

## 1. Creat a numpy array contaning the number from 1 to 10

```
In [5]: import numpy as np
arr1=np.arange(1,11)
print("Array from 1 to 10:",arr1)
```

Array from 1 to 10: [ 1 2 3 4 5 6 7 8 9 10]

## 2.convert a given python list to NumPy array.

```
In [7]: import numpy as np
Python_list=[1,2,3,4,5]
arr2=np.array(Python_list)
print("a NumPy array:",arr2)
```

a NumPy array: [1 2 3 4 5]

## 3.creat 50 evenly spaced numbers between 1 and 10.

```
In [9]: import numpy as np
arr3=np.linspace(1,10,50)
print("50 evenly spaced numbers between 1 and 10:",arr3)
```

50 evenly spaced numbers between 1 and 10: [ 1. 1.18367347 1.36734694 1.55102041 1.73469388 1.91836735 2.10204082 2.28571429 2.46938776 2.65306122 2.83673469 3.02040816 3.20408163 3.3877551 3.57142857 3.75510204 3.93877551 4.12244898 4.30612245 4.48979592 4.67346939 4.85714286 5.04081633 5.2244898 5.40816327 5.59183673 5.7755102 5.95918367 6.14285714 6.32653061 6.51020408 6.69387755 6.87755102 7.06122449 7.24489796 7.42857143 7.6122449 7.79591837 7.97959184 8.16326531 8.34693878 8.53061224 8.71428571 8.89795918 9.08163265 9.26530612 9.44897959 9.63265306 9.81632653 10.]

## 4.creat a 5\*5 matrix which contains random samples from standard normal distribution.

```
In [10]: import numpy as np
arr4=np.random.rand(5,5)
print(arr4)

[[0.87951994 0.30163818 0.89643925 0.41668598 0.07000324]
 [0.44138137 0.55615812 0.5233967 0.12620528 0.067751 ]
 [0.25604845 0.83884744 0.69691191 0.7466793 0.97425659]
 [0.40107255 0.1554991 0.43159004 0.45767966 0.48983731]
 [0.61689392 0.52757085 0.53561927 0.6809763 0.3550213 ]]
```

## 5.creat 20 random integer numbers between 1 to 100 as a numpy array.

```
In [11]: import numpy as np
arr5=np.random.randint(1,101,20)
print(arr5)

[80  3 29 68 15 60  2 68 21 51 85 42 93 68 94 94 14 37 81 29]
```

## 6.given the numpy array 'arr',reverse its elements and find its size

```
In [12]: import numpy as np
arr6=np.array([1,2,3,4,5])
arr6_reversed=arr6[::-1]
arr6_size=arr6.size
print("reversed arr:",arr6_reversed,"size of array:",arr6_size)

reversed arr: [5 4 3 2 1] size of array: 5
```

## 7.find the mean median and standard deviation of the following numpy array.

```
In [14]: import numpy as np
arr7=np.array([1,2,3,4,5,6,7,8,9,10])
mean=np.mean(arr7)
median=np.median(arr7)
std_dev=np.std(arr7)
print(mean)
print(median)
print(std_dev)

5.5
5.5
2.8722813232690143
```

## 8.creat a 3\*3 matrix with all values set to 1.

```
In [15]: import numpy as np  
arr8=np.ones((3,3))  
print(arr8)
```

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

## 9.creat a 3\*3 matrix with all values set to 0.

```
In [17]: import numpy as np  
arr9=np.zeros((3,3))  
print(arr9)
```

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

## 10.given a two numpy arrays arrr1 and arr2 ,concatenate them horizontally

```
In [18]: import numpy as np  
arr10_1=np.array([1,2,3])  
arr10_2=np.array([4,5,6])  
arr10=np.hstack((arr10_1,arr10_2))  
print(arr10)
```

```
[1 2 3 4 5 6]
```

## 11.creat a nmpy array contaning all even numbers from 0 10 20

```
In [20]: import numpy as np  
arr11=np.arange(0,21,2)  
print(arr11)
```

```
[ 0  2  4  6  8 10 12 14 16 18 20]
```

## 12.perform elements-wise multiplication of two numpy arrays 'a' and 'b'.

```
In [21]: import numpy as np
a=np.array([1,2,3])
b=np.array([4,5,6])
element_wise_mult=a*b
print(element_wise_mult)
```

[ 4 10 18]

## 13.Reshape the numpy array into a 2\*3 matrix

```
In [22]: import numpy as np
arr13=np.array([1,2,3,4,5,6])
reshaped=arr13.reshape(2,3)
print(reshaped)
```

[[1 2 3]  
[4 5 6]]

## 14.Find the maximum and minimum values in the numpy array.

```
In [24]: import numpy as np
arr14=np.array([1,3,7,2,9])
max_value=np.max(arr14)
min_value=np.min(arr14)
print("max:",max_value)
print("min:",min_value)
```

max: 9  
min: 1

## 15.calculate the dot product of two numpy arrays x and y.

```
In [25]: import numpy as np
x=np.array([1,2])
y=np.array([3,4])
dot_product=np.dot(x,y)
print(dot_product)
```

11

## 16.creat a 2D numpy array with shape(3,4) filled with random floating point numbers between 0 and 1.

```
In [26]: import numpy as np
arr16=np.random.rand(3,4)
print(arr16)

[[0.92939046 0.00510697 0.26970947 0.3652704 ]
 [0.0899147  0.0084585  0.40606984 0.16125173]
 [0.82181083 0.32847676 0.67444287 0.01586925]]
```

## 17.Transpose a 2D Numpy array

```
In [27]: import numpy as np
arr17=np.array([[1,2],[3,4],[5,6]])
transpose_arr17=arr17.T
print(transpose_arr17)

[[1 3 5]
 [2 4 6]]
```

## 18.create a slice(0 to 5 index)of a given array and set the value 100 to the slice

```
In [28]: import numpy as np
arr18=np.array([1,2,3,4,5,6,7,8])
arr18[0:5]=100
print(arr18)

[100 100 100 100 100   6   7   8]
```

## 19.slice the first two rows and the last two columns of a 2D numpy array(right top corner).

```
In [30]: import numpy as np
arr19=np.array([[1,2,3,4],[5,6,7,8],[9,10,11,12]])
slice19=arr19[:2,-2:]
print(slice19)

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

## 20.given a numpy array of numbers,how can you filter out all values greater than 5.

```
In [31]: import numpy as np
arr20=np.array([1,2,3,4,5,6,7,8])
filtered_arr20=arr20[arr20<=5]
print(filtered_arr20)
```

```
[1 2 3 4 5]
```

## 21.using boolean indexing,extract all even numbers from a numpy array

```
In [32]: import numpy as np
arr21=np.array([1,2,3,4,5,6,7,8])
even_arr21=arr21[arr21%2==0]
print(even_arr21)
```

```
[2 4 6 8]
```