Numpy assignment

1)Use NumPy to generate an array of 25 random numbers sampled from a standard normal distribution.

In [1]:

```
import numpy as np
rand_num = np.random.normal(0,1,25)
print("25 random numbers from a standard normal distribution:")
print(rand_num)
```

```
25 random numbers from a standard normal distribution:
[-1.39124365  0.14131823  0.39604709  1.09302266  1.33046025  0.17170622
 -0.30654235 -0.34352281 -2.887695   -0.52519274 -0.27050036 -1.27810899
  0.27078124  0.07480362  1.96369355 -1.3783626  -0.39804067  0.76004541
 -0.8723479   0.42667167 -0.00459128  0.07614662  0.63667285 -0.71017903
  1.64759275]
```

2)Create a random vector of size 30 and find the mean value.

In [3]:

```
Z = np.random.random(30)
m = Z.mean()
print (m)
```

```
0.5318107481805653
```

```
3)Insert 1 to 100 numbers in a NumPy array and reshape it to 10*10 matrix.
```

In [17]:

```
Z = np.arange(1,101).reshape(10,10)
print (Z)
```

```
[[  1   2   3   4   5   6   7   8   9  10]
 [ 11  12  13  14  15  16  17  18  19  20]
 [ 21  22  23  24  25  26  27  28  29  30]
 [ 31  32  33  34  35  36  37  38  39  40]
 [ 41  42  43  44  45  46  47  48  49  50]
 [ 51  52  53  54  55  56  57  58  59  60]
 [ 61  62  63  64  65  66  67  68  69  70]
 [ 71  72  73  74  75  76  77  78  79  80]
 [ 81  82  83  84  85  86  87  88  89  90]
 [ 91  92  93  94  95  96  97  98  99 100]]
```

4)Create a 10x10 array with random values and find the minimum and maximum values.

In [18]:

```
import numpy as np
x = np.random.random((10,10))
print("Original Array:")
print(x)
xmin, xmax = x.min(), x.max()
print("Minimum and Maximum Values:")
print(xmin, xmax)
```

```
Original Array:
[[0.42958291 0.07799276 0.68211344 0.38122548 0.34884835 0.50938741
  0.42761716 0.83909032 0.44742466 0.34966489]
 [0.70793123 0.93000261 0.692127   0.71952492 0.79529372 0.31084325
  0.58410681 0.36231693 0.17187315 0.09459983]
 [0.52270484 0.96386301 0.4657373  0.74279846 0.53783666 0.63370708
  0.7394654  0.53666374 0.83073461 0.05127695]
 [0.79380969 0.74103158 0.31887964 0.40745744 0.17224898 0.46374739
  0.64596553 0.17275941 0.34458568 0.86453478]
 [0.12130235 0.24463477 0.76700109 0.56181797 0.29721155 0.14945271
  0.9421712  0.65073313 0.73113604 0.03014493]
 [0.88543532 0.21623579 0.30344035 0.05216048 0.75008815 0.55784177
  0.35271898 0.44036633 0.97731063 0.4013406 ]
 [0.67939387 0.10095569 0.40249977 0.92447415 0.66157968 0.61735195
  0.86405752 0.15088676 0.13470482 0.25205887]
 [0.20772228 0.91623273 0.6493833  0.54761979 0.24063287 0.14185405
  0.33488348 0.24780715 0.27841309 0.97498064]
 [0.63833624 0.64124441 0.39154434 0.33030444 0.19248421 0.25300167
  0.76242535 0.59728832 0.70640788 0.00889326]
 [0.61265711 0.07005039 0.4533872  0.65601608 0.41180399 0.65842036
  0.28829233 0.18562964 0.83491923 0.0295568 ]]
Minimum and Maximum Values:
0.008893261643908357 0.9773106327590351
```

5)Find Dot product of two arrays f = np.array([1,2]) g = np.array([4,5])

In [19]:

```
f = np.array([1,2])
g = np.array([4,5])
np.dot(f,g)
```

Out[19]:

```
14
```

6)Concatenate following arrays along axis=0 x=np.array([[1,2],[3,4]]) y=np.array([[5,6]])

In [20]:

```
x = np.array([[1, 2], [3, 4]])
y = np.array([[5, 6]])
np.concatenate((x, y), axis=0)
```

Out[20]:

```
array([[1, 2],
       [3, 4],
       [5, 6]])
```

7)How to get the common items between two python NumPy arrays?
a = np.array([1,2,3,2,3,4,3,4,5,6]) b = np.array([7,2,10,2,7,4,9,4,9,8])

In [22]:

```
a = np.array([1,2,3,2,3,4,3,4,5,6])
b = np.array([7,2,10,2,7,4,9,4,9,8])
print(np.intersect1d(a, b))
```

```
[2 4]
```

8)Sort the numpy array: arr = np.array([10,5,8,4,7,2,3,1])

In [27]:

```
import numpy as np
x = np.array([10,5,8,4,7,2,3,1])
def selection_sort(x):
    for i in range(len(x)):
        swap = i + np.argmin(x[i:])
        (x[i], x[swap]) = (x[swap], x[i])
    return x
selection_sort(x)
```

Out[27]:

```
array([ 1,  2,  3,  4,  5,  7,  8, 10])
```