Contents

[1. Tạo mảng n chiều với các phần tử bằng 1](#_heading=h.gjdgxs) 1

[2. Tạo mảng n chiều với các phần tử bằng 0](#_heading=h.30j0zll) 2

[3. Mảng 0 được đóng khung bởi các phần tử 1](#_heading=h.1fob9te) 3

[4. Đổi các giá trị 1 thành 0 và 0 thành 1 trong mảng](#_heading=h.3znysh7) 4

[5. Đảo mảng nhị phân n chiều](#_heading=h.i2fgtew1rxej) 6

[6. Mặt nạ nhị phân](#_heading=h.u9nwtrmn6dm9) 6

[7. Số nguyên từ 0-100](#_heading=h.5s14c4u3onzh) 9

[8. Số chẵn dương từ 2-1000](#_heading=h.z821jaou1064) 10

[9. Nhập 1 số để tạo mảng 2 chiều](#_heading=h.ar7rv6wwwp7) 10

[10. Chuyển mảng 1 chiều thành hàng trong mảng 2 chiều](#_heading=h.fecm8x3f2qnb) 11

[11. Đổi từ mảng 1 chiều sang cột trong mảng 2 chiều](#_heading=h.q21ouo95ecwd) 12

[12. Tạo mảng hằng n chiều](#_heading=h.go9k3dhufza1) **Error! Bookmark not defined.**

[13. Tạo mảng 2 chiều](#_heading=h.57txjcztsxm) 15

[14. Lợp mảng tạo thành mảng mới](#_heading=h.sywv2pblm2xn) 17

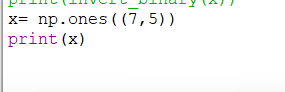
|  |  |
| --- | --- |
| Tên | MSSV |
| Trần Nguyễn Quang Lâm | 20139040 |
| Tô Gia Huy | 20139003 |

# 1. **Tạo mảng n chiều với các phần tử bằng 1**

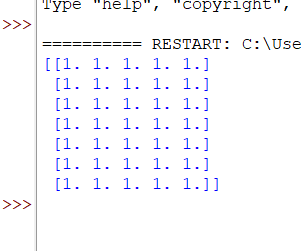
Tạo mảng 2 chiều với mỗi phần tử bằng 1

Thư viện NumPy có hàm *np.ones()* để dùng cho việc tạo mảng này, dữ liệu đầu vào là định dạng mảng và số phần tử tương ứng, tạo ra mảng theo dữ liệu đã cấp với mỗi phần tử bằng 1

Tạo một mảng với 7 hàng và 5 cột, các phần tử đều bằng 1



Kết quả



Code tạo ra một mảng 2 chiều với số cột là 7, số hàng là 5 và các phần tử bằng 1

Giải thích : import numpy as np: gọi đến thư việc numpy và lưu nó dưới tên np

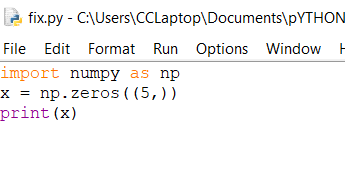
gọi hàm ones với cú pháp np.ones() và truyền vào đối số 7,5 , hàm sẽ hiểu là 7 cột và 5 hàng nên tạo ra mảng có giá trị như hình ở trên

# 2. Tạo mảng n chiều với các phần tử bằng 0

Tạo mảng 2 chiều với tất cả phần tử bằng 0

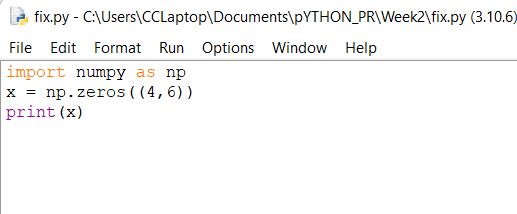
Thư viện NumPy có hàm *np.zeros()* để dùng cho việc tạo mảng này, dữ liệu đầu vào là định dạng mảng và số phần tử tương ứng, tạo ra mảng theo dữ liệu đã cấp với mỗi phần tử bằng 0

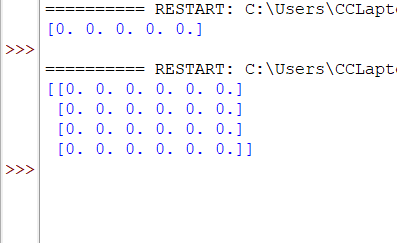
Ví dụ:





Tạo ra mảng 2 chiều với số cột là 4 và số hàng là 6, tất cả phần tử bằng 0



Kết quả  


Giải thích : import numpy as np: gọi đến thư việc numpy và lưu nó dưới tên np

Gọi hàm ones với cú pháp np.zeros() và truyền vào đối số 4, 6, hàm sẽ hiểu là 6 cột và 4 hàng nên tạo ra mảng có giá trị như hình ở trên

# 3. Mảng 0 được đóng khung bởi các phần tử 1

Triển khai hàm framed\_zeros(). Với dữ liệu đầu vào là số cột và hàng, hàm này tạo ra mảng 2 chiều tương ứng với số cột hàng đã cấp, trong đó các phần tử bằng 0, ngoại trừ các phần tử ở hàng và cột ngoài cùng bằng 1

Đầu tiên khai báo hàm framed\_zeros() với 2 đối số:

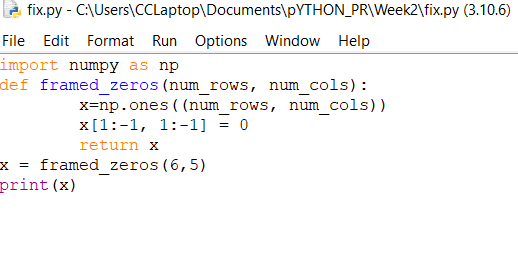
+num\_rows: số hàng của mảng

+num\_cols: số cột của mảng

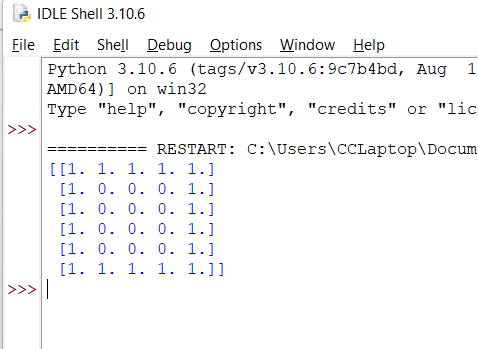
Triển khai hàm để nó trả về mảng 2 chiều với kích thước đã cho, các phần tử bằng 0 ngoại trừ hàng và cột ngoài cùng:

+Dùng hàm np.ones() để khởi tạo tất cả các giá trị bằng 1

+Đổi tất cả giá trị ở trong mảng về 0 : hàng và cột kế ngoài cùng đều có vị trí từ 1:-1



Kết quả:



Giải thích : import numpy as np: gọi đến thư việc numpy và lưu nó dưới tên np

Khai báo hàm framed\_zeros() với 2 đối số là cột và hàng, khởi tạo mảng với kích thước đã cho, tất cả phần tử bằng 1. x= np.ones(6,5)

-Đổi tất cả các phần tử từ hàng 2 cột 2 đến hàng và cột kế cuối (vị trí là [1:-1,1:-1] thành 0 : x[1:-1,1:-1]=0, sau đó trả về x

# 4. Đổi các giá trị 1 thành 0 và 0 thành 1 trong mảng

Tập triển khai hàm alternate\_ones\_zeros(), dữ liệu đầu vào là số hàng và cột, tạo ra mảng 2 chiều với kích cỡ đã cho, giá trị các phần tử lần lượt là 1, 0

Đầu tiên triển khai hàm alternate\_ones\_zeros() với 2 đối số truyền vào

-num\_rows: số hàng

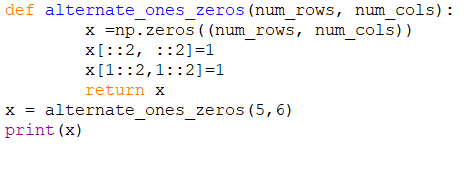
-num\_cols: số cột

Triển khai hàm để nó trả về mảng 2 chiều với kích thước đã cho, các phần từ lần lượt bằng 0 và 1:

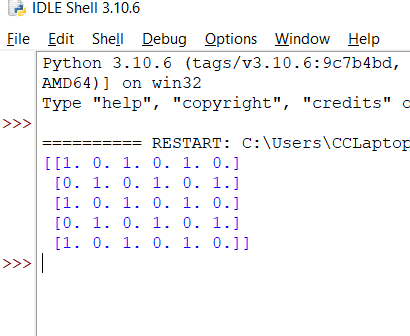
-Khỏi tạo mảng với tất cả các giá trị bằng 0 hoặc bằng 1

-Cho các giá trị xen kẽ nhau với khoảng cách là 2 (::2) =1 hoặc bằng 0 nếu khởi tạo mảng với giá trị 1

Code:



Kết quả



Giải thích: Đề yêu cầu tạo ra một mảng với các giá trị 1, 0 xen kẽ nên ta chỉ cần

-Tạo mảng toàn bộ giá trị = 0: x=np.zeros(5,6)

-Bắt đầu chạy từ hàng 0, cột 0 và cách mỗi hàng mỗi cột sẽ thay đổi giá trị thành 1 qua lệnh x[::2,::2]=1 VD: x[0,0]=1 , x[0,2]=1, x[0,4]=1, ….x[2,0], x[2,2]=1 và tương tự. Xử lý hết các hàng và cột chẵn

-Đổi tiếp : x[1::2,1::2]=1. Hàng bắt đầu từ 1 và Cột bắt đầu từ 1. Ta xử lý hết các hàng và cột lẻ

# 5. Đảo mảng nhị phân n chiều

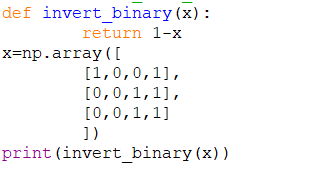
Triển khai hàm invert\_binary(), dữ liệu đầu vào một mảng, trả về mảng có phần tử = 0 tại vị trí phần tử= 1 và trả về phần tử = 0 tại vị trí phần tử bằng 1. Hay nói đơn giản là thay 0 thành 1, thay 1 thành 0.

Các bước triển khai hàm invert\_binary():

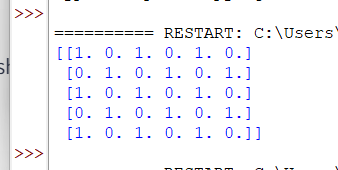
-Khai báo hàm với một đối số

+x mảng chứa các phần tử 1 và 0

-Triển khai hàm để trả về kết quả 1 tại vị trí phần tử 0 và trả về 0 tại vị trí phần tử 1: lấy phần bù x[i,j] = 1 -x



Kết quả:



Giải thích thuật toán: khởi tạo hàm invert\_binary để đổi giá trị các phần tử trong mảng, lấy phần bù để từ 1 ra 0 và từ 0 ra 1(return 1-x)

Khởi tạo một mảng x với các phần từ 0 và 1. Gọi hàm invert\_binary đã tạo và truyền vào đối số chính là mảng x, kết quả trả về là mảng x với các phần tử đã được biến đổi lấy phần bù.

# 6. Mặt nạ nhị phân

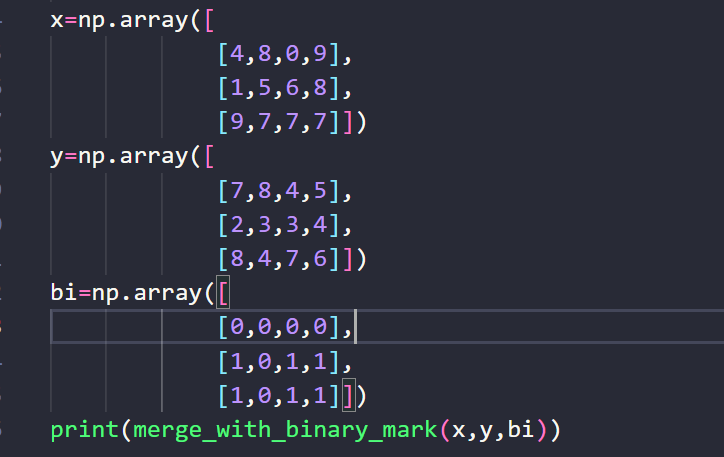
Sử dụng một mặt nạ với các phần tử nhị phân sẽ cho thấy những gì ta muốn cho thấy và che đi những phần tử ta không muốn xuất ra kết quả

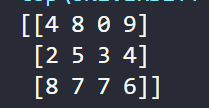
Triển khai hàm merge\_with\_binary\_mask() để kết hợp 2 mảng với cùng kích cỡ, sử dụng binary mask

Calendar

Description automatically generated

Kết quả mảng trả về sẽ là mảng có cùng kích thước với x và y, có phần từ của x tại các vị trí mà binary mask = 0 và có các phần tử của y tại các vị trí mà binary mask = 1





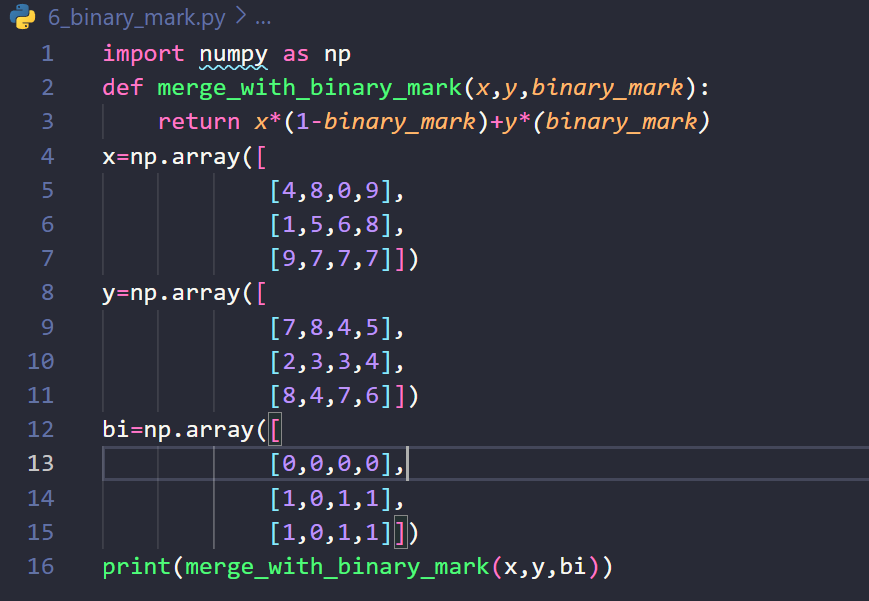
Các bước triển khai hàm merge\_with\_binary\_mask() với 3 đối số

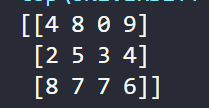
-x: mảng x

-y: mảng y

-binary\_mask mảng nhị phân với kích thước giống như x

Triển khai merge\_with\_binary\_mask() để nó trả về phần tử của x tại các vị trí mà phần tử của mảng binary mask này = 0 và phần tử của y tại vị trí các phần tử mảng mask này = 1





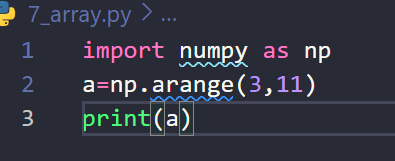
# 7. Số nguyên từ 0-100

Tạo mảng 1 chiều chứa số từ 0 - 100

You might want to look at the [numpy.arange() function](https://numpy.org/doc/stable/reference/generated/numpy.arange.html?highlight=arange#numpy.arange). This function can be used to create a ndarray with all numbers between two given values.

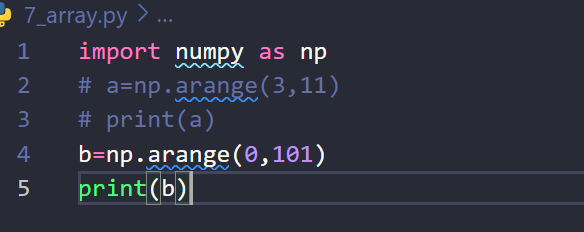
Sử dụng numpy.arrange() function. Function này dùng để tạo mảng với tất cả các số ở giữa 2 số mình nhập.

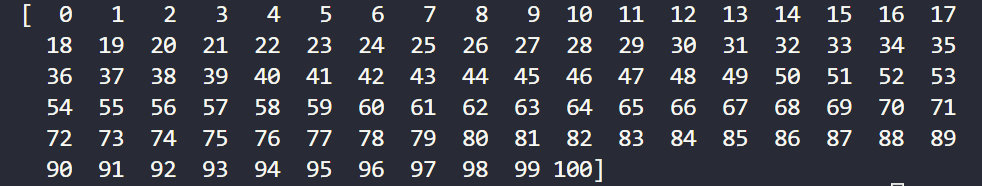
VD tạo mảng chứa các giá trị từ 3- 10





Tạo mảng 1 chiều từ 0 -100:





# 

# 8. Số chẵn dương từ 2-1000

Tạo mảng 1 chiều với số chẵn từ 2 đến 1000.

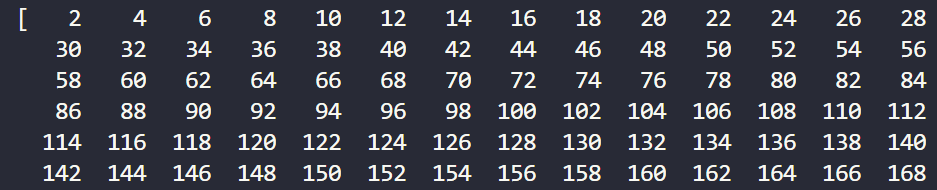
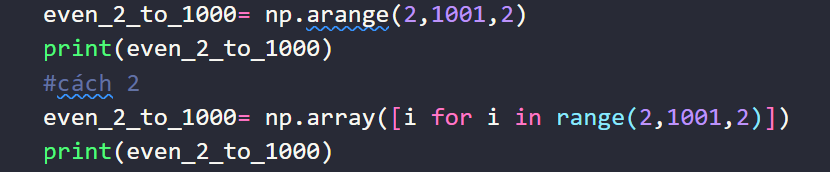
Sử dụng [numpy.arange() function](https://numpy.org/doc/stable/reference/generated/numpy.arange.html?highlight=arange#numpy.arange). Chúng ta sẽ thêm bước nhảy cho hàm để tạo mảng theo ý của mình.

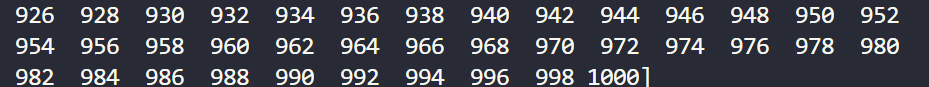
VD: tạo mảng số lẻ (1,11):





Tạo mảng 1 chiều chứa các số chẵn từ 2 đến 1000

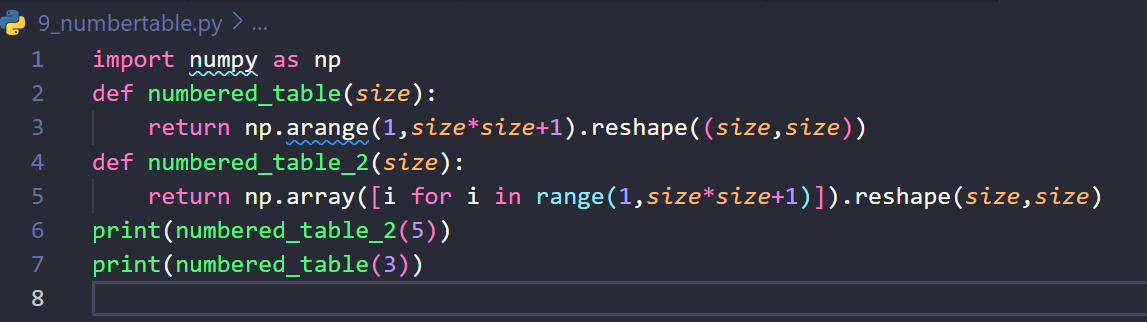
.......

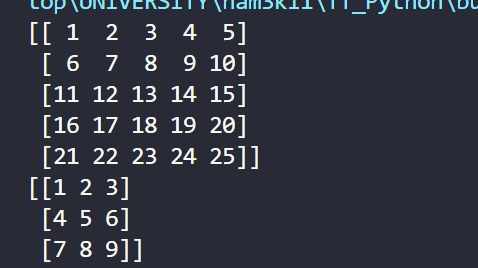


# 9. Nhập 1 số để tạo mảng 2 chiều

Nhập vào 1 số và tạo mảng 2 chiều có số cột và số hàng của số đó

1. định nghĩa một hàm numbered\_table() với 1 đối số: size(là số hàng và cột của mảng 2 chiều ta tạo)
2. Thực hiện hàm đó và trả về mảng (size,size) chứa từ 1 đến size mũ 2

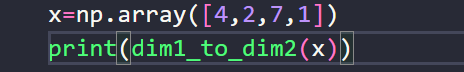




Giải thích: thư viện numpy hỗ trợ hàm reshape, biến mảng 1 chiều thành mảng n chiều mà ta muốn

# 10. Chuyển mảng 1 chiều thành hàng trong mảng 2 chiều

Chạy hàm dim1\_to\_dim2() làm mảng 1 chiều thành mảng 2 chiều với 1 hàng chứa các giá trị cũ.



Dấu ngoặc kép trong output cho ta biết đó là 1 mảng 2 chiều với 1 hàng duy nhất

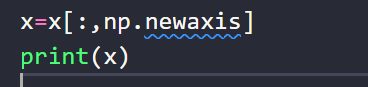
Cách 1 để làm là sử dụng hàm  ndarray.reshape().

Cách 2 là chuyển mảng 1 chiều vào mảng 2 chiều bằng hàm [np.newaxis constant](https://numpy.org/doc/stable/reference/constants.html?highlight=new%20axis#numpy.newaxis):

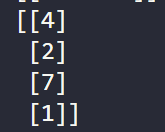
A picture containing graphical user interface

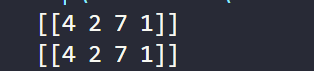
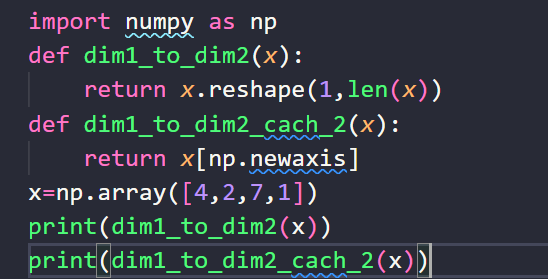
Description automatically generated

Ta có thể biến mảng 1 chiều thành mảng 2 chiều với 1 cột duy nhất ta làm:



Kết quả:





# 11. Đổi từ mảng 1 chiều sang cột trong mảng 2 chiều

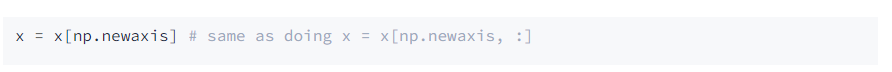
Chạy hàm row\_to\_col(), hàm cần input mảng 1 chiều và trả về mảng 2 chiều với 1 cột bao gồm giá trị cũ.

A picture containing background pattern

Description automatically generated

Một cách để đạt được điều này là sử dụng ndarray.reshape (). Một cách thay thế để biến mảng 1 chiều thành mảng 2 chiều là sử dụng hằng số np.newaxis.

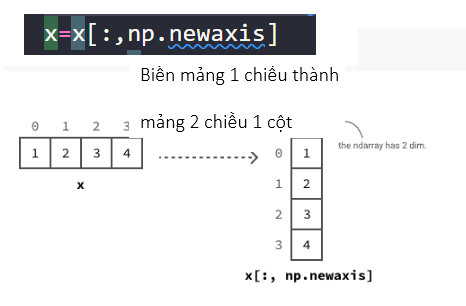
Để làm mảng 1 chiều thành mảng 2 chiều ta có thể làm

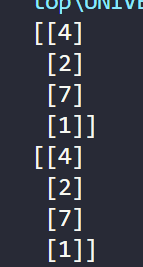
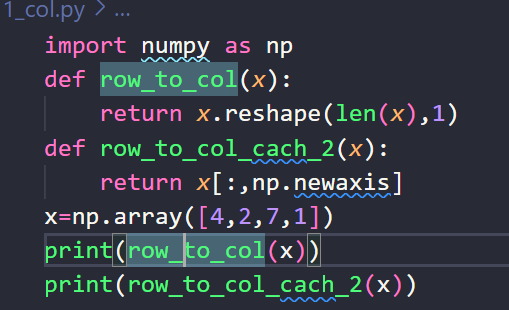


Diagram

Description automatically generated

Để biến mảng 1 chiều thành mảng 2 chiều với một cột duy nhất, chúng ta có thể thực hiện:





# 

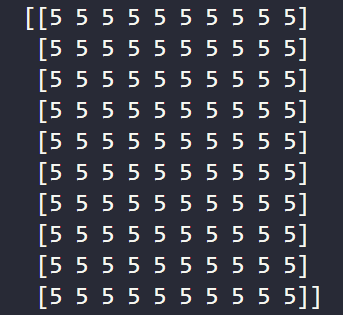
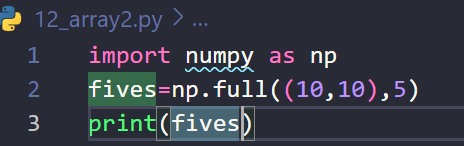
# 

# 12. Tạo mảng hằng n chiều

Tạo mảng 10x10 mỗi phần tử bằng 5:

Dùng [numpy.full() function](https://numpy.org/doc/stable/reference/generated/numpy.full.html?highlight=full#numpy.full) để làm:

Assign to a variable fives, an ndarray with shape (10, 10) whose entries are all equal to 5.



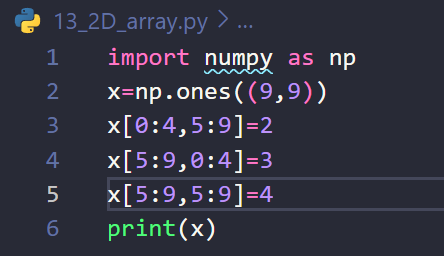
# 13. Tạo mảng 2 chiều

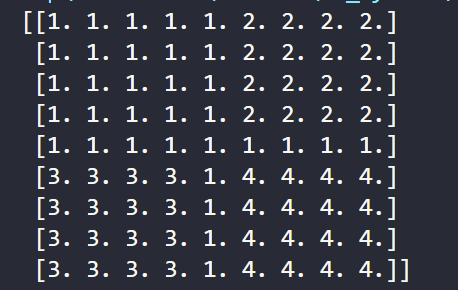
Tạo 1 mảng như hình ở dưới :

Table

Description automatically generated with medium confidence

Tạo chương trình ndarray trong sơ đồ. Gán kết quả cho một biến x.





# 14. Lợp mảng tạo thành mảng mới

Tạo 1 mảng như bên dưới:

Table

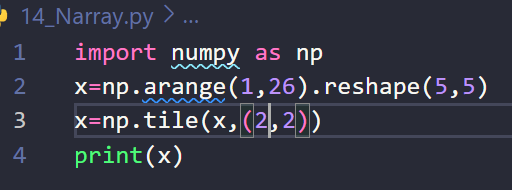
Description automatically generated

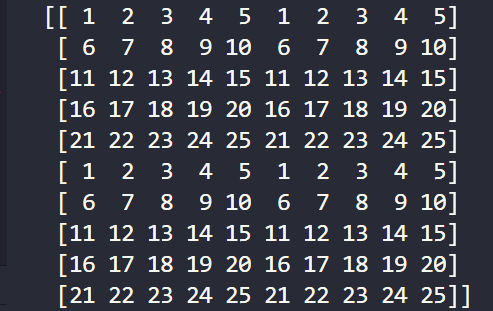
Một hàm có thể giúp bạn đạt được điều này là hàm numpy.tile (). Nó cho phép ta xây dựng một ndarray bằng cách lặp lại một ndarray đã cho. Sơ đồ sau đây cho thấy một ví dụ trong đó chúng ta sử dụng hàm numpy.tile () để xếp một mảng 2 x 2.

Table

Description automatically generated

Đối số thứ hai trong hàm numpy.tile () là shape (2, 3). Đối số này chỉ định rằng chúng ta muốn sao chép x hai lần theo chiều dọc và ba lần theo chiều ngang.





Giải thích: ta tạo mảng x 1 chiều gán các giá trị từ 1 đến 25, sau đó reshape lại thành mảng 2 chiều 5 hàng 5 cột. Cuối cùng là dùng hàm tile(x,(2,2)) để sao chép x 2 lần theo chiều dọc và 2 lần theo chiều ngang