

# Practical - 4

## AIM

Create/Define single dimension / multidimensional arrays, and arrays with specific values like array of all ones, all zeros, array with random values within a range, or a diagonal matrix.

## CODE & OUTPUT

```
In [12]: # a single dimension array
a = [1, 2, 3, 4, 5]
print(a)
pets = ["cat", "dog", "parrot", "penguin"]
print(pets)
```

```
[1, 2, 3, 4, 5]
['cat', 'dog', 'parrot', 'penguin']
```

```
In [2]: # multi dimension array
a = [
    [1, 2, 3, 5],
    [6, 7, 8, 9]
]
print(a)
```

```
[[1, 2, 3, 5], [6, 7, 8, 9]]
```

```
In [3]: # a heterogeneous array
b = [
    ["abc", 2, 3, "def"],
    ["lmn", 65, 78, "pqr"],
    [2, 3, "rtq"]
]
print(b)
```

```
[['abc', 2, 3, 'def'], ['lmn', 65, 78, 'pqr'], [2, 3, 'rtq']]
```

```
In [15]: import numpy as np
# array of all one's

all_ones = np.ones(5)
print(all_ones)
```

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

```
In [16]: sampl = np.random.uniform(low=0.5, high=13.3, size=(50,))
print(sampl)
```

```
[ 3.00390644  6.86060384  3.73487058  2.54787728  5.33702127 10.04217568
  4.70741437  2.97739194 11.17034491  0.74343435 11.92874866  9.79987514
 12.48589661 12.10019853  3.03320249  6.44899118  5.99595095  3.62402906
  2.16674465  4.64316812 11.81884631 11.77653742  9.47225222  1.73536195
  6.58311688  8.99752053 12.94227914 11.43925206 13.12735881 11.82238161
  8.31027989  5.17438522  6.00322935  7.63086345  5.04591807  1.09850169
 13.20911789  5.51307101  9.43694768 10.73630036  6.94277879  8.69868398
  6.97500634  1.60495441  4.24731595  8.105219    7.49581479  6.78191176
  2.84891337  9.77626211]
```

```
In [18]: # array of zeros
all_zeros = np.zeros(6)
print(all_zeros)
```

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

```
In [19]: # array with a diagonal matrix
x = np.arange(9).reshape((3,3))
print(x)
```

```
[[0 1 2]
 [3 4 5]
 [6 7 8]]
```