

\* library \*

\* Numpy \*

\* ~~import~~ import numpy as np

1) np.array ( )

print → list = [1, 2, 3, 4] → [1, 2, 3, 4]  
 → np.array(list) → [1 2 3 4]  
 ↓  
 ndarray  
 ↖  
 number of dimension

\* a = np.array(10)

a.ndim → dimension of array.

\* الوصول إلى index في array

arr[1][1][1]

or ↓

arr[1, 1, 1]



## \* custom Dimensions \*

```
AZmeD = np.array([1,2,3], ndmin=3)
```

3 dimension ← array ال ←

```
print(AZmeD) → [[[ 1 2 3 ]]]
```

```
print(AZmeD[0]) → [[ 1 2 3 ]]
```

\* list → Data كذا نوعه

array لود فليكي كذا نوعه  
string .

\* np.arange(100)

arr فيزي اقامه

0 → 99

\* X = np.arange(50)

\* item size → حجم العنصر الواحد

\* size → حجم ال array كذا  
بتي فيزي كام عنبر



## Final Notes

\*  $x = \text{np.array}([1, 2, 3, 4])$   
 $\text{print}(x.dtype) \rightarrow \text{int64}$   
 نوع array (int64)

\*  $x = \text{np.array}([ [1, 2, 3], [2, 3, 4] ])$

$\text{print}(x.shape) \rightarrow (2, 3)$

← size array 2، 3 عناصر

$\text{print}(x.size) \rightarrow 6 \rightarrow$  عدد عناصر

\*  $x = \text{np.array}([1, 5, 2, 5, 3, 5], dtype = \text{np.int4})$

$\text{print}(x) \rightarrow [1 \ 2 \ 3]$

← int 4

\* how to create array

$dtype = \text{int}$

→  $x = \text{np.zeros}((3, 4))$  ← how to create array  
 \* 3 array of size 4

→  $\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$



\*  $X = \text{np.ones}(4, 5)$  ✓

~~\*  $X = \text{np.ones}$~~

\*  $X = \text{np.full}((4, 3), 5)$

← shape      الرقم الذي يملأ به

\*  $X = \text{np.eye}(5)$  →  $(5 \times 5)$  array  
والـ diag يملأه بـ 1

\*  $X = \text{np.diag}([10, 20, 30, 40])$

→  $(4 \times 4)$  array والـ diag يملأه بـ 1 (مفرد)

\*  $X = \text{np.arange}(10)$

$\text{print}(X)$  →  $[1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9]$

\*  $\text{np.arange}(4, 10)$

→ 4 → 9

✓



\*  $\text{np.arange}(1, 14, 3)$   $\xrightarrow{\text{step}}$

خطوة 3 يعني  $\rightarrow [1, 4, 7, 10, 13]$

\*  $y = [1, 2, 3, 4, 5, 6]$

$x = \text{np.reshape}(y, (3, 2))$

$\rightarrow \begin{bmatrix} [1, 2] \\ [3, 4] \\ [5, 6] \end{bmatrix}$

لعدد الأرقام في كل صف  
مع الخطأ يعني error

\*  $x = \text{np.linspace}(0, 30, 10)$

← أبعاد array تكون من 0 إلى 30  
والتي هي 10 عناصر  
enclused يعني

\*  $x = \text{np.random.randint}(4, 15, (3, 2))$   $\xrightarrow{\text{shape}}$

بأرقام عشوائية بين 4 و 15



\* `x = np.random.normal(0, 0, size=(5, 5))`

$\swarrow$  mean       $\swarrow$  std       $\swarrow$  حجم

\* `x = [ ]`

`x.mean()`  $\rightarrow$  ✓

`x.std()`  $\rightarrow$  ✓

`x.max()`  $\rightarrow$  ✓

`x.min()`  $\rightarrow$  ✓

`print((x > 0).sum())`  $\rightarrow$  عدد الأعداد الموجبة

وزن على التوالي

`x.median()`  $\rightarrow$  ✓

\* `np.delete(y, [0, 2], axis=0)`

$\swarrow$  array       $\swarrow$  حذف       $\swarrow$  = 0  $\rightarrow$  صفوف  
 $\swarrow$  index       $\swarrow$  = 1  $\rightarrow$  أعمدة

\*

`x = [1 2 3]`

`x = np.append(x, 6)`

`print(x)`  $\rightarrow$  `[1 2 3 6]`



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

```
w = np.append(y, [[10], [11], [12]], axis=1)
```

```
↳ [[1 2 3 10]
    [4 5 6 11]
    [7 8 9 12]]
```

منقط  
في آخر عمود

\* axis=1 في سطر في آخر صف

```
✓ [7 8 9 12]]
```

```
* x = np.array([[1, 2, 3], [5, 2, 8], [1, 2, 3]])
```

```
print(*np.unique(x)) → [1 2 3 5 8]
```

بتعديل المتكرر

\*

```
x = [(x > 10) & (x < 17)] = -1
```

```
print(x)
```

صحيح للقيم التي بين 10 و 17

بقية -1 و يطبع الباقي صاف

```
* print(x[x > 10])
```

طبع الـ 10 فما فوق