

Université Djilali Liabes - Sidi Bel Abbès
Faculté des sciences économiques, commerciales et sciences de gestion
1^{ère} année LMD
Python 2
2025-2026
TP N° 1

Ecrire un programme python qui permet :

- 1) de créer les ndarray a1= [8 4 3] et a2= [[4 2] [4 6]] en utilisant la fonction Numpy.array()
 - 2) de créer les ndarray a3= [1. 1. 1.], a4= [[0. 0. 0.] [0. 0. 0.]] en utilisant les fonctions Numpy.zeros() et Numpy.ones ()
 - 3) de créer les ndarray a5= [3 4 5 6 7] et a6= [2 4 6] en utilisant la fonction Numpy.arange()
 - 4) d'afficher tous les ndarray (a1 à a6)
 - 5) d'afficher les attributs (shape et ndim) de a4
 - 6) d'afficher le premier et dernier élément de a5
 - 7) d'afficher les éléments a5[1] à a5[3]
 - 8) d'afficher les éléments supérieurs à 4 de a5
 - 9) d'afficher les éléments a2[0, 0] et a2[1, 1]
-

Université Djilali Liabes - Sidi Bel Abbès
Faculté des sciences économiques, commerciales et sciences de gestion
1^{ère} année LMD
Python 2
2025-2026
TP N° 1

Ecrire un programme python qui permet :

- 1) de créer les ndarray a1= [8 4 3] et a2= [[4 2] [4 6]] en utilisant la fonction Numpy.array()
- 2) de créer les ndarray a3= [1. 1. 1.], a4= [[0. 0. 0.] [0. 0. 0.]] en utilisant les fonctions Numpy.zeros() et Numpy.ones ()
- 3) de créer les ndarray a5= [3 4 5 6 7] et a6= [2 4 6] en utilisant la fonction Numpy.arange()
- 4) d'afficher tous les ndarray (a1 à a6)
- 5) d'afficher les attributs (shape et ndim) de a4
- 6) d'afficher le premier et dernier élément de a5
- 7) d'afficher les éléments a5[1] à a5[3]
- 8) d'afficher les éléments supérieurs à 4 de a5
- 9) d'afficher les éléments a2[0, 0] et a2[1, 1]

Correction : Ecrire un programme python qui permet :

1. de créer les ndarray **a1= [8 4 3]** et **a2= [[4 2] [4 6]]** en utilisant la fonction **Numpy.array()**

Un ndarray (N-dimensional array) Permet de créer un tableau multidimensionnel dans le qu'elle on va stocker et manipuler des données numériques du même type (entiers, réels, etc.),

```
import numpy as np
# Création des ndarray
a1 = np.array([8, 4, 3]) # a1 = [8 4 3]
a2 = np.array([[4, 2],[4, 6]]) # a2 = [[4 2]
                                #      [4 6]]

# Affichage
print("a1 =", a1)
print("a2 =", a2)
```

2. de créer les ndarray **a3= [1. 1. 1.]**, **a4= [[0. 0. 0.] [0. 0. 0.]]** et **a5= [0 0 0 0]** en utilisant les fonctions **Numpy.zeros()** et **Numpy.ones ()**

```
import numpy as np
a3 = np.ones(3) # a3 = [1. 1. 1.]

a4 = np.zeros((2, 3)) # a4 = [[0. 0. 0.]
                            #      [0. 0. 0.]]

a5 = np.zeros(4, dtype=int) # a5 = [0 0 0 0]

print("a3 =", a3)
print("a4 =", a4)
print("a5 =", a5)
```

3. Créer les ndarray a6= [3 4 5 6 7] et a7= [2 4 6] en utilisant la fonction Numpy.arange()

```
import numpy as np
a6 = np.arange(3, 8)    #[3 4 5 6 7]
a7 = np.arange(2, 7, 2) #[2 4 6]
print(a6)
print(a7)
```

4. Afficher tous les ndarray (a1 à a7)

```
import numpy as np
a1 = np.array([8, 4, 3])           # a1 = [8 4 3]
a2 = np.array([[4, 2],[4, 6]])    # a2 = [[4 2]
                                   #      [4 6]]

a3 = np.ones(3)                   # a3 = [1. 1. 1.]

a4 = np.zeros((2, 3))             # a4 = [[0. 0. 0.]
                                   #      [0. 0. 0.]]

a5 = np.zeros(4, dtype=int)       # a5 = [0 0 0 0]
a6 = np.arange(3, 8)              #[3 4 5 6 7]

a7 = np.arange(2, 7, 2)           #[2 4 6]

print("a1 =", a1)                 # a1 = [8 4 3]
print("a2 =", a2)                 # a2 = [[4 2]
                                   #      [4 6]]

print("a3 =", a3)                 # a3 = [1. 1. 1.]
print("a4 =", a4)                 # a4 = [[0. 0. 0.]
                                   #      [0. 0. 0.]]

print("a5 =", a5)                 # a5 = [0 0 0 0]
print("a6 =", a6)                 #[3 4 5 6 7]

print("a7 =", a7)                 #[2 4 6]
```

5 Afficher les attributs (shape et ndim) de a4

```
import numpy as np
a4 = np.zeros((2, 3)) # a4 = [[0. 0. 0.]
                        #      [0. 0. 0.]]
print("Shape :", a4.shape) # Shape : (2, 3)
print("Nombre de dimensions :", a4.ndim) # Nombre de
dimensions : 2
```

6 Afficher le premier et dernier élément de a5

```
import numpy as np
a5 = np.arange(3,8) # a5 = [3,4,5,6,7]
print("Premier élément de a5:", a5[0]) #Premier élément : 3
print("Dernier élément de a5:", a5[-1]) #Dernier élément : 7
```

7 Afficher les éléments a5[1] à a5[3]

```
print(a5[1:4]) # [4 5 6]
```

8 Affichage des éléments supérieurs à 4 de a5

```
print(a5[a5 > 4]) # [5 6 7 ]
```

9 Afficher les éléments a2[0, 0] et a2[1, 1]

```
import numpy as np
a2 = np.array([[4, 2],[4, 6]])
print("a2[0, 0]=" , a2[0, 0]) # 4
print("a2[1, 1]=" , a2[1, 1]) # 6
```