],
             [6.9, 3.1, 4.9, 1.5, 2. ],
             [5.5, 2.3, 4. , 1.3, 2. ],
             [6.5, 2.8, 4.6, 1.5, 2. ],
             [5.7, 2.8, 4.5, 1.3, 2. ],
             [6.3, 3.3, 4.7, 1.6, 2. ],
             [6.6, 2.9, 4.6, 1.3, 2. ],
             [5.2, 2.7, 3.9, 1.4, 2. ],
             [5.9, 3. , 4.2, 1.5, 2. ],
             [6. , 2.2, 4. , 1. , 2. ],
             [6.1, 2.9, 4.7, 1.4, 2. ],
             [5.6, 2.9, 3.6, 1.3, 2. ],
             [6.7, 3.1, 4.4, 1.4, 2. ],
             [5.6, 3. , 4.5, 1.5, 2. ],
             [5.8, 2.7, 4.1, 1. , 2. ],
```

[6.2, 2.2, 4.5, 1.5, 2. ],  
[5.6, 2.5, 3.9, 1.1, 2. ],  
[5.9, 3.2, 4.8, 1.8, 2. ],  
[6.1, 2.8, 4. , 1.3, 2. ],  
[6.3, 2.5, 4.9, 1.5, 2. ],  
[6.1, 2.8, 4.7, 1.2, 2. ],  
[6.4, 2.9, 4.3, 1.3, 2. ],  
[6.6, 3. , 4.4, 1.4, 2. ],  
[6.8, 2.8, 4.8, 1.4, 2. ],  
[6.7, 3. , 5. , 1.7, 2. ],  
[6. , 2.9, 4.5, 1.5, 2. ],  
[5.7, 2.6, 3.5, 1. , 2. ],  
[5.5, 2.4, 3.8, 1.1, 2. ],  
[5.5, 2.4, 3.7, 1. , 2. ],  
[5.8, 2.7, 3.9, 1.2, 2. ],  
[6. , 2.7, 5.1, 1.6, 2. ],  
[5.4, 3. , 4.5, 1.5, 2. ],  
[6. , 3.4, 4.5, 1.6, 2. ],  
[6.7, 3.1, 4.7, 1.5, 2. ],  
[6.3, 2.3, 4.4, 1.3, 2. ],  
[5.6, 3. , 4.1, 1.3, 2. ],  
[5.5, 2.5, 4. , 1.3, 2. ],  
[5.5, 2.6, 4.4, 1.2, 2. ],  
[6.1, 3. , 4.6, 1.4, 2. ],  
[5.8, 2.6, 4. , 1.2, 2. ],  
[5.6, 2.7, 4.2, 1.3, 2. ],  
[5.7, 3. , 4.2, 1.2, 2. ],  
[5.7, 2.9, 4.2, 1.3, 2. ],  
[6.2, 2.9, 4.3, 1.3, 2. ],  
[5.1, 2.5, 3. , 1.1, 2. ],  
[5.7, 2.8, 4.1, 1.3, 2. ],  
[6.3, 3.3, 6. , 2.5, 3. ],  
[5.8, 2.7, 5.1, 1.9, 3. ],  
[7.1, 3. , 5.9, 2.1, 3. ],  
[6.3, 2.9, 5.6, 1.8, 3. ],  
[6.5, 3. , 5.8, 2.2, 3. ],  
[7.6, 3. , 6.6, 2.1, 3. ],  
[7.3, 2.9, 6.3, 1.8, 3. ],  
[6.7, 2.5, 5.8, 1.8, 3. ],  
[7.2, 3.6, 6.1, 2.5, 3. ],  
[6.5, 3.2, 5.1, 2. , 3. ],  
[6.4, 2.7, 5.3, 1.9, 3. ],  
[6.8, 3. , 5.5, 2.1, 3. ],  
[5.7, 2.5, 5. , 2. , 3. ],  
[5.8, 2.8, 5.1, 2.4, 3. ],  
[6.4, 3.2, 5.3, 2.3, 3. ],  
[6.5, 3. , 5.5, 1.8, 3. ],

```

[7.7, 3.8, 6.7, 2.2, 3. ],
[7.7, 2.6, 6.9, 2.3, 3. ],
[6. , 2.2, 5. , 1.5, 3. ],
[6.9, 3.2, 5.7, 2.3, 3. ],
[5.6, 2.8, 4.9, 2. , 3. ],
[7.7, 2.8, 6.7, 2. , 3. ],
[6.3, 2.7, 4.9, 1.8, 3. ],
[6.7, 3.3, 5.7, 2.1, 3. ],
[7.2, 3.2, 6. , 1.8, 3. ],
[6.2, 2.8, 4.8, 1.8, 3. ],
[6.1, 3. , 4.9, 1.8, 3. ],
[6.4, 2.8, 5.6, 2.1, 3. ],
[7.2, 3. , 5.8, 1.6, 3. ],
[7.4, 2.8, 6.1, 1.9, 3. ],
[7.9, 3.8, 6.4, 2. , 3. ],
[6.4, 2.8, 5.6, 2.2, 3. ],
[6.3, 2.8, 5.1, 1.5, 3. ],
[6.1, 2.6, 5.6, 1.4, 3. ],
[7.7, 3. , 6.1, 2.3, 3. ],
[6.3, 3.4, 5.6, 2.4, 3. ],
[6.4, 3.1, 5.5, 1.8, 3. ],
[6. , 3. , 4.8, 1.8, 3. ],
[6.9, 3.1, 5.4, 2.1, 3. ],
[6.7, 3.1, 5.6, 2.4, 3. ],
[6.9, 3.1, 5.1, 2.3, 3. ],
[5.8, 2.7, 5.1, 1.9, 3. ],
[6.8, 3.2, 5.9, 2.3, 3. ],
[6.7, 3.3, 5.7, 2.5, 3. ],
[6.7, 3. , 5.2, 2.3, 3. ],
[6.3, 2.5, 5. , 1.9, 3. ],
[6.5, 3. , 5.2, 2. , 3. ],
[6.2, 3.4, 5.4, 2.3, 3. ],
[5.9, 3. , 5.1, 1.8, 3. ]])

```

```
[20]: iris[:,0] >5
```

```

[20]: array([ True, False, False, False, False,  True, False, False, False,
          False,  True, False, False, False,  True,  True,  True,  True,
           True,  True,  True,  True, False,  True, False, False, False,
           True,  True, False, False,  True,  True,  True, False, False,
           True, False, False,  True, False, False, False, False,  True,
          False,  True, False,  True, False,  True,  True,  True,  True,
           True,  True,  True, False,  True,  True, False,  True,  True,
           True,  True,  True,  True,  True,  True,  True,  True,  True,
           True,  True,  True,  True,  True,  True,  True,  True,  True,
           True,  True,  True,  True,  True,  True,  True,  True,  True,
           True,  True,  True, False,  True,  True,  True,  True,  True,

```

```
True, True, True, True, True, True, True, False, True,
True, True, True, True, True, True, True, True, True,
True, True, True, True, True, True, True, True, True,
True, True, True, True, True, True, True, True, True,
True, True, True, True, True, True, True, True, True,
True, True, True, True, True, True]
```

### 1.2.3 2.2.3. Fancy indexing

```
[21]: notre_tableau=np.array([1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20])
print("Contenu de notre_tableau :")
print(notre_tableau)
print("Nouveau tableau après fancy indexing :")
print(notre_tableau[np.array([[4,10,5],[12,8,9]])])
```

```
Contenu de notre_tableau :
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20]
Nouveau tableau après fancy indexing :
[[ 5 11  6]
 [13  9 10]]
```

## 1.3 2.3. Accéder aux éléments par tranche (slicing)

### 1.3.1 2.3.1. Slicing sur un tableau NumPy à 1 dimension

```
[22]: notre_tableau[:3]
```

```
[22]: array([1, 2, 3])
```

```
[23]: notre_tableau[10:]
```

```
[23]: array([11, 12, 13, 14, 15, 16, 17, 18, 19, 20])
```

```
[24]: notre_tableau[3:7]
```

```
[24]: array([4, 5, 6, 7])
```

```
[25]: notre_tableau[::2]
```

```
[25]: array([ 1,  3,  5,  7,  9, 11, 13, 15, 17, 19])
```

### 1.3.2 2.3.2. Slicing sur un tableau NumPy à 2 dimensions

```
[26]: iris[:,2,:4]
```

```
[26]: array([[5.1, 3.5, 1.4, 0.2],
            [4.9, 3. , 1.4, 0.2]])
```

```
[27]: iris[145:,2:]
```

```
[27]: array([[5.2, 2.3, 3. ],  
          [5. , 1.9, 3. ],  
          [5.2, 2. , 3. ],  
          [5.4, 2.3, 3. ],  
          [5.1, 1.8, 3. ]])
```

```
[28]: iris[0,:]
```

```
[28]: array([5.1, 3.5, 1.4, 0.2, 1. ])
```

```
[29]: iris[:,0]
```

```
[29]: array([5.1, 4.9, 4.7, 4.6, 5. , 5.4, 4.6, 5. , 4.4, 4.9, 5.4, 4.8, 4.8,  
          4.3, 5.8, 5.7, 5.4, 5.1, 5.7, 5.1, 5.4, 5.1, 4.6, 5.1, 4.8, 5. ,  
          5. , 5.2, 5.2, 4.7, 4.8, 5.4, 5.2, 5.5, 4.9, 5. , 5.5, 4.9, 4.4,  
          5.1, 5. , 4.5, 4.4, 5. , 5.1, 4.8, 5.1, 4.6, 5.3, 5. , 7. , 6.4,  
          6.9, 5.5, 6.5, 5.7, 6.3, 4.9, 6.6, 5.2, 5. , 5.9, 6. , 6.1, 5.6,  
          6.7, 5.6, 5.8, 6.2, 5.6, 5.9, 6.1, 6.3, 6.1, 6.4, 6.6, 6.8, 6.7,  
          6. , 5.7, 5.5, 5.5, 5.8, 6. , 5.4, 6. , 6.7, 6.3, 5.6, 5.5, 5.5,  
          6.1, 5.8, 5. , 5.6, 5.7, 5.7, 6.2, 5.1, 5.7, 6.3, 5.8, 7.1, 6.3,  
          6.5, 7.6, 4.9, 7.3, 6.7, 7.2, 6.5, 6.4, 6.8, 5.7, 5.8, 6.4, 6.5,  
          7.7, 7.7, 6. , 6.9, 5.6, 7.7, 6.3, 6.7, 7.2, 6.2, 6.1, 6.4, 7.2,  
          7.4, 7.9, 6.4, 6.3, 6.1, 7.7, 6.3, 6.4, 6. , 6.9, 6.7, 6.9, 5.8,  
          6.8, 6.7, 6.7, 6.3, 6.5, 6.2, 5.9])
```

```
[30]: iris[-1,:]
```

```
[30]: array([5.9, 3. , 5.1, 1.8, 3. ])
```

## 1.4 2.4. Notion de vue et copie

```
[31]: sous_tableau=notre_tableau[:5]  
      sous_tableau
```

```
[31]: array([1, 2, 3, 4, 5])
```

```
[32]: sous_tableau[0]=10  
      sous_tableau
```

```
[32]: array([10, 2, 3, 4, 5])
```

```
[33]: notre_tableau
```

```
[33]: array([10, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17,  
          18, 19, 20])
```

```
[34]: notre_tableau=np.array([1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20])
      sous_tableau=notre_tableau[:5].copy()
      sous_tableau[0]=10
      print(sous_tableau)
      print(notre_tableau)
```

```
[10  2  3  4  5]
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20]
```

## 2 3. Les opérations mathématiques avec NumPy

### 2.1 3.1. Les opérations arithmétiques

```
[35]: mon_tableau=np.array([[10,11,12],[13,14,15],[16,17,18]])
      mon_tableau
```

```
[35]: array([[10, 11, 12],
            [13, 14, 15],
            [16, 17, 18]])
```

```
[36]: import numpy as np
      np.add(mon_tableau,10)
```

```
[36]: array([[20, 21, 22],
            [23, 24, 25],
            [26, 27, 28]])
```

```
[37]: import numpy as np
      np.multiply(mon_tableau,20)
```

```
[37]: array([[200, 220, 240],
            [260, 280, 300],
            [320, 340, 360]])
```

```
[38]: mon_tableau2=np.array([10,20,30])
      mon_tableau
```

```
[38]: array([[10, 11, 12],
            [13, 14, 15],
            [16, 17, 18]])
```

```
[39]: import numpy as np
      np.subtract(mon_tableau,mon_tableau2)
```

```
[39]: array([[ 0, -9, -18],
            [ 3, -6, -15],
            [ 6, -3, -12]])
```



```
[40]: mon_tableau3=np.array([[1,2,3],[4,5,6],[7,8,9]])
mon_tableau3
```

```
[40]: array([[1, 2, 3],
           [4, 5, 6],
           [7, 8, 9]])
```

```
[41]: np.multiply(mon_tableau,mon_tableau3)
```

```
[41]: array([[ 10,  22,  36],
           [ 52,  70,  90],
           [112, 136, 162]])
```

## 2.2 3.2. Les fonctions d'agrégations

```
[42]: import numpy as np
print("Somme sur l'ensemble des éléments:")
print(np.sum(iris))
print("Somme par colonne:")
print(np.sum(iris, axis=0))
print("Somme par ligne:")
print(np.sum(iris, axis=1))
```

Somme sur l'ensemble des éléments:

2378.7

Somme par colonne:

[876.5 458.6 563.7 179.9 300. ]

Somme par ligne:

```
[11.2 10.5 10.4 10.4 11.2 12.4 10.7 11.1  9.9 10.6 11.8 11.  10.3  9.5
 12.2 13.  12.  11.3 12.5 11.7 11.7 11.7 10.4 11.6 11.3 10.8 11.4 11.4
 11.2 10.7 10.7 11.7 11.9 12.3 10.7 10.6 11.5 11.  9.9 11.2 11.1  9.4
 10.1 11.7 12.2 10.5 11.7 10.4 11.7 10.9 18.3 17.6 18.4 15.1 17.4 16.3
 17.9 13.6 17.4 15.2 13.5 16.6 15.2 17.1 15.4 17.6 16.6 15.6 16.4 15.1
 17.7 16.2 17.2 16.8 16.9 17.4 17.8 18.4 16.9 14.8 14.8 14.6 15.6 17.4
 16.4 17.5 18.  16.3 16.  15.3 15.7 17.1 15.6 13.6 15.8 16.1 16.1 16.7
 13.7 15.9 21.1 18.5 21.1 19.6 20.5 22.3 16.6 21.3 19.8 22.4 19.8 19.3
 20.4 18.2 19.1 20.2 19.8 23.4 22.5 17.7 21.1 18.3 22.2 18.7 20.8 21.2
 18.6 18.8 19.9 20.6 21.2 23.1 20.  18.7 18.7 22.1 20.7 19.8 18.6 20.5
 20.8 20.4 18.5 21.2 21.2 20.2 18.7 19.7 20.3 18.8]
```

```
[43]: import numpy as np
print(np.mean(iris, axis=0))
print(np.std(iris, axis=0))
```

```
[5.84333333 3.05733333 3.758      1.19933333 2.          ]
[0.82530129 0.43441097 1.75940407 0.75969263 0.81649658]
```

```
[44]: import numpy as np
print(np.max(iris, axis=0))
```

```
[7.9 4.4 6.9 2.5 3. ]
```

### 3 4. Inspecter un tableau grâce aux attributs de NumPy

```
[45]: print("Attribut size:")
print(iris.size)
print("Attribut shape:")
print(iris.shape)
print("Attribut ndim:")
print(iris.ndim)
print("Attribut dtype:")
print(iris.dtype)
```

```
Attribut size:
750
Attribut shape:
(150, 5)
Attribut ndim:
2
Attribut dtype:
float64
```

```
[46]: iris.T
```

```
[46]: array([[5.1, 4.9, 4.7, 4.6, 5. , 5.4, 4.6, 5. , 4.4, 4.9, 5.4, 4.8, 4.8,
4.3, 5.8, 5.7, 5.4, 5.1, 5.7, 5.1, 5.4, 5.1, 4.6, 5.1, 4.8, 5. ,
5. , 5.2, 5.2, 4.7, 4.8, 5.4, 5.2, 5.5, 4.9, 5. , 5.5, 4.9, 4.4,
5.1, 5. , 4.5, 4.4, 5. , 5.1, 4.8, 5.1, 4.6, 5.3, 5. , 7. , 6.4,
6.9, 5.5, 6.5, 5.7, 6.3, 4.9, 6.6, 5.2, 5. , 5.9, 6. , 6.1, 5.6,
6.7, 5.6, 5.8, 6.2, 5.6, 5.9, 6.1, 6.3, 6.1, 6.4, 6.6, 6.8, 6.7,
6. , 5.7, 5.5, 5.5, 5.8, 6. , 5.4, 6. , 6.7, 6.3, 5.6, 5.5, 5.5,
6.1, 5.8, 5. , 5.6, 5.7, 5.7, 6.2, 5.1, 5.7, 6.3, 5.8, 7.1, 6.3,
6.5, 7.6, 4.9, 7.3, 6.7, 7.2, 6.5, 6.4, 6.8, 5.7, 5.8, 6.4, 6.5,
7.7, 7.7, 6. , 6.9, 5.6, 7.7, 6.3, 6.7, 7.2, 6.2, 6.1, 6.4, 7.2,
7.4, 7.9, 6.4, 6.3, 6.1, 7.7, 6.3, 6.4, 6. , 6.9, 6.7, 6.9, 5.8,
6.8, 6.7, 6.7, 6.3, 6.5, 6.2, 5.9],
[3.5, 3. , 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3.4, 3. ,
3. , 4. , 4.4, 3.9, 3.5, 3.8, 3.8, 3.4, 3.7, 3.6, 3.3, 3.4, 3. ,
3.4, 3.5, 3.4, 3.2, 3.1, 3.4, 4.1, 4.2, 3.1, 3.2, 3.5, 3.6, 3. ,
3.4, 3.5, 2.3, 3.2, 3.5, 3.8, 3. , 3.8, 3.2, 3.7, 3.3, 3.2, 3.2,
3.1, 2.3, 2.8, 2.8, 3.3, 2.4, 2.9, 2.7, 2. , 3. , 2.2, 2.9, 2.9,
3.1, 3. , 2.7, 2.2, 2.5, 3.2, 2.8, 2.5, 2.8, 2.9, 3. , 2.8, 3. ,
2.9, 2.6, 2.4, 2.4, 2.7, 2.7, 3. , 3.4, 3.1, 2.3, 3. , 2.5, 2.6,
3. , 2.6, 2.3, 2.7, 3. , 2.9, 2.9, 2.5, 2.8, 3.3, 2.7, 3. , 2.9,
```

```

3. , 3. , 2.5, 2.9, 2.5, 3.6, 3.2, 2.7, 3. , 2.5, 2.8, 3.2, 3. ,
3.8, 2.6, 2.2, 3.2, 2.8, 2.8, 2.7, 3.3, 3.2, 2.8, 3. , 2.8, 3. ,
2.8, 3.8, 2.8, 2.8, 2.6, 3. , 3.4, 3.1, 3. , 3.1, 3.1, 3.1, 2.7,
3.2, 3.3, 3. , 2.5, 3. , 3.4, 3. ],
[1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.6, 1.4,
1.1, 1.2, 1.5, 1.3, 1.4, 1.7, 1.5, 1.7, 1.5, 1. , 1.7, 1.9, 1.6,
1.6, 1.5, 1.4, 1.6, 1.6, 1.5, 1.5, 1.4, 1.5, 1.2, 1.3, 1.4, 1.3,
1.5, 1.3, 1.3, 1.3, 1.6, 1.9, 1.4, 1.6, 1.4, 1.5, 1.4, 4.7, 4.5,
4.9, 4. , 4.6, 4.5, 4.7, 3.3, 4.6, 3.9, 3.5, 4.2, 4. , 4.7, 3.6,
4.4, 4.5, 4.1, 4.5, 3.9, 4.8, 4. , 4.9, 4.7, 4.3, 4.4, 4.8, 5. ,
4.5, 3.5, 3.8, 3.7, 3.9, 5.1, 4.5, 4.5, 4.7, 4.4, 4.1, 4. , 4.4,
4.6, 4. , 3.3, 4.2, 4.2, 4.2, 4.3, 3. , 4.1, 6. , 5.1, 5.9, 5.6,
5.8, 6.6, 4.5, 6.3, 5.8, 6.1, 5.1, 5.3, 5.5, 5. , 5.1, 5.3, 5.5,
6.7, 6.9, 5. , 5.7, 4.9, 6.7, 4.9, 5.7, 6. , 4.8, 4.9, 5.6, 5.8,
6.1, 6.4, 5.6, 5.1, 5.6, 6.1, 5.6, 5.5, 4.8, 5.4, 5.6, 5.1, 5.1,
5.9, 5.7, 5.2, 5. , 5.2, 5.4, 5.1],
[0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0.2, 0.1,
0.1, 0.2, 0.4, 0.4, 0.3, 0.3, 0.3, 0.2, 0.4, 0.2, 0.5, 0.2, 0.2,
0.4, 0.2, 0.2, 0.2, 0.2, 0.4, 0.1, 0.2, 0.2, 0.2, 0.2, 0.1, 0.2,
0.2, 0.3, 0.3, 0.2, 0.6, 0.4, 0.3, 0.2, 0.2, 0.2, 0.2, 1.4, 1.5,
1.5, 1.3, 1.5, 1.3, 1.6, 1. , 1.3, 1.4, 1. , 1.5, 1. , 1.4, 1.3,
1.4, 1.5, 1. , 1.5, 1.1, 1.8, 1.3, 1.5, 1.2, 1.3, 1.4, 1.4, 1.7,
1.5, 1. , 1.1, 1. , 1.2, 1.6, 1.5, 1.6, 1.5, 1.3, 1.3, 1.3, 1.2,
1.4, 1.2, 1. , 1.3, 1.2, 1.3, 1.3, 1.1, 1.3, 2.5, 1.9, 2.1, 1.8,
2.2, 2.1, 1.7, 1.8, 1.8, 2.5, 2. , 1.9, 2.1, 2. , 2.4, 2.3, 1.8,
2.2, 2.3, 1.5, 2.3, 2. , 2. , 1.8, 2.1, 1.8, 1.8, 1.8, 2.1, 1.6,
1.9, 2. , 2.2, 1.5, 1.4, 2.3, 2.4, 1.8, 1.8, 2.1, 2.4, 2.3, 1.9,
2.3, 2.5, 2.3, 1.9, 2. , 2.3, 1.8],
[1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. ,
1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. ,
1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. ,
1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 1. , 2. , 2. ,
2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. ,
2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. ,
2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. ,
2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 2. , 3. , 3. , 3. , 3. ,
3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. ,
3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. ,
3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. ,
3. , 3. , 3. , 3. , 3. , 3. , 3. , 3. ]])

```

```
[47]: (iris.T).shape
```

```
[47]: (5, 150)
```

## 4 5. Manipulation de tableaux NumPy

### 4.1 5.1. Ajouter et supprimer des éléments d'un tableau

#### 4.1.1 5.1.1. Ajouter des éléments dans un tableau

```
[48]: np.append(notre_tableau, [21,22,23,24,25])
```

```
[48]: array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
          18, 19, 20, 21, 22, 23, 24, 25])
```

```
[49]: np.append(iris, [[10,10,10,10,10]], axis=0)
```

```
[49]: array([[ 5.1,  3.5,  1.4,  0.2,  1. ],
          [ 4.9,  3. ,  1.4,  0.2,  1. ],
          [ 4.7,  3.2,  1.3,  0.2,  1. ],
          [ 4.6,  3.1,  1.5,  0.2,  1. ],
          [ 5. ,  3.6,  1.4,  0.2,  1. ],
          [ 5.4,  3.9,  1.7,  0.4,  1. ],
          [ 4.6,  3.4,  1.4,  0.3,  1. ],
          [ 5. ,  3.4,  1.5,  0.2,  1. ],
          [ 4.4,  2.9,  1.4,  0.2,  1. ],
          [ 4.9,  3.1,  1.5,  0.1,  1. ],
          [ 5.4,  3.7,  1.5,  0.2,  1. ],
          [ 4.8,  3.4,  1.6,  0.2,  1. ],
          [ 4.8,  3. ,  1.4,  0.1,  1. ],
          [ 4.3,  3. ,  1.1,  0.1,  1. ],
          [ 5.8,  4. ,  1.2,  0.2,  1. ],
          [ 5.7,  4.4,  1.5,  0.4,  1. ],
          [ 5.4,  3.9,  1.3,  0.4,  1. ],
          [ 5.1,  3.5,  1.4,  0.3,  1. ],
          [ 5.7,  3.8,  1.7,  0.3,  1. ],
          [ 5.1,  3.8,  1.5,  0.3,  1. ],
          [ 5.4,  3.4,  1.7,  0.2,  1. ],
          [ 5.1,  3.7,  1.5,  0.4,  1. ],
          [ 4.6,  3.6,  1. ,  0.2,  1. ],
          [ 5.1,  3.3,  1.7,  0.5,  1. ],
          [ 4.8,  3.4,  1.9,  0.2,  1. ],
          [ 5. ,  3. ,  1.6,  0.2,  1. ],
          [ 5. ,  3.4,  1.6,  0.4,  1. ],
          [ 5.2,  3.5,  1.5,  0.2,  1. ],
          [ 5.2,  3.4,  1.4,  0.2,  1. ],
          [ 4.7,  3.2,  1.6,  0.2,  1. ],
          [ 4.8,  3.1,  1.6,  0.2,  1. ],
          [ 5.4,  3.4,  1.5,  0.4,  1. ],
          [ 5.2,  4.1,  1.5,  0.1,  1. ],
          [ 5.5,  4.2,  1.4,  0.2,  1. ],
          [ 4.9,  3.1,  1.5,  0.2,  1. ],
```

[ 5. , 3.2, 1.2, 0.2, 1. ],  
[ 5.5, 3.5, 1.3, 0.2, 1. ],  
[ 4.9, 3.6, 1.4, 0.1, 1. ],  
[ 4.4, 3. , 1.3, 0.2, 1. ],  
[ 5.1, 3.4, 1.5, 0.2, 1. ],  
[ 5. , 3.5, 1.3, 0.3, 1. ],  
[ 4.5, 2.3, 1.3, 0.3, 1. ],  
[ 4.4, 3.2, 1.3, 0.2, 1. ],  
[ 5. , 3.5, 1.6, 0.6, 1. ],  
[ 5.1, 3.8, 1.9, 0.4, 1. ],  
[ 4.8, 3. , 1.4, 0.3, 1. ],  
[ 5.1, 3.8, 1.6, 0.2, 1. ],  
[ 4.6, 3.2, 1.4, 0.2, 1. ],  
[ 5.3, 3.7, 1.5, 0.2, 1. ],  
[ 5. , 3.3, 1.4, 0.2, 1. ],  
[ 7. , 3.2, 4.7, 1.4, 2. ],  
[ 6.4, 3.2, 4.5, 1.5, 2. ],  
[ 6.9, 3.1, 4.9, 1.5, 2. ],  
[ 5.5, 2.3, 4. , 1.3, 2. ],  
[ 6.5, 2.8, 4.6, 1.5, 2. ],  
[ 5.7, 2.8, 4.5, 1.3, 2. ],  
[ 6.3, 3.3, 4.7, 1.6, 2. ],  
[ 4.9, 2.4, 3.3, 1. , 2. ],  
[ 6.6, 2.9, 4.6, 1.3, 2. ],  
[ 5.2, 2.7, 3.9, 1.4, 2. ],  
[ 5. , 2. , 3.5, 1. , 2. ],  
[ 5.9, 3. , 4.2, 1.5, 2. ],  
[ 6. , 2.2, 4. , 1. , 2. ],  
[ 6.1, 2.9, 4.7, 1.4, 2. ],  
[ 5.6, 2.9, 3.6, 1.3, 2. ],  
[ 6.7, 3.1, 4.4, 1.4, 2. ],  
[ 5.6, 3. , 4.5, 1.5, 2. ],  
[ 5.8, 2.7, 4.1, 1. , 2. ],  
[ 6.2, 2.2, 4.5, 1.5, 2. ],  
[ 5.6, 2.5, 3.9, 1.1, 2. ],  
[ 5.9, 3.2, 4.8, 1.8, 2. ],  
[ 6.1, 2.8, 4. , 1.3, 2. ],  
[ 6.3, 2.5, 4.9, 1.5, 2. ],  
[ 6.1, 2.8, 4.7, 1.2, 2. ],  
[ 6.4, 2.9, 4.3, 1.3, 2. ],  
[ 6.6, 3. , 4.4, 1.4, 2. ],  
[ 6.8, 2.8, 4.8, 1.4, 2. ],  
[ 6.7, 3. , 5. , 1.7, 2. ],  
[ 6. , 2.9, 4.5, 1.5, 2. ],  
[ 5.7, 2.6, 3.5, 1. , 2. ],  
[ 5.5, 2.4, 3.8, 1.1, 2. ],  
[ 5.5, 2.4, 3.7, 1. , 2. ],

[ 5.8, 2.7, 3.9, 1.2, 2. ],  
[ 6. , 2.7, 5.1, 1.6, 2. ],  
[ 5.4, 3. , 4.5, 1.5, 2. ],  
[ 6. , 3.4, 4.5, 1.6, 2. ],  
[ 6.7, 3.1, 4.7, 1.5, 2. ],  
[ 6.3, 2.3, 4.4, 1.3, 2. ],  
[ 5.6, 3. , 4.1, 1.3, 2. ],  
[ 5.5, 2.5, 4. , 1.3, 2. ],  
[ 5.5, 2.6, 4.4, 1.2, 2. ],  
[ 6.1, 3. , 4.6, 1.4, 2. ],  
[ 5.8, 2.6, 4. , 1.2, 2. ],  
[ 5. , 2.3, 3.3, 1. , 2. ],  
[ 5.6, 2.7, 4.2, 1.3, 2. ],  
[ 5.7, 3. , 4.2, 1.2, 2. ],  
[ 5.7, 2.9, 4.2, 1.3, 2. ],  
[ 6.2, 2.9, 4.3, 1.3, 2. ],  
[ 5.1, 2.5, 3. , 1.1, 2. ],  
[ 5.7, 2.8, 4.1, 1.3, 2. ],  
[ 6.3, 3.3, 6. , 2.5, 3. ],  
[ 5.8, 2.7, 5.1, 1.9, 3. ],  
[ 7.1, 3. , 5.9, 2.1, 3. ],  
[ 6.3, 2.9, 5.6, 1.8, 3. ],  
[ 6.5, 3. , 5.8, 2.2, 3. ],  
[ 7.6, 3. , 6.6, 2.1, 3. ],  
[ 4.9, 2.5, 4.5, 1.7, 3. ],  
[ 7.3, 2.9, 6.3, 1.8, 3. ],  
[ 6.7, 2.5, 5.8, 1.8, 3. ],  
[ 7.2, 3.6, 6.1, 2.5, 3. ],  
[ 6.5, 3.2, 5.1, 2. , 3. ],  
[ 6.4, 2.7, 5.3, 1.9, 3. ],  
[ 6.8, 3. , 5.5, 2.1, 3. ],  
[ 5.7, 2.5, 5. , 2. , 3. ],  
[ 5.8, 2.8, 5.1, 2.4, 3. ],  
[ 6.4, 3.2, 5.3, 2.3, 3. ],  
[ 6.5, 3. , 5.5, 1.8, 3. ],  
[ 7.7, 3.8, 6.7, 2.2, 3. ],  
[ 7.7, 2.6, 6.9, 2.3, 3. ],  
[ 6. , 2.2, 5. , 1.5, 3. ],  
[ 6.9, 3.2, 5.7, 2.3, 3. ],  
[ 5.6, 2.8, 4.9, 2. , 3. ],  
[ 7.7, 2.8, 6.7, 2. , 3. ],  
[ 6.3, 2.7, 4.9, 1.8, 3. ],  
[ 6.7, 3.3, 5.7, 2.1, 3. ],  
[ 7.2, 3.2, 6. , 1.8, 3. ],  
[ 6.2, 2.8, 4.8, 1.8, 3. ],  
[ 6.1, 3. , 4.9, 1.8, 3. ],  
[ 6.4, 2.8, 5.6, 2.1, 3. ],

```

[ 7.2,  3. ,  5.8,  1.6,  3. ],
[ 7.4,  2.8,  6.1,  1.9,  3. ],
[ 7.9,  3.8,  6.4,  2. ,  3. ],
[ 6.4,  2.8,  5.6,  2.2,  3. ],
[ 6.3,  2.8,  5.1,  1.5,  3. ],
[ 6.1,  2.6,  5.6,  1.4,  3. ],
[ 7.7,  3. ,  6.1,  2.3,  3. ],
[ 6.3,  3.4,  5.6,  2.4,  3. ],
[ 6.4,  3.1,  5.5,  1.8,  3. ],
[ 6. ,  3. ,  4.8,  1.8,  3. ],
[ 6.9,  3.1,  5.4,  2.1,  3. ],
[ 6.7,  3.1,  5.6,  2.4,  3. ],
[ 6.9,  3.1,  5.1,  2.3,  3. ],
[ 5.8,  2.7,  5.1,  1.9,  3. ],
[ 6.8,  3.2,  5.9,  2.3,  3. ],
[ 6.7,  3.3,  5.7,  2.5,  3. ],
[ 6.7,  3. ,  5.2,  2.3,  3. ],
[ 6.3,  2.5,  5. ,  1.9,  3. ],
[ 6.5,  3. ,  5.2,  2. ,  3. ],
[ 6.2,  3.4,  5.4,  2.3,  3. ],
[ 5.9,  3. ,  5.1,  1.8,  3. ],
[10. , 10. , 10. , 10. , 10. ]])

```

```
[50]: import numpy as np
np.insert(notre_tableau, 1, 42)
```

```
[50]: array([ 1, 42,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
          17, 18, 19, 20])
```

```
[51]: import numpy as np
np.insert(notre_tableau, 1, [42,43,44,45,46])
```

```
[51]: array([ 1, 42, 43, 44, 45, 46,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12,
          13, 14, 15, 16, 17, 18, 19, 20])
```

```
[52]: import numpy as np
np.insert(iris, 1, [10,10,10,10,10], axis = 0)
```

```
[52]: array([[ 5.1,  3.5,  1.4,  0.2,  1. ],
          [10. , 10. , 10. , 10. , 10. ],
          [ 4.9,  3. ,  1.4,  0.2,  1. ],
          [ 4.7,  3.2,  1.3,  0.2,  1. ],
          [ 4.6,  3.1,  1.5,  0.2,  1. ],
          [ 5. ,  3.6,  1.4,  0.2,  1. ],
          [ 5.4,  3.9,  1.7,  0.4,  1. ],
          [ 4.6,  3.4,  1.4,  0.3,  1. ],
          [ 5. ,  3.4,  1.5,  0.2,  1. ],
```

[ 4.4, 2.9, 1.4, 0.2, 1. ],  
[ 4.9, 3.1, 1.5, 0.1, 1. ],  
[ 5.4, 3.7, 1.5, 0.2, 1. ],  
[ 4.8, 3.4, 1.6, 0.2, 1. ],  
[ 4.8, 3. , 1.4, 0.1, 1. ],  
[ 4.3, 3. , 1.1, 0.1, 1. ],  
[ 5.8, 4. , 1.2, 0.2, 1. ],  
[ 5.7, 4.4, 1.5, 0.4, 1. ],  
[ 5.4, 3.9, 1.3, 0.4, 1. ],  
[ 5.1, 3.5, 1.4, 0.3, 1. ],  
[ 5.7, 3.8, 1.7, 0.3, 1. ],  
[ 5.1, 3.8, 1.5, 0.3, 1. ],  
[ 5.4, 3.4, 1.7, 0.2, 1. ],  
[ 5.1, 3.7, 1.5, 0.4, 1. ],  
[ 4.6, 3.6, 1. , 0.2, 1. ],  
[ 5.1, 3.3, 1.7, 0.5, 1. ],  
[ 4.8, 3.4, 1.9, 0.2, 1. ],  
[ 5. , 3. , 1.6, 0.2, 1. ],  
[ 5. , 3.4, 1.6, 0.4, 1. ],  
[ 5.2, 3.5, 1.5, 0.2, 1. ],  
[ 5.2, 3.4, 1.4, 0.2, 1. ],  
[ 4.7, 3.2, 1.6, 0.2, 1. ],  
[ 4.8, 3.1, 1.6, 0.2, 1. ],  
[ 5.4, 3.4, 1.5, 0.4, 1. ],  
[ 5.2, 4.1, 1.5, 0.1, 1. ],  
[ 5.5, 4.2, 1.4, 0.2, 1. ],  
[ 4.9, 3.1, 1.5, 0.2, 1. ],  
[ 5. , 3.2, 1.2, 0.2, 1. ],  
[ 5.5, 3.5, 1.3, 0.2, 1. ],  
[ 4.9, 3.6, 1.4, 0.1, 1. ],  
[ 4.4, 3. , 1.3, 0.2, 1. ],  
[ 5.1, 3.4, 1.5, 0.2, 1. ],  
[ 5. , 3.5, 1.3, 0.3, 1. ],  
[ 4.5, 2.3, 1.3, 0.3, 1. ],  
[ 4.4, 3.2, 1.3, 0.2, 1. ],  
[ 5. , 3.5, 1.6, 0.6, 1. ],  
[ 5.1, 3.8, 1.9, 0.4, 1. ],  
[ 4.8, 3. , 1.4, 0.3, 1. ],  
[ 5.1, 3.8, 1.6, 0.2, 1. ],  
[ 4.6, 3.2, 1.4, 0.2, 1. ],  
[ 5.3, 3.7, 1.5, 0.2, 1. ],  
[ 5. , 3.3, 1.4, 0.2, 1. ],  
[ 7. , 3.2, 4.7, 1.4, 2. ],  
[ 6.4, 3.2, 4.5, 1.5, 2. ],  
[ 6.9, 3.1, 4.9, 1.5, 2. ],  
[ 5.5, 2.3, 4. , 1.3, 2. ],  
[ 6.5, 2.8, 4.6, 1.5, 2. ],



[ 5.7, 2.8, 4.5, 1.3, 2. ],  
[ 6.3, 3.3, 4.7, 1.6, 2. ],  
[ 4.9, 2.4, 3.3, 1. , 2. ],  
[ 6.6, 2.9, 4.6, 1.3, 2. ],  
[ 5.2, 2.7, 3.9, 1.4, 2. ],  
[ 5. , 2. , 3.5, 1. , 2. ],  
[ 5.9, 3. , 4.2, 1.5, 2. ],  
[ 6. , 2.2, 4. , 1. , 2. ],  
[ 6.1, 2.9, 4.7, 1.4, 2. ],  
[ 5.6, 2.9, 3.6, 1.3, 2. ],  
[ 6.7, 3.1, 4.4, 1.4, 2. ],  
[ 5.6, 3. , 4.5, 1.5, 2. ],  
[ 5.8, 2.7, 4.1, 1. , 2. ],  
[ 6.2, 2.2, 4.5, 1.5, 2. ],  
[ 5.6, 2.5, 3.9, 1.1, 2. ],  
[ 5.9, 3.2, 4.8, 1.8, 2. ],  
[ 6.1, 2.8, 4. , 1.3, 2. ],  
[ 6.3, 2.5, 4.9, 1.5, 2. ],  
[ 6.1, 2.8, 4.7, 1.2, 2. ],  
[ 6.4, 2.9, 4.3, 1.3, 2. ],  
[ 6.6, 3. , 4.4, 1.4, 2. ],  
[ 6.8, 2.8, 4.8, 1.4, 2. ],  
[ 6.7, 3. , 5. , 1.7, 2. ],  
[ 6. , 2.9, 4.5, 1.5, 2. ],  
[ 5.7, 2.6, 3.5, 1. , 2. ],  
[ 5.5, 2.4, 3.8, 1.1, 2. ],  
[ 5.5, 2.4, 3.7, 1. , 2. ],  
[ 5.8, 2.7, 3.9, 1.2, 2. ],  
[ 6. , 2.7, 5.1, 1.6, 2. ],  
[ 5.4, 3. , 4.5, 1.5, 2. ],  
[ 6. , 3.4, 4.5, 1.6, 2. ],  
[ 6.7, 3.1, 4.7, 1.5, 2. ],  
[ 6.3, 2.3, 4.4, 1.3, 2. ],  
[ 5.6, 3. , 4.1, 1.3, 2. ],  
[ 5.5, 2.5, 4. , 1.3, 2. ],  
[ 5.5, 2.6, 4.4, 1.2, 2. ],  
[ 6.1, 3. , 4.6, 1.4, 2. ],  
[ 5.8, 2.6, 4. , 1.2, 2. ],  
[ 5. , 2.3, 3.3, 1. , 2. ],  
[ 5.6, 2.7, 4.2, 1.3, 2. ],  
[ 5.7, 3. , 4.2, 1.2, 2. ],  
[ 5.7, 2.9, 4.2, 1.3, 2. ],  
[ 6.2, 2.9, 4.3, 1.3, 2. ],  
[ 5.1, 2.5, 3. , 1.1, 2. ],  
[ 5.7, 2.8, 4.1, 1.3, 2. ],  
[ 6.3, 3.3, 6. , 2.5, 3. ],  
[ 5.8, 2.7, 5.1, 1.9, 3. ],

[ 7.1, 3. , 5.9, 2.1, 3. ],  
[ 6.3, 2.9, 5.6, 1.8, 3. ],  
[ 6.5, 3. , 5.8, 2.2, 3. ],  
[ 7.6, 3. , 6.6, 2.1, 3. ],  
[ 4.9, 2.5, 4.5, 1.7, 3. ],  
[ 7.3, 2.9, 6.3, 1.8, 3. ],  
[ 6.7, 2.5, 5.8, 1.8, 3. ],  
[ 7.2, 3.6, 6.1, 2.5, 3. ],  
[ 6.5, 3.2, 5.1, 2. , 3. ],  
[ 6.4, 2.7, 5.3, 1.9, 3. ],  
[ 6.8, 3. , 5.5, 2.1, 3. ],  
[ 5.7, 2.5, 5. , 2. , 3. ],  
[ 5.8, 2.8, 5.1, 2.4, 3. ],  
[ 6.4, 3.2, 5.3, 2.3, 3. ],  
[ 6.5, 3. , 5.5, 1.8, 3. ],  
[ 7.7, 3.8, 6.7, 2.2, 3. ],  
[ 7.7, 2.6, 6.9, 2.3, 3. ],  
[ 6. , 2.2, 5. , 1.5, 3. ],  
[ 6.9, 3.2, 5.7, 2.3, 3. ],  
[ 5.6, 2.8, 4.9, 2. , 3. ],  
[ 7.7, 2.8, 6.7, 2. , 3. ],  
[ 6.3, 2.7, 4.9, 1.8, 3. ],  
[ 6.7, 3.3, 5.7, 2.1, 3. ],  
[ 7.2, 3.2, 6. , 1.8, 3. ],  
[ 6.2, 2.8, 4.8, 1.8, 3. ],  
[ 6.1, 3. , 4.9, 1.8, 3. ],  
[ 6.4, 2.8, 5.6, 2.1, 3. ],  
[ 7.2, 3. , 5.8, 1.6, 3. ],  
[ 7.4, 2.8, 6.1, 1.9, 3. ],  
[ 7.9, 3.8, 6.4, 2. , 3. ],  
[ 6.4, 2.8, 5.6, 2.2, 3. ],  
[ 6.3, 2.8, 5.1, 1.5, 3. ],  
[ 6.1, 2.6, 5.6, 1.4, 3. ],  
[ 7.7, 3. , 6.1, 2.3, 3. ],  
[ 6.3, 3.4, 5.6, 2.4, 3. ],  
[ 6.4, 3.1, 5.5, 1.8, 3. ],  
[ 6. , 3. , 4.8, 1.8, 3. ],  
[ 6.9, 3.1, 5.4, 2.1, 3. ],  
[ 6.7, 3.1, 5.6, 2.4, 3. ],  
[ 6.9, 3.1, 5.1, 2.3, 3. ],  
[ 5.8, 2.7, 5.1, 1.9, 3. ],  
[ 6.8, 3.2, 5.9, 2.3, 3. ],  
[ 6.7, 3.3, 5.7, 2.5, 3. ],  
[ 6.7, 3. , 5.2, 2.3, 3. ],  
[ 6.3, 2.5, 5. , 1.9, 3. ],  
[ 6.5, 3. , 5.2, 2. , 3. ],  
[ 6.2, 3.4, 5.4, 2.3, 3. ],

```
[ 5.9, 3. , 5.1, 1.8, 3. ]])
```

```
[53]: import numpy as np
      np.insert(iris, 1, [10,10,10,10,10])
```

```
[53]: array([[ 5.1, 10. , 10. , 10. , 10. , 10. , 3.5, 1.4, 0.2, 1. , 4.9,
           3. , 1.4, 0.2, 1. , 4.7, 3.2, 1.3, 0.2, 1. , 4.6, 3.1,
           1.5, 0.2, 1. , 5. , 3.6, 1.4, 0.2, 1. , 5.4, 3.9, 1.7,
           0.4, 1. , 4.6, 3.4, 1.4, 0.3, 1. , 5. , 3.4, 1.5, 0.2,
           1. , 4.4, 2.9, 1.4, 0.2, 1. , 4.9, 3.1, 1.5, 0.1, 1. ,
           5.4, 3.7, 1.5, 0.2, 1. , 4.8, 3.4, 1.6, 0.2, 1. , 4.8,
           3. , 1.4, 0.1, 1. , 4.3, 3. , 1.1, 0.1, 1. , 5.8, 4. ,
           1.2, 0.2, 1. , 5.7, 4.4, 1.5, 0.4, 1. , 5.4, 3.9, 1.3,
           0.4, 1. , 5.1, 3.5, 1.4, 0.3, 1. , 5.7, 3.8, 1.7, 0.3,
           1. , 5.1, 3.8, 1.5, 0.3, 1. , 5.4, 3.4, 1.7, 0.2, 1. ,
           5.1, 3.7, 1.5, 0.4, 1. , 4.6, 3.6, 1. , 0.2, 1. , 5.1,
           3.3, 1.7, 0.5, 1. , 4.8, 3.4, 1.9, 0.2, 1. , 5. , 3. ,
           1.6, 0.2, 1. , 5. , 3.4, 1.6, 0.4, 1. , 5.2, 3.5, 1.5,
           0.2, 1. , 5.2, 3.4, 1.4, 0.2, 1. , 4.7, 3.2, 1.6, 0.2,
           1. , 4.8, 3.1, 1.6, 0.2, 1. , 5.4, 3.4, 1.5, 0.4, 1. ,
           5.2, 4.1, 1.5, 0.1, 1. , 5.5, 4.2, 1.4, 0.2, 1. , 4.9,
           3.1, 1.5, 0.2, 1. , 5. , 3.2, 1.2, 0.2, 1. , 5.5, 3.5,
           1.3, 0.2, 1. , 4.9, 3.6, 1.4, 0.1, 1. , 4.4, 3. , 1.3,
           0.2, 1. , 5.1, 3.4, 1.5, 0.2, 1. , 5. , 3.5, 1.3, 0.3,
           1. , 4.5, 2.3, 1.3, 0.3, 1. , 4.4, 3.2, 1.3, 0.2, 1. ,
           5. , 3.5, 1.6, 0.6, 1. , 5.1, 3.8, 1.9, 0.4, 1. , 4.8,
           3. , 1.4, 0.3, 1. , 5.1, 3.8, 1.6, 0.2, 1. , 4.6, 3.2,
           1.4, 0.2, 1. , 5.3, 3.7, 1.5, 0.2, 1. , 5. , 3.3, 1.4,
           0.2, 1. , 7. , 3.2, 4.7, 1.4, 2. , 6.4, 3.2, 4.5, 1.5,
           2. , 6.9, 3.1, 4.9, 1.5, 2. , 5.5, 2.3, 4. , 1.3, 2. ,
           6.5, 2.8, 4.6, 1.5, 2. , 5.7, 2.8, 4.5, 1.3, 2. , 6.3,
           3.3, 4.7, 1.6, 2. , 4.9, 2.4, 3.3, 1. , 2. , 6.6, 2.9,
           4.6, 1.3, 2. , 5.2, 2.7, 3.9, 1.4, 2. , 5. , 2. , 3.5,
           1. , 2. , 5.9, 3. , 4.2, 1.5, 2. , 6. , 2.2, 4. , 1. ,
           2. , 6.1, 2.9, 4.7, 1.4, 2. , 5.6, 2.9, 3.6, 1.3, 2. ,
           6.7, 3.1, 4.4, 1.4, 2. , 5.6, 3. , 4.5, 1.5, 2. , 5.8,
           2.7, 4.1, 1. , 2. , 6.2, 2.2, 4.5, 1.5, 2. , 5.6, 2.5,
           3.9, 1.1, 2. , 5.9, 3.2, 4.8, 1.8, 2. , 6.1, 2.8, 4. ,
           1.3, 2. , 6.3, 2.5, 4.9, 1.5, 2. , 6.1, 2.8, 4.7, 1.2,
           2. , 6.4, 2.9, 4.3, 1.3, 2. , 6.6, 3. , 4.4, 1.4, 2. ,
           6.8, 2.8, 4.8, 1.4, 2. , 6.7, 3. , 5. , 1.7, 2. , 6. ,
           2.9, 4.5, 1.5, 2. , 5.7, 2.6, 3.5, 1. , 2. , 5.5, 2.4,
           3.8, 1.1, 2. , 5.5, 2.4, 3.7, 1. , 2. , 5.8, 2.7, 3.9,
           1.2, 2. , 6. , 2.7, 5.1, 1.6, 2. , 5.4, 3. , 4.5, 1.5,
           2. , 6. , 3.4, 4.5, 1.6, 2. , 6.7, 3.1, 4.7, 1.5, 2. ,
           6.3, 2.3, 4.4, 1.3, 2. , 5.6, 3. , 4.1, 1.3, 2. , 5.5,
           2.5, 4. , 1.3, 2. , 5.5, 2.6, 4.4, 1.2, 2. , 6.1, 3. ,
```

```

4.6, 1.4, 2. , 5.8, 2.6, 4. , 1.2, 2. , 5. , 2.3, 3.3,
1. , 2. , 5.6, 2.7, 4.2, 1.3, 2. , 5.7, 3. , 4.2, 1.2,
2. , 5.7, 2.9, 4.2, 1.3, 2. , 6.2, 2.9, 4.3, 1.3, 2. ,
5.1, 2.5, 3. , 1.1, 2. , 5.7, 2.8, 4.1, 1.3, 2. , 6.3,
3.3, 6. , 2.5, 3. , 5.8, 2.7, 5.1, 1.9, 3. , 7.1, 3. ,
5.9, 2.1, 3. , 6.3, 2.9, 5.6, 1.8, 3. , 6.5, 3. , 5.8,
2.2, 3. , 7.6, 3. , 6.6, 2.1, 3. , 4.9, 2.5, 4.5, 1.7,
3. , 7.3, 2.9, 6.3, 1.8, 3. , 6.7, 2.5, 5.8, 1.8, 3. ,
7.2, 3.6, 6.1, 2.5, 3. , 6.5, 3.2, 5.1, 2. , 3. , 6.4,
2.7, 5.3, 1.9, 3. , 6.8, 3. , 5.5, 2.1, 3. , 5.7, 2.5,
5. , 2. , 3. , 5.8, 2.8, 5.1, 2.4, 3. , 6.4, 3.2, 5.3,
2.3, 3. , 6.5, 3. , 5.5, 1.8, 3. , 7.7, 3.8, 6.7, 2.2,
3. , 7.7, 2.6, 6.9, 2.3, 3. , 6. , 2.2, 5. , 1.5, 3. ,
6.9, 3.2, 5.7, 2.3, 3. , 5.6, 2.8, 4.9, 2. , 3. , 7.7,
2.8, 6.7, 2. , 3. , 6.3, 2.7, 4.9, 1.8, 3. , 6.7, 3.3,
5.7, 2.1, 3. , 7.2, 3.2, 6. , 1.8, 3. , 6.2, 2.8, 4.8,
1.8, 3. , 6.1, 3. , 4.9, 1.8, 3. , 6.4, 2.8, 5.6, 2.1,
3. , 7.2, 3. , 5.8, 1.6, 3. , 7.4, 2.8, 6.1, 1.9, 3. ,
7.9, 3.8, 6.4, 2. , 3. , 6.4, 2.8, 5.6, 2.2, 3. , 6.3,
2.8, 5.1, 1.5, 3. , 6.1, 2.6, 5.6, 1.4, 3. , 7.7, 3. ,
6.1, 2.3, 3. , 6.3, 3.4, 5.6, 2.4, 3. , 6.4, 3.1, 5.5,
1.8, 3. , 6. , 3. , 4.8, 1.8, 3. , 6.9, 3.1, 5.4, 2.1,
3. , 6.7, 3.1, 5.6, 2.4, 3. , 6.9, 3.1, 5.1, 2.3, 3. ,
5.8, 2.7, 5.1, 1.9, 3. , 6.8, 3.2, 5.9, 2.3, 3. , 6.7,
3.3, 5.7, 2.5, 3. , 6.7, 3. , 5.2, 2.3, 3. , 6.3, 2.5,
5. , 1.9, 3. , 6.5, 3. , 5.2, 2. , 3. , 6.2, 3.4, 5.4,
2.3, 3. , 5.9, 3. , 5.1, 1.8, 3. ])

```

#### 4.1.2 5.1.2. Supprimer des éléments d'un tableau

```
[54]: np.delete(iris, 1, axis = 1)
```

```
[54]: array([[5.1, 1.4, 0.2, 1. ],
 [4.9, 1.4, 0.2, 1. ],
 [4.7, 1.3, 0.2, 1. ],
 [4.6, 1.5, 0.2, 1. ],
 [5. , 1.4, 0.2, 1. ],
 [5.4, 1.7, 0.4, 1. ],
 [4.6, 1.4, 0.3, 1. ],
 [5. , 1.5, 0.2, 1. ],
 [4.4, 1.4, 0.2, 1. ],
 [4.9, 1.5, 0.1, 1. ],
 [5.4, 1.5, 0.2, 1. ],
 [4.8, 1.6, 0.2, 1. ],
 [4.8, 1.4, 0.1, 1. ],
 [4.3, 1.1, 0.1, 1. ],
 [5.8, 1.2, 0.2, 1. ],
```

[5.7, 1.5, 0.4, 1. ],  
[5.4, 1.3, 0.4, 1. ],  
[5.1, 1.4, 0.3, 1. ],  
[5.7, 1.7, 0.3, 1. ],  
[5.1, 1.5, 0.3, 1. ],  
[5.4, 1.7, 0.2, 1. ],  
[5.1, 1.5, 0.4, 1. ],  
[4.6, 1. , 0.2, 1. ],  
[5.1, 1.7, 0.5, 1. ],  
[4.8, 1.9, 0.2, 1. ],  
[5. , 1.6, 0.2, 1. ],  
[5. , 1.6, 0.4, 1. ],  
[5.2, 1.5, 0.2, 1. ],  
[5.2, 1.4, 0.2, 1. ],  
[4.7, 1.6, 0.2, 1. ],  
[4.8, 1.6, 0.2, 1. ],  
[5.4, 1.5, 0.4, 1. ],  
[5.2, 1.5, 0.1, 1. ],  
[5.5, 1.4, 0.2, 1. ],  
[4.9, 1.5, 0.2, 1. ],  
[5. , 1.2, 0.2, 1. ],  
[5.5, 1.3, 0.2, 1. ],  
[4.9, 1.4, 0.1, 1. ],  
[4.4, 1.3, 0.2, 1. ],  
[5.1, 1.5, 0.2, 1. ],  
[5. , 1.3, 0.3, 1. ],  
[4.5, 1.3, 0.3, 1. ],  
[4.4, 1.3, 0.2, 1. ],  
[5. , 1.6, 0.6, 1. ],  
[5.1, 1.9, 0.4, 1. ],  
[4.8, 1.4, 0.3, 1. ],  
[5.1, 1.6, 0.2, 1. ],  
[4.6, 1.4, 0.2, 1. ],  
[5.3, 1.5, 0.2, 1. ],  
[5. , 1.4, 0.2, 1. ],  
[7. , 4.7, 1.4, 2. ],  
[6.4, 4.5, 1.5, 2. ],  
[6.9, 4.9, 1.5, 2. ],  
[5.5, 4. , 1.3, 2. ],  
[6.5, 4.6, 1.5, 2. ],  
[5.7, 4.5, 1.3, 2. ],  
[6.3, 4.7, 1.6, 2. ],  
[4.9, 3.3, 1. , 2. ],  
[6.6, 4.6, 1.3, 2. ],  
[5.2, 3.9, 1.4, 2. ],  
[5. , 3.5, 1. , 2. ],  
[5.9, 4.2, 1.5, 2. ],

[6. , 4. , 1. , 2. ],  
[6.1, 4.7, 1.4, 2. ],  
[5.6, 3.6, 1.3, 2. ],  
[6.7, 4.4, 1.4, 2. ],  
[5.6, 4.5, 1.5, 2. ],  
[5.8, 4.1, 1. , 2. ],  
[6.2, 4.5, 1.5, 2. ],  
[5.6, 3.9, 1.1, 2. ],  
[5.9, 4.8, 1.8, 2. ],  
[6.1, 4. , 1.3, 2. ],  
[6.3, 4.9, 1.5, 2. ],  
[6.1, 4.7, 1.2, 2. ],  
[6.4, 4.3, 1.3, 2. ],  
[6.6, 4.4, 1.4, 2. ],  
[6.8, 4.8, 1.4, 2. ],  
[6.7, 5. , 1.7, 2. ],  
[6. , 4.5, 1.5, 2. ],  
[5.7, 3.5, 1. , 2. ],  
[5.5, 3.8, 1.1, 2. ],  
[5.5, 3.7, 1. , 2. ],  
[5.8, 3.9, 1.2, 2. ],  
[6. , 5.1, 1.6, 2. ],  
[5.4, 4.5, 1.5, 2. ],  
[6. , 4.5, 1.6, 2. ],  
[6.7, 4.7, 1.5, 2. ],  
[6.3, 4.4, 1.3, 2. ],  
[5.6, 4.1, 1.3, 2. ],  
[5.5, 4. , 1.3, 2. ],  
[5.5, 4.4, 1.2, 2. ],  
[6.1, 4.6, 1.4, 2. ],  
[5.8, 4. , 1.2, 2. ],  
[5. , 3.3, 1. , 2. ],  
[5.6, 4.2, 1.3, 2. ],  
[5.7, 4.2, 1.2, 2. ],  
[5.7, 4.2, 1.3, 2. ],  
[6.2, 4.3, 1.3, 2. ],  
[5.1, 3. , 1.1, 2. ],  
[5.7, 4.1, 1.3, 2. ],  
[6.3, 6. , 2.5, 3. ],  
[5.8, 5.1, 1.9, 3. ],  
[7.1, 5.9, 2.1, 3. ],  
[6.3, 5.6, 1.8, 3. ],  
[6.5, 5.8, 2.2, 3. ],  
[7.6, 6.6, 2.1, 3. ],  
[4.9, 4.5, 1.7, 3. ],  
[7.3, 6.3, 1.8, 3. ],  
[6.7, 5.8, 1.8, 3. ],

```
[7.2, 6.1, 2.5, 3. ],
[6.5, 5.1, 2. , 3. ],
[6.4, 5.3, 1.9, 3. ],
[6.8, 5.5, 2.1, 3. ],
[5.7, 5. , 2. , 3. ],
[5.8, 5.1, 2.4, 3. ],
[6.4, 5.3, 2.3, 3. ],
[6.5, 5.5, 1.8, 3. ],
[7.7, 6.7, 2.2, 3. ],
[7.7, 6.9, 2.3, 3. ],
[6. , 5. , 1.5, 3. ],
[6.9, 5.7, 2.3, 3. ],
[5.6, 4.9, 2. , 3. ],
[7.7, 6.7, 2. , 3. ],
[6.3, 4.9, 1.8, 3. ],
[6.7, 5.7, 2.1, 3. ],
[7.2, 6. , 1.8, 3. ],
[6.2, 4.8, 1.8, 3. ],
[6.1, 4.9, 1.8, 3. ],
[6.4, 5.6, 2.1, 3. ],
[7.2, 5.8, 1.6, 3. ],
[7.4, 6.1, 1.9, 3. ],
[7.9, 6.4, 2. , 3. ],
[6.4, 5.6, 2.2, 3. ],
[6.3, 5.1, 1.5, 3. ],
[6.1, 5.6, 1.4, 3. ],
[7.7, 6.1, 2.3, 3. ],
[6.3, 5.6, 2.4, 3. ],
[6.4, 5.5, 1.8, 3. ],
[6. , 4.8, 1.8, 3. ],
[6.9, 5.4, 2.1, 3. ],
[6.7, 5.6, 2.4, 3. ],
[6.9, 5.1, 2.3, 3. ],
[5.8, 5.1, 1.9, 3. ],
[6.8, 5.9, 2.3, 3. ],
[6.7, 5.7, 2.5, 3. ],
[6.7, 5.2, 2.3, 3. ],
[6.3, 5. , 1.9, 3. ],
[6.5, 5.2, 2. , 3. ],
[6.2, 5.4, 2.3, 3. ],
[5.9, 5.1, 1.8, 3. ]])
```

```
[55]: np.delete(iris, [1,2,3,4], axis = 0)
```

```
[55]: array([[5.1, 3.5, 1.4, 0.2, 1. ],
            [5.4, 3.9, 1.7, 0.4, 1. ],
            [4.6, 3.4, 1.4, 0.3, 1. ],
```

[5. , 3.4, 1.5, 0.2, 1. ],  
[4.4, 2.9, 1.4, 0.2, 1. ],  
[4.9, 3.1, 1.5, 0.1, 1. ],  
[5.4, 3.7, 1.5, 0.2, 1. ],  
[4.8, 3.4, 1.6, 0.2, 1. ],  
[4.8, 3. , 1.4, 0.1, 1. ],  
[4.3, 3. , 1.1, 0.1, 1. ],  
[5.8, 4. , 1.2, 0.2, 1. ],  
[5.7, 4.4, 1.5, 0.4, 1. ],  
[5.4, 3.9, 1.3, 0.4, 1. ],  
[5.1, 3.5, 1.4, 0.3, 1. ],  
[5.7, 3.8, 1.7, 0.3, 1. ],  
[5.1, 3.8, 1.5, 0.3, 1. ],  
[5.4, 3.4, 1.7, 0.2, 1. ],  
[5.1, 3.7, 1.5, 0.4, 1. ],  
[4.6, 3.6, 1. , 0.2, 1. ],  
[5.1, 3.3, 1.7, 0.5, 1. ],  
[4.8, 3.4, 1.9, 0.2, 1. ],  
[5. , 3. , 1.6, 0.2, 1. ],  
[5. , 3.4, 1.6, 0.4, 1. ],  
[5.2, 3.5, 1.5, 0.2, 1. ],  
[5.2, 3.4, 1.4, 0.2, 1. ],  
[4.7, 3.2, 1.6, 0.2, 1. ],  
[4.8, 3.1, 1.6, 0.2, 1. ],  
[5.4, 3.4, 1.5, 0.4, 1. ],  
[5.2, 4.1, 1.5, 0.1, 1. ],  
[5.5, 4.2, 1.4, 0.2, 1. ],  
[4.9, 3.1, 1.5, 0.2, 1. ],  
[5. , 3.2, 1.2, 0.2, 1. ],  
[5.5, 3.5, 1.3, 0.2, 1. ],  
[4.9, 3.6, 1.4, 0.1, 1. ],  
[4.4, 3. , 1.3, 0.2, 1. ],  
[5.1, 3.4, 1.5, 0.2, 1. ],  
[5. , 3.5, 1.3, 0.3, 1. ],  
[4.5, 2.3, 1.3, 0.3, 1. ],  
[4.4, 3.2, 1.3, 0.2, 1. ],  
[5. , 3.5, 1.6, 0.6, 1. ],  
[5.1, 3.8, 1.9, 0.4, 1. ],  
[4.8, 3. , 1.4, 0.3, 1. ],  
[5.1, 3.8, 1.6, 0.2, 1. ],  
[4.6, 3.2, 1.4, 0.2, 1. ],  
[5.3, 3.7, 1.5, 0.2, 1. ],  
[5. , 3.3, 1.4, 0.2, 1. ],  
[7. , 3.2, 4.7, 1.4, 2. ],  
[6.4, 3.2, 4.5, 1.5, 2. ],  
[6.9, 3.1, 4.9, 1.5, 2. ],  
[5.5, 2.3, 4. , 1.3, 2. ],



[6.5, 2.8, 4.6, 1.5, 2. ],  
[5.7, 2.8, 4.5, 1.3, 2. ],  
[6.3, 3.3, 4.7, 1.6, 2. ],  
[4.9, 2.4, 3.3, 1. , 2. ],  
[6.6, 2.9, 4.6, 1.3, 2. ],  
[5.2, 2.7, 3.9, 1.4, 2. ],  
[5. , 2. , 3.5, 1. , 2. ],  
[5.9, 3. , 4.2, 1.5, 2. ],  
[6. , 2.2, 4. , 1. , 2. ],  
[6.1, 2.9, 4.7, 1.4, 2. ],  
[5.6, 2.9, 3.6, 1.3, 2. ],  
[6.7, 3.1, 4.4, 1.4, 2. ],  
[5.6, 3. , 4.5, 1.5, 2. ],  
[5.8, 2.7, 4.1, 1. , 2. ],  
[6.2, 2.2, 4.5, 1.5, 2. ],  
[5.6, 2.5, 3.9, 1.1, 2. ],  
[5.9, 3.2, 4.8, 1.8, 2. ],  
[6.1, 2.8, 4. , 1.3, 2. ],  
[6.3, 2.5, 4.9, 1.5, 2. ],  
[6.1, 2.8, 4.7, 1.2, 2. ],  
[6.4, 2.9, 4.3, 1.3, 2. ],  
[6.6, 3. , 4.4, 1.4, 2. ],  
[6.8, 2.8, 4.8, 1.4, 2. ],  
[6.7, 3. , 5. , 1.7, 2. ],  
[6. , 2.9, 4.5, 1.5, 2. ],  
[5.7, 2.6, 3.5, 1. , 2. ],  
[5.5, 2.4, 3.8, 1.1, 2. ],  
[5.5, 2.4, 3.7, 1. , 2. ],  
[5.8, 2.7, 3.9, 1.2, 2. ],  
[6. , 2.7, 5.1, 1.6, 2. ],  
[5.4, 3. , 4.5, 1.5, 2. ],  
[6. , 3.4, 4.5, 1.6, 2. ],  
[6.7, 3.1, 4.7, 1.5, 2. ],  
[6.3, 2.3, 4.4, 1.3, 2. ],  
[5.6, 3. , 4.1, 1.3, 2. ],  
[5.5, 2.5, 4. , 1.3, 2. ],  
[5.5, 2.6, 4.4, 1.2, 2. ],  
[6.1, 3. , 4.6, 1.4, 2. ],  
[5.8, 2.6, 4. , 1.2, 2. ],  
[5. , 2.3, 3.3, 1. , 2. ],  
[5.6, 2.7, 4.2, 1.3, 2. ],  
[5.7, 3. , 4.2, 1.2, 2. ],  
[5.7, 2.9, 4.2, 1.3, 2. ],  
[6.2, 2.9, 4.3, 1.3, 2. ],  
[5.1, 2.5, 3. , 1.1, 2. ],  
[5.7, 2.8, 4.1, 1.3, 2. ],  
[6.3, 3.3, 6. , 2.5, 3. ],

[5.8, 2.7, 5.1, 1.9, 3. ],  
[7.1, 3. , 5.9, 2.1, 3. ],  
[6.3, 2.9, 5.6, 1.8, 3. ],  
[6.5, 3. , 5.8, 2.2, 3. ],  
[7.6, 3. , 6.6, 2.1, 3. ],  
[4.9, 2.5, 4.5, 1.7, 3. ],  
[7.3, 2.9, 6.3, 1.8, 3. ],  
[6.7, 2.5, 5.8, 1.8, 3. ],  
[7.2, 3.6, 6.1, 2.5, 3. ],  
[6.5, 3.2, 5.1, 2. , 3. ],  
[6.4, 2.7, 5.3, 1.9, 3. ],  
[6.8, 3. , 5.5, 2.1, 3. ],  
[5.7, 2.5, 5. , 2. , 3. ],  
[5.8, 2.8, 5.1, 2.4, 3. ],  
[6.4, 3.2, 5.3, 2.3, 3. ],  
[6.5, 3. , 5.5, 1.8, 3. ],  
[7.7, 3.8, 6.7, 2.2, 3. ],  
[7.7, 2.6, 6.9, 2.3, 3. ],  
[6. , 2.2, 5. , 1.5, 3. ],  
[6.9, 3.2, 5.7, 2.3, 3. ],  
[5.6, 2.8, 4.9, 2. , 3. ],  
[7.7, 2.8, 6.7, 2. , 3. ],  
[6.3, 2.7, 4.9, 1.8, 3. ],  
[6.7, 3.3, 5.7, 2.1, 3. ],  
[7.2, 3.2, 6. , 1.8, 3. ],  
[6.2, 2.8, 4.8, 1.8, 3. ],  
[6.1, 3. , 4.9, 1.8, 3. ],  
[6.4, 2.8, 5.6, 2.1, 3. ],  
[7.2, 3. , 5.8, 1.6, 3. ],  
[7.4, 2.8, 6.1, 1.9, 3. ],  
[7.9, 3.8, 6.4, 2. , 3. ],  
[6.4, 2.8, 5.6, 2.2, 3. ],  
[6.3, 2.8, 5.1, 1.5, 3. ],  
[6.1, 2.6, 5.6, 1.4, 3. ],  
[7.7, 3. , 6.1, 2.3, 3. ],  
[6.3, 3.4, 5.6, 2.4, 3. ],  
[6.4, 3.1, 5.5, 1.8, 3. ],  
[6. , 3. , 4.8, 1.8, 3. ],  
[6.9, 3.1, 5.4, 2.1, 3. ],  
[6.7, 3.1, 5.6, 2.4, 3. ],  
[6.9, 3.1, 5.1, 2.3, 3. ],  
[5.8, 2.7, 5.1, 1.9, 3. ],  
[6.8, 3.2, 5.9, 2.3, 3. ],  
[6.7, 3.3, 5.7, 2.5, 3. ],  
[6.7, 3. , 5.2, 2.3, 3. ],  
[6.3, 2.5, 5. , 1.9, 3. ],  
[6.5, 3. , 5.2, 2. , 3. ],

```
[6.2, 3.4, 5.4, 2.3, 3. ],  
[5.9, 3. , 5.1, 1.8, 3. ]])
```

```
[56]: np.delete(notre_tableau,1)
```

```
[56]: array([ 1,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17, 18,  
          19, 20])
```

## 4.2 5.2. Diviser un tableau NumPy (split, hsplit et vsplit)

### 4.2.1 5.2.1. Sur un tableau à une dimension

```
[57]: notre_tableau
```

```
[57]: array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17,  
          18, 19, 20])
```

```
[58]: import numpy as np  
notre_tableau1, notre_tableau2, notre_tableau3 = np.split(notre_tableau, [2, 6])  
print(notre_tableau1, notre_tableau2, notre_tableau3)
```

```
[1 2] [3 4 5 6] [ 7  8  9 10 11 12 13 14 15 16 17 18 19 20]
```

### 4.2.2 5.2.2. Sur un tableau à deux dimensions

```
[59]: array=np.array([[2,5,6],[9,10,11],[5,6,9]], dtype="float")  
print("Avant division")  
print(array)  
array1, array2 = np.vsplit(array, [2])  
print("Après division")  
print(array1)  
print(array2)
```

```
Avant division  
[[ 2.  5.  6.]  
 [ 9. 10. 11.]  
 [ 5.  6.  9.]]  
Après division  
[[ 2.  5.  6.]  
 [ 9. 10. 11.]]  
[[5. 6. 9.]]
```

```
[60]: array= np.array([[2,5,6],[9,10,11],[5,6,9]], dtype="float")  
print("Avant division")  
print(array)  
array1, array2 = np.hsplit(array, [2])  
print("Après division")  
print(array1)
```

```
print(array2)
```

```
Avant division
[[ 2.  5.  6.]
 [ 9. 10. 11.]
 [ 5.  6.  9.]]
Après division
[[ 2.  5.]
 [ 9. 10.]
 [ 5.  6.]]
[[ 6.]
 [11.]
 [ 9.]]
```

### 4.3 5.3. Concaténer/combiner des tableaux

#### 4.3.1 5.3.1. La fonction concatenate()

```
[61]: array1 = np.array([1, 2, 3,4,5])
      array2 = np.array([6,7,8,9])
      print(array1)
      print(array2)
      np.concatenate([array1, array2])
```

```
[1 2 3 4 5]
[6 7 8 9]
```

```
[61]: array([1, 2, 3, 4, 5, 6, 7, 8, 9])
```

```
[62]: array1= np.array([[1,2,3],[4,5,6],[7,8,9]], dtype="float")
      array2= np.array([[2,5,6],[9,10,11],[5,6,9]], dtype="float")
      print(array1)
      print(array2)
      np.concatenate([array1, array2])
```

```
[[1.  2.  3.]
 [4.  5.  6.]
 [7.  8.  9.]]
[[ 2.  5.  6.]
 [ 9. 10. 11.]
 [ 5.  6.  9.]]
```

```
[62]: array([[ 1.,  2.,  3.],
             [ 4.,  5.,  6.],
             [ 7.,  8.,  9.],
             [ 2.,  5.,  6.],
             [ 9., 10., 11.],
             [ 5.,  6.,  9.]])
```

```
[63]: array3= np.array([[4,10,13],[7,14,15]], dtype="float")
      np.concatenate([array1, array2,array3])
```

```
[63]: array([[ 1.,  2.,  3.],
            [ 4.,  5.,  6.],
            [ 7.,  8.,  9.],
            [ 2.,  5.,  6.],
            [ 9., 10., 11.],
            [ 5.,  6.,  9.],
            [ 4., 10., 13.],
            [ 7., 14., 15.]])
```

### 4.3.2 5.3.2. La fonction vstack() et hstack()

```
[64]: import numpy as np
      array1 = np.array([15,16,47])
      array2 = np.array([[1,2,3],[4,5,6],[7,8,9]], dtype="float")
      print(array1)
      print("----")
      print(array2)
      print("----")
      print(np.vstack([array1, array2]))
      array1 = np.array([[15],[16],[47]])
      print("----")
      print(array1)
      print("----")
      print(array2)
      print("----")
      print(np.hstack([array1, array2]))
```

```
[15 16 47]
----
[[1. 2. 3.]
 [4. 5. 6.]
 [7. 8. 9.]]
----
[[15. 16. 47.]
 [ 1.  2.  3.]
 [ 4.  5.  6.]
 [ 7.  8.  9.]]
----
[[15]
 [16]
 [47]]
----
[[1. 2. 3.]
 [4. 5. 6.]
```

```
[7. 8. 9.]
---
[[15.  1.  2.  3.]
 [16.  4.  5.  6.]
 [47.  7.  8.  9.]]
```

```
[65]: # multiplication de deux tableaux ndarrays
array2*array2
```

```
[65]: array([[ 1.,  4.,  9.],
            [16., 25., 36.],
            [49., 64., 81.]])
```

```
[66]: # produit matriciel entre les deux matrices array2
np.mat(array2)*np.mat(array2)
```

```
[66]: matrix([[ 30.,  36.,  42.],
             [ 66.,  81.,  96.],
             [102., 126., 150.]])
```

```
[ ]:
```