

Ex no:1 Name:NUMPY

Aim :

To install Numpy package and do the basic functions

Description:

1. Declare the Numpy array
2. Create an array with full of zero values
3. Create an array with a Scalar values filled
4. Create an array with random values
5. Reshape and Flattening the array.
6. Convert an array from one type to another.
7. Do slicing operations in an array.
8. Do join functions (join,horizontal join,vertical join and depth join)
9. Do index retrivel and basic operation with respect to index
10. Sorting operation of an array
11. Filtering operation based on array value
12. Vector Operation - Addition ,Subtraction, Multiplication and Division
13. Scalar Operation and Vectorize operation.

#1

```
import numpy as np
```

```
a=np.array([[1,2,4],[5,8,7]])
```

```
print(a)
```

```
o/p: [[1 2 4]
       [5 8 7]]
```

#2

```
b=np.zeros((3,4))
```

```
print(b)
```

```
o/p: [[0. 0. 0. 0.]
       [0. 0. 0. 0.]
       [0. 0. 0. 0.]]
```

#3

```
c=np.full((3,3),6)
```

```
print(c)
```

```
o/p: [[6 6 6]
       [6 6 6]
       [6 6 6]]
```

#4

```
d=np.random.random((2,2))
```

```
print(d)
```

```
o/p: [[0.45156484 0.40124375]
      [0.74302574 0.45256523]]
```

```
#5
```

```
a=np.array([[1,2,3,4],[5,2,4,2],[1,2,0,1]])
```

```
na=a.reshape(4,3)
```

```
f=a.flatten()
```

```
print(a)
```

```
print(na)
```

```
print(f)
```

```
o/p: [[1 2 3 4]
      [5 2 4 2]
      [1 2 0 1]]
[[1 2 3]
 [4 5 2]
 [4 2 1]
 [2 0 1]]
[1 2 3 4 5 2 4 2 1 2 0 1]
```

```
#6
```

```
nt=a.astype('f')
```

```
print(nt)
```

```
o/p: [[1. 2. 3. 4.]
      [5. 2. 4. 2.]
      [1. 2. 0. 1.]]
```

```
#7
```

```
arr=np.array([[1,2,3,4],[5,2,4,2],[3,5,8,9],[5,9,2,0],[1,2,0,1]])
```

```
print(arr[0:4:2])
```

```
print(arr[:,3])
```

```
print(arr[2:,2:])
```

```
print(arr[2,3])
```

```
print(arr[3:])
```

```
print(arr[:,3])
```

```
o/p:
```

```
[ [1 2 3 4]
  [3 5 8 9]]
[ [1 2 3]
  [5 2 4]]
[ [8 9]
  [2 0]]
```

```

[0 1]]
9
[[5 9 2 0]
 [1 2 0 1]]
[[1 2 3]
 [5 2 4]
 [3 5 8]
 [5 9 2]
 [1 2 0]]
#8
a1=np.array([1,2,3])
a2=np.array([4,5,6])
print(np.concatenate((a1,a2)))
print(np.hstack((a1,a2)))
print(np.vstack((a1,a2)))
print(np.dstack((a1,a2)))
o/p: [1 2 3 4 5 6]

```

```

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

```

```

#9
a=np.array([1,2,4,6,5,4])
print(np.where(a==4))
print(np.where(a%2==0))
o/p: (array([2, 5]),)

(array([1, 2, 3, 5]),)

```

```

#10
d=np.sort(a)
print(d)
o/p: [1 2 4 4 5 6]

```

```

#11
a=np.array([41,42,43,44])
b=[True,False,True,False]
na=a>42
newa=a[na]
print(na)
print(newa)
O/p: [False False True True]
[43 44]

```

```
#12
a=np.array([1,2,3,4,5,6])
b=np.array([7,8,9,10,11,12])
print(a+b)
print(a-b)
print(a*b)
print(a/b)
print(a%b)
o/p: [ 8 10 12 14 16 18]
[-6 -6 -6 -6 -6 -6]
[ 7 16 27 40 55 72]
[0.14285714 0.25      0.33333333 0.4      0.45454545 0.5      ]
[1 2 3 4 5 6]
```

```
#13 (a)
a=np.array([1,2,3,4,5,6])
print(a)

import numpy as np

def my_function(x, y):
    if x > y:
        return x - y
    else:
        return x + y
vec_function = np.vectorize(my_function)

a = np.array([1, 5, 3])
b = np.array([4, 2, 6])

result = vec_function(a, b)
print(result)
o/p: [5 3 9]
```