

## مرحباً طلابي:

أردت توضيح بعض النقاط في الدرس السابق وهي:

- طريقة عمل الدالة linspace:  
درسنا في المرحلة الثانوية المتتابة الحسابية وعلما كيفية التمييز بينها وبين المتتابة الهندسية إلخ...  
وعلمنا أن الحد العام للمتتابة الحسابية هو  $a + (n - 1) \times d$  حيث أن:
  - $a$  تمثل الحد الأول من المتتابة
  - $n$  تمثل عدد حدود المتتابة
  - $d$  تمثل الفرق بين حد من حدود المتتابة والحد الذي يسبقه

وعلمنا أن  $L$  تمثل الحد الأخير من المتتابة وهي تساوي الحد العام للمتتابة أي  $L = a + (n - 1) \times d$

ففي الدالة linspace:

- المدخل الأول هو عبارة عن الحد الأول من المتتابة وبنرمز له بالحرف (  $a$  )
- المدخل الثاني هو عبارة عن الحد الأخير من المتتابة وبنرمز له بالحرف (  $L$  )
- المدخل الثالث عبارة عن عدد حدود المتتابة الحسابية وبنرمز لها بالحرف (  $n$  )

فسترجع لنا الدالة مصفوفة أو ممكن نقول هنا متتابة حدها الأول المدخل الأول وحدها الأخير المدخل الثاني وعدد حدودها المدخل الثالث ويتم حساب قيمة  $d$  التي تمثل الفرق بين حدين متتاليين عن طريق العلاقة التالية:

$$d = L - a \text{ مقسومة على } n - 1$$

سأعطيك بعض الأمثلة السهلة البسيطة لتبين لك طريقة عملها

```
In [1]: import numpy as np  
np.linspace(2,3,5)
```

```
Out[1]: array([2. , 2.25, 2.5 , 2.75, 3.  ])
```

كما تلاحظ في المثال السابق الحد الأول هو المدخل الأول والحد الأخير هو المدخل الثاني وتم حساب قيمة  $d$  من العلاقة  $d = L - a$  مقسومة على  $n - 1$  وهنا تساوي ربع فتلاحظ الزيادة بمقدار ربع عن الحد الذي يسبقه

من الممكن تحديد عدد الصفوف والأعمدة أي شكل المصفوفة وتحديد موضع ظهور الواحد أنظر الأمثلة التالية:

```
In [2]: np.eye(2,3)
```

```
Out[2]: array([[1., 0., 0.],
               [0., 1., 0.]])
```

```
In [3]: np.eye(4,6)
```

```
Out[3]: array([[1., 0., 0., 0., 0., 0.],
               [0., 1., 0., 0., 0., 0.],
               [0., 0., 1., 0., 0., 0.],
               [0., 0., 0., 1., 0., 0.]])
```

المدخل الأول هو عدد الصفوف والمدخل الثاني عدد الأعمدة

وتأخذ أيضاً مدخل ثالث يحدد موضع الواحد

```
In [4]: np.eye(4,4,0)
```

```
Out[4]: array([[1., 0., 0., 0.],
               [0., 1., 0., 0.],
               [0., 0., 1., 0.],
               [0., 0., 0., 1.]])
```

فمثلاً

```
In [5]: np.eye(4,4,1)
```

```
Out[5]: array([[0., 1., 0., 0.],
               [0., 0., 1., 0.],
               [0., 0., 0., 1.],
               [0., 0., 0., 0.]])
```

```
In [6]: np.eye(4,4,3)
```

```
Out[6]: array([[0., 0., 0., 1.],
               [0., 0., 0., 0.],
               [0., 0., 0., 0.],
               [0., 0., 0., 0.]])
```

```
In [7]: np.eye(4,4,-2)
```

```
Out[7]: array([[0., 0., 0., 0.],
               [0., 0., 0., 0.],
               [1., 0., 0., 0.],
               [0., 1., 0., 0.]])
```

الصورة التالية توضح ما يمثلها المدخل الثالث

In [22]: `np.eye(4,4,0)`

Out[22]: `array([[ 1., 0., 0., 0.],  
[ 0., 1., 0., 0.],  
[ 0., 0., 1., 0.],  
[ 0., 0., 0., 1.]])`

مرحباً طلابي:

أردت توضيح بعض النقاط في هذا الدرس:

طريقة الحصول على قطعة من المصفوفة علمنا أنها بهذه الطريقة

```
In [1]: import numpy as np
arr = np.arange(50).reshape(5,10)
arr
```

```
Out[1]: array([[ 0,  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, 26, 27, 28, 29],
               [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
               [40, 41, 42, 43, 44, 45, 46, 47, 48, 49]])
```

```
In [2]: arr[1:3,3:5]
```

```
Out[2]: array([[13, 14],
               [23, 24]])
```

هذه الصورة توضح فكرة أخذ قطعة من المصفوفة

```
In [31]: arr = np.arange(50).reshape(5,10)
arr
```

```
Out[31]: array([[ 0,  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, 26, 27, 28, 29],
                [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
                [40, 41, 42, 43, 44, 45, 46, 47, 48, 49]])
```

مرحباً طلابي:

أردت توضيح بعض النقاط في هذا الدرس:

- طريقة عمل الدالة `np.exp()`

وظيفتها أنها تقوم بأخذ كل عنصر أو حد من حدود المصفوفة وترفعه كأس لعدد أويلر 2.718281828 وهو ثابت رياضي يرمز له بالحرف العربي هـ أو e بالإنجليزية للمزيد عن عدد أويلر

(([https://en.wikipedia.org/wiki/E\\_\(mathematical\\_constant\)](https://en.wikipedia.org/wiki/E_(mathematical_constant)) ([https://en.wikipedia.org/wiki/E\\_\(mathematical\\_constant\)](https://en.wikipedia.org/wiki/E_(mathematical_constant))

وبالتالي ترجع لنا النمو الأسّي

[نمو\\_أسّي/https://ar.wikipedia.org/wiki](https://ar.wikipedia.org/wiki/نمو_أسّي) ([نمو\\_أسّي/https://ar.wikipedia.org/wiki](https://ar.wikipedia.org/wiki/نمو_أسّي))

```
In [1]: import numpy as np
arr = np.arange(10)
np.exp(arr)
```

```
Out[1]: array([1.00000000e+00, 2.71828183e+00, 7.38905610e+00, 2.00855369e+01,
5.45981500e+01, 1.48413159e+02, 4.03428793e+02, 1.09663316e+03,
2.98095799e+03, 8.10308393e+03])
```

```
In [2]: 2.718281828 ** 2
```

```
Out[2]: 7.389056096435021
```

```
In [3]: 2.718281828 ** 3
```

```
Out[3]: 20.085536913011932
```

وال e+01 معناها أنه قسم الناتج على 10

وال e+02 معناها قسم الناتج على 100

وال e+03 معناها قسم الناتج على 1000

وهكذا...

## مرحباً طلابي

أردت توضيح بعض النقاط قبل البدء في حل الواجب

أولاً : الميثود sum

```
In [1]: import numpy as np
arr = np.arange(25).reshape(5,5)
arr
```

```
Out[1]: array([[ 0,  1,  2,  3,  4],
               [ 5,  6,  7,  8,  9],
               [10, 11, 12, 13, 14],
               [15, 16, 17, 18, 19],
               [20, 21, 22, 23, 24]])
```

```
In [2]: arr.sum()
```

```
Out[2]: 300
```

وظيفة هذه الميثود إرجاع مجموع المصفوفة

هناك مدخل يسمى axis قيمته الافتراضية None وفي حالة إذا كان هذا المدخل None يتم حساب المجموع الكلي وإذا كان 0 يرجع لنا مصفوفة تحتوي على مجموع كل عمود هكذا

```
In [3]: arr.sum(axis=0)
```

```
Out[3]: array([50, 55, 60, 65, 70])
```

حتي تتأكد

```
In [4]: print(arr)
print("-"*30)
print(arr.sum(axis=0))
```

```
[[ 0  1  2  3  4]
 [ 5  6  7  8  9]
 [10 11 12 13 14]
 [15 16 17 18 19]
 [20 21 22 23 24]]
-----
[50 55 60 65 70]
```

وإذا كانت axis تساوي 1 يتم جمع كل صف هكذا

```
In [5]: arr.sum(axis=1)
```

```
Out[5]: array([ 10,  35,  60,  85, 110])
```

---

ثانياً: الميثود std

وظيفتها أرجاع الانحراف المعياري للمصفوفة

للمزيد عن الانحراف المعياري

[انحراف\\_معيارى/https://ar.wikipedia.org/wiki](https://ar.wikipedia.org/wiki/انحراف_معيارى) [انحراف\\_معيارى/https://ar.wikipedia.org/wiki](https://ar.wikipedia.org/wiki/انحراف_معيارى)

مثال:

In [6]: arr.std()

Out[6]: 7.211102550927978