



للمبتدئين مكتبة نخباي - بالعربي



Qusay AL-Btoush

github.com/qusaybtoush

<https://www.linkedin.com/in/qusayal-btoush>

<https://www.kaggle.com/qusaybtoush1990>

NumPy

للمبتدئين

هي مكتبة مفيدة للعمل مع البيانات الرقمية والعمليات الرياضية والعلمية. من إجراء العمليات الرياضية على مصفوفات ومصفوفات متعددة الأبعاد بكفاءة عالية. تساعد في تسهيل الحسابات الرياضية المعقدة وتوفير أدوات قوية للتعامل مع البيانات العددية.

تُستخدم على نطاق واسع في مجالات مثل علم البيانات والتعلم الآلي والحساب العلمي، وهي أساسية في برمجة عند التعامل مع البيانات العددية.

- Qusay AL-Btoush : إذا كان لدي أي سؤال تواصل معنا

- <https://github.com/qusaybtoush>
- <https://www.linkedin.com/in/qusayal-btoush/>
- <https://www.kaggle.com/qusaybtoush1990>

```
In [1]: # تحميل المكتبة
import numpy as np
```

```
In [61]: # انشاء مصفوفة
x= np.array([1,2,3,4,5])
y= np.array([4,5,8,9,10])
print (x)
print (y)
```

```
[1 2 3 4 5]
[ 4  5  8  9 10]
```

```
In [7]: # انشاء مصفوفة تحتوي فقط على رقم واحد فقط اختر عدد الصفوف و عدد الاعمده
one = np.ones((3,4)) # صفوف 3 , 4 اعمده
print(one)
```

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

```
In [9]: # انشاء مصفوفة تحتوي فقط على رقم صفر فقط اختر عدد الصفوف و عدد الاعمده
ze = np.zeros((5,3)) # 5 rows, 3 column/ 5 صفوف
print(ze)
```

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

```
In [11]: # انشاء مصفوفة تحتوي فقط على رقم الذي تريده فقط اختر عدد الصفوف و عدد الاعمده
fu = np.full((3,6),8) # 3 صفوف و 6 اعمده
print(fu)
```

```
[[8 8 8 8 8 8]
 [8 8 8 8 8 8]
 [8 8 8 8 8 8]]
```

In [12]: *انشاء مصفوفه من رقم معين الى رقم معين بخطوات معينه ايضا*
`st = np.arange(5,31,5) # 5 الى 31 و الخطوه 5`
`print(st)`

```
[ 5 10 15 20 25 30]
```

In [21]: *انشاء مصفوفه من الى رقم معي و عدد عناصر معين*
`li = np.linspace(1,5,6) # 6 وعدد العناصر 6`
`print(li)`

```
[1.  1.8 2.6 3.4 4.2 5. ]
```

In [24]: *انشاء مصفوفه رقم الابعاد الذي تريده و رقم واحد بشكل قطر و الباقي اصفار*
`np.eye(4)`

Out[24]: `array([[1., 0., 0., 0.],
 [0., 1., 0., 0.],
 [0., 0., 1., 0.],
 [0., 0., 0., 1.]])`

In [31]: *انشاء مصفوفه ب ارقام عشويه من صفر الى واحد و انت تحدد الابعاد المصفوفه*
`np.random.rand(2,2) #array 2 rows, 2 column and the element from 0 to 1`

Out[31]: `array([[0.07635816, 0.99177734],
 [0.23383263, 0.93246568]])`

In [37]: *انشاء مصفوفه برقم صحيح وانت تحدد اعلى رقم تريده*
`np.random.randint(5,size=(4,4)) # 5 الى صفر ارقام عشوائيه من صفر الى 5`
و 4 اعمده وايضا اعلى قيمه 5 اذا ارقام عشوائيه من صفر الى 5

Out[37]: `array([[4, 0, 1, 0],
 [2, 4, 0, 2],
 [1, 1, 2, 4],
 [2, 0, 3, 1]])`

In [39]: *انشاء مصفوفه ذات بعدين*
`arr= np.array([[1,3,5,9],[10,15,6,8]])`
`print (arr)`

```
[[ 1  3  5  9]
 [10 15  6  8]]
```

In [45]: *ابعاد المصفوفه معرفه /عرض*
`np.ndim(arr) # استخدم فنكشن ndim`

Out[45]: 2

In [72]: *عرض العناصر الغير مكرره يونيك*
`np.unique(arr) # استخدم فنكشن unique`

Out[72]: array([1, 3, 5, 6, 8, 9, 10, 15])

In [46]: *عرض شكل المصفوفة و الابعاد*
`np.shape(arr)` *لدينا 2 صف و 4 اعمده , shape استخدم فنكشن*

Out[46]: (2, 4)

In [48]: *عرض كم عدد العناصر في المصفوفة*
`np.size(arr)` *لدينا 8 عناصر , size استخدم فنكشن*

Out[48]: 8

In [52]: *عرض نوع الداتا*
`arr.dtype` *استخدم فنكشن داتا تايب*

Out[52]: dtype('int32')

In [58]: *يمكن انشاء مصفوفات تحتوي على لوجيك صح و خطأ*
`bol = np.array([True, False, False, False, True])`
`bol.dtype` *نوع الداتا تايب بولين*

Out[58]: dtype('bool')

In [54]: *يمكن قلب المصفوفة*
`np.flip(arr)`

Out[54]: array([[8, 6, 15, 10],
[9, 5, 3, 1]])

In [87]: *عرض المصفوفة مره اخرى*
`arr`

Out[87]: array([[1, 3, 5, 9],
[10, 15, 6, 8]])

In [88]: *عرض العناصر حسب تحديد الموقع صفر اول صف و 1 ثاني صف*
`arr[1]` *ثاني صف*

Out[88]: array([10, 15, 6, 8])

In [106...]: *[عامود, صف]*
عرض كل الصفوف : , و اول عامود
`arr[:,1]`

Out[106...]: array([3, 15])

In [95]: *عرض العناصر حسب الموقع المحدد صف اول و العامود الثاني*
`arr[1,2]`

Out[95]: 6

In [96]: *يمكن قلب المصفوفة*
`np.transpose(arr)`

Out[96]: array([[1, 10],
 [3, 15],
 [5, 6],
 [9, 8]])

In [101... *يمكن تغيير حجم المصفوفة وتكرار العناصر مره اخرى*
`np.resize(arr,(5,5))`

Out[101... array([[1, 3, 5, 9, 10],
 [15, 6, 8, 1, 3],
 [5, 9, 10, 15, 6],
 [8, 1, 3, 5, 9],
 [10, 15, 6, 8, 1]])

In [102... *اضافه عناصر الى المصفوفه من خلال فنكشن ابند*
`np.append(arr,(1,5,6))`

Out[102... array([1, 3, 5, 9, 10, 15, 6, 8, 1, 5, 6])

In [105... *يمكن حذف عناصر من المصفوفه*
`np.delete(arr,[1,1])`

Out[105... array([1, 5, 9, 10, 15, 6, 8])

In [62]: *الرجوع الى عناصر المصفوفه اكس و واي الذي تم تعريفهم في البدايه*
`print (x)`
`print (y)`

```
[1 2 3 4 5]
[ 4 5 8 9 10]
```

In [65]: *يمكن اجراء اي عمليات حسابيه على المصفوفات*
طرح, قسمه, ضرب, جمع

```
print (x + y ) #جمع
print ( x - y ) #طرح
print (x / y ) #قسمه
print (x * y ) #ضرب
```

```
[ 5  7 11 13 15]
[-3 -3 -5 -5 -5]
[0.25    0.4    0.375    0.44444444 0.5    ]
[ 4 10 24 36 50]
```

In [67]: *# يمكن اجراء العديد من العمليات الرياضيه , Like sin, sqrt, cos, log ...*
`print (np.sin(x))`
`print (np.sqrt(x))`
`print (np.cos(y))`
`print (np.log(y))`

```
[ 0.84147098  0.90929743  0.14112001 -0.7568025  -0.95892427]
[1.          1.41421356  1.73205081  2.          2.23606798]
[-0.65364362  0.28366219 -0.14550003 -0.91113026 -0.83907153]
[1.38629436  1.60943791  2.07944154  2.19722458  2.30258509]
```

In [70]:

```
# يمكن استخدام فيكشن ل الاحصاء جمع اقل و اكبر قيمه متوسط وترتيب القيم ايضا
print (x)

print ("The Sum :",np.sum(x) )
print ("The Minimum : ",np.min(x) )
print ("The Maximum ",np.max(x) )

print ("The Average :",np.mean(x) )
print ("The Median : " , np.median(x) )
print ("The Sort the array : " , np.sort(x) )
```

```
[1 2 3 4 5]
The Sum : 15
The Minimum : 1
The Maximum 5
The Average : 3.0
The Median : 3.0
The Sort the array : [1 2 3 4 5]
```

- Qusay AL-Btoush : اذا كان لدي اي سؤال تواصل معنا
- <https://github.com/qusaybtoush>
- <https://www.linkedin.com/in/qusayal-btoush/>
- <https://www.kaggle.com/qusaybtoush1990>