# Python NumPy Tutorial

Check numpy install or not >>Import numpy as np

B >> add new cells

Ctr+enter >> execute code

# >> to create headers

Add html code also in code

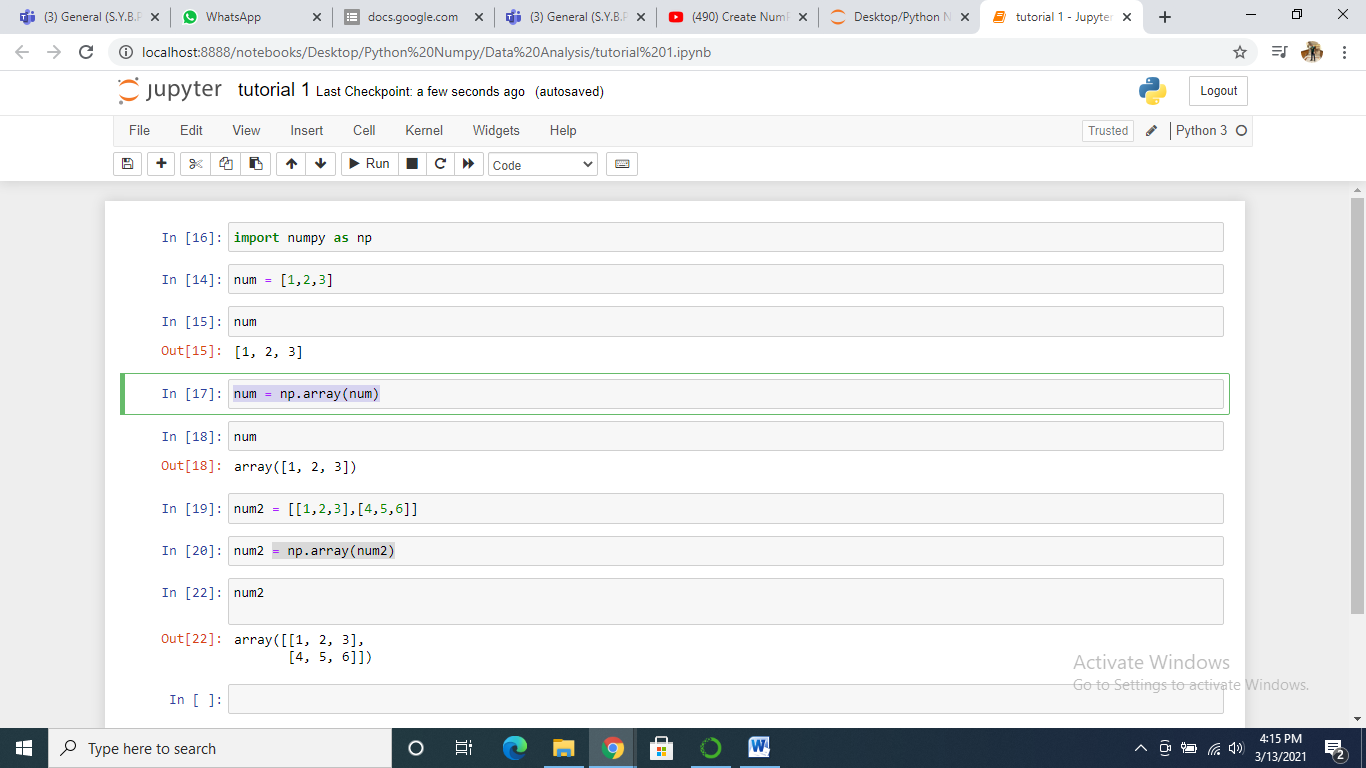
Commands :

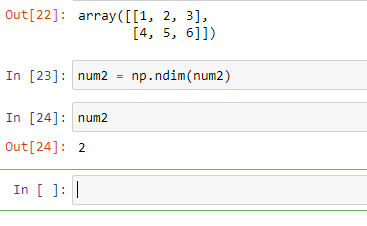
* num = np.array(num)
* num = np.ndim(num) >> to get the dimension 1D, 2D, 3D

1D>> array

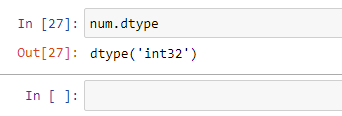
2D>> Vector

3D>> Tensor >> muti dimension array





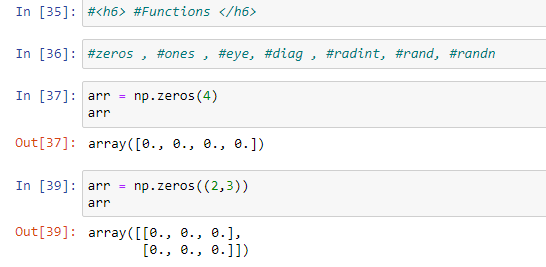
* num.shape
* num.dtype >> to get the datatype



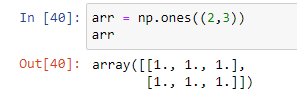
Functions

#zeros , #ones , #eye, #diag , #radint, #rand, #randn

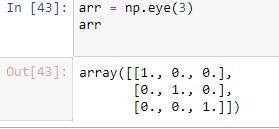
* #zeros ,



* #ones

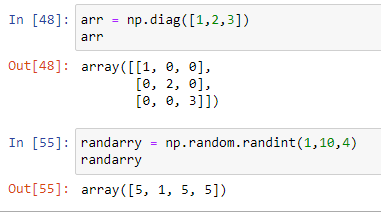


* # eye >> diagonal 1 other 0



#diag >> put diagonal value

#randint >> random.randint(1,10,4) >>1 is from , 10 is to and 4 is no of values



1 is from , 10 is to and 4 is no of values

#rand , #randn, #mean

Random function is import from Numpy Library :

Rand() : values begins 0 and less than 1 -- [0,1] uniform distribution – non negative

Randn(): normal distribution [-,+] – negstive no included

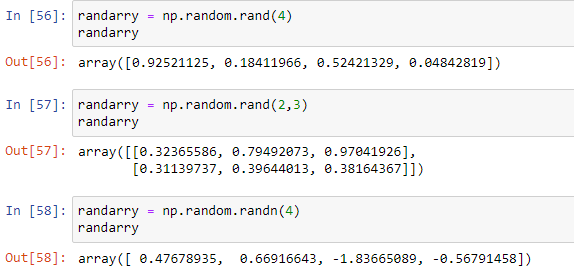
Ranf() : random\_sample

Randint(1,10,5,int) [from,to,size,dtype]

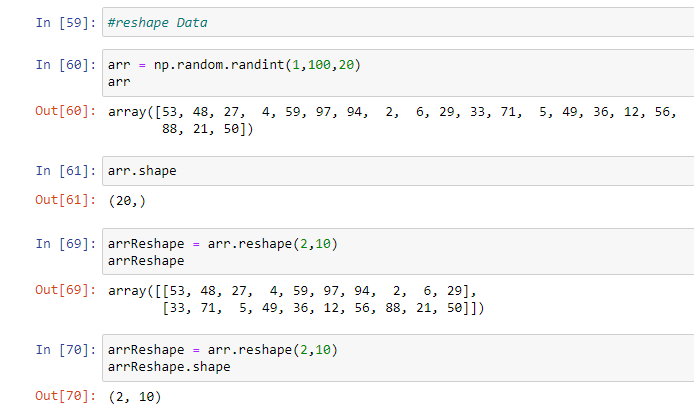
# rand >> return from 0 to 1

#randn >> return close to 0

#mean >> return average



### Reshape data

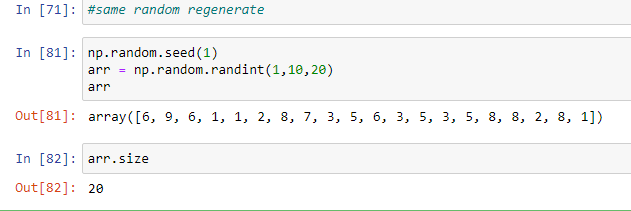


Np.arrange

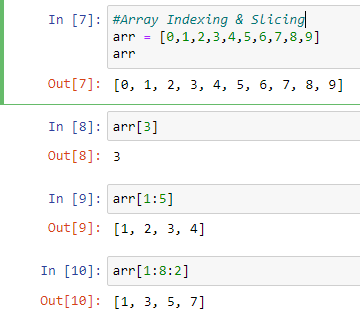
Np.size

Np.reshape(2,-1) >>2 indicates no of rows and -1 indicates the multiplication oof the 2 which same as the results like the array size is 20 >> 2 \* 10 >> 20.

### Same random number regenerate

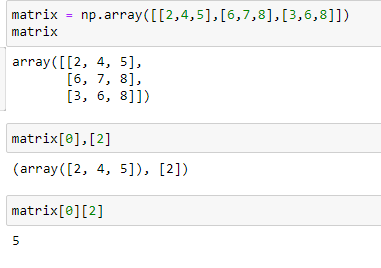


##Array Indexing & Slicing



Matrix

2-d Array.

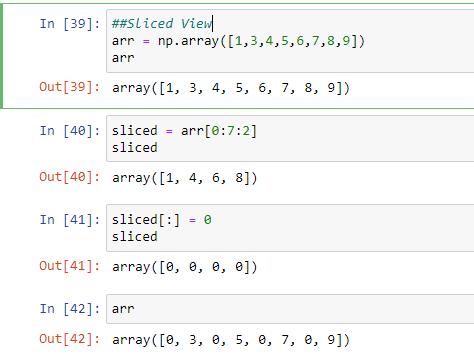


## View & copy

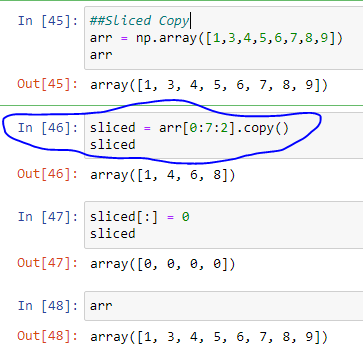
View same array use karta hai and vo alag se memory nahi leta hai same array ko use karta hai and usme hi change karta hai

And copy method me vo array ka copy create karta hai and alag se memory bhi store karta hai.

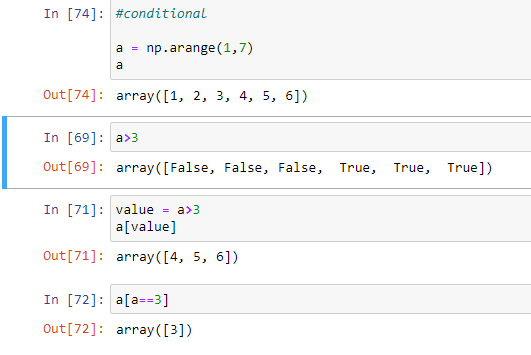
#view



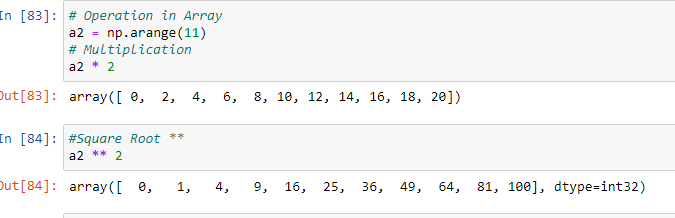
#copy



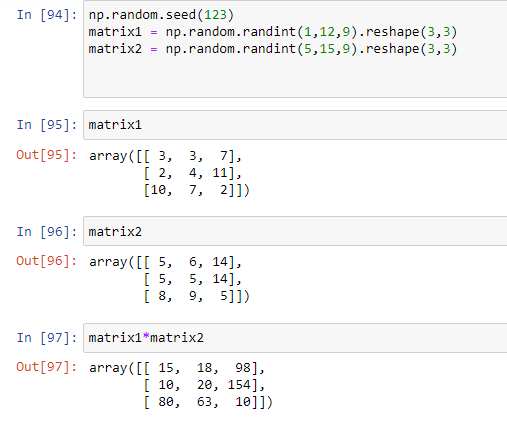
#Conditional



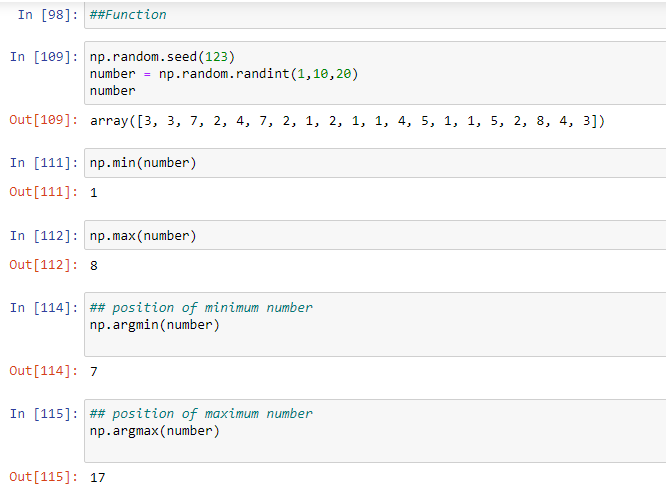
Operation Addition, Multiplication, Square root, Divide

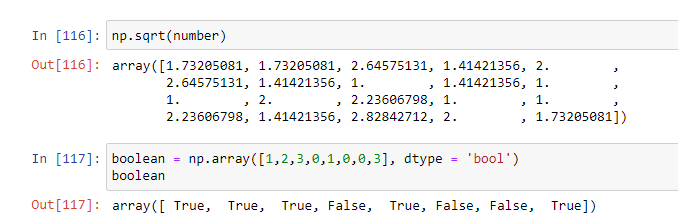


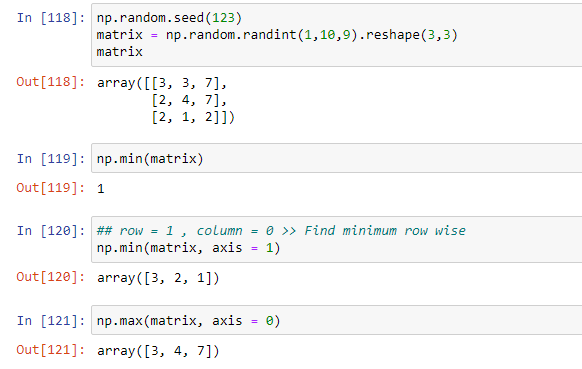
Operation on multi Array



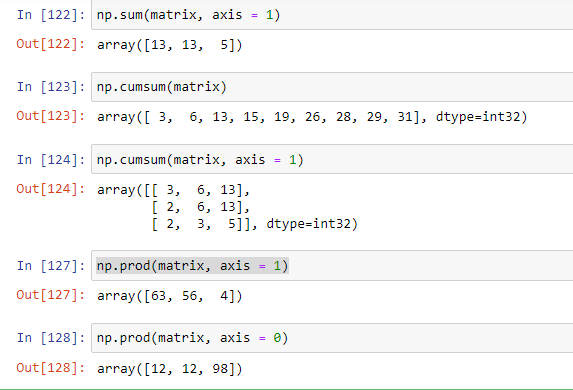
Function >> min, max, argmin, argmax, sqrt



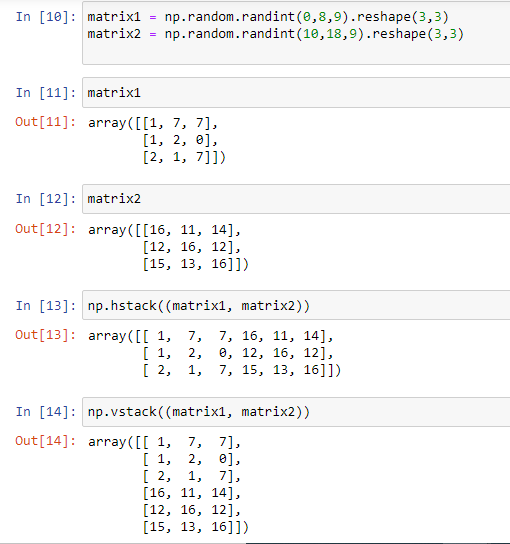




Axis row = 1 and column =0



Horizontal and vertical stack.



Function remaing in the lecture

Alternatative hstack and vstack is np.concatenate([a, b], axis=0/1)