

'''

Experiment 11 : Working with NumPy arrays.

Name : Khan Arshad Abdulla

Roll No : 20CO24

Academic Year : 2021-22

THEORY:

NumPy is a Python library used for working with arrays.

It also has functions for working in domain of linear algebra, fourier transform, and matrices.

NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

NumPy stands for Numerical Python.

In Python we have lists that serve the purpose of arrays, but they are slow to process.

NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.

The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.

Arrays are very frequently used in data science, where speed and resources are very important.

NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently.

This behavior is called locality of reference in computer science.

This is the main reason why NumPy is faster than lists. Also it is optimized to work with latest CPU architectures.

'''

```
import numpy as np
```

```
#creating numpy arrays
```

```
#create 1D and 2D arrays using array function
```

```
arr1d = np.array([1,2])
```

```
arr2d = np.array([[1,2,3],[4,5,6],[7,8,9]],dtype=float)
```

```
#create array using arange function
```

```
arr_ar = np.arange(1,100,9)
```

```
#create array using linspace function
```

```
arr_lp = np.linspace(10,11,10)
```

```
#create array using logspace function
```

```
arr_logp = np.logspace(10,15,10)
```

```
#create array using various functions
```

```
arr_zeros = np.zeros((2,3))
```

```
arr_ones = np.ones((3,2))
```

```
arr_empty = np.empty((2,4))
```

```
arr_identity = np.identity(3)
```

```
'''
```

```
#accessing array elements
```

```
print('arr1d =',arr1d)
```

```
print('arr2d =')
```

```
for row in arr2d:
```

```
    print(row)
```

```
print('arr_ar =',arr_ar)
```

```
print('arr_lp =',arr_lp,' having size of',arr_lp.size)
```

```
print('arr_logp =',arr_logp,' having size of',arr_logp.size)
```

```
print('arr_zeros =\n',arr_zeros)
```

```
print('arr_ones =\n',arr_ones)
```

```
print('arr_empty =\n',arr_empty)
```

```
print('arr_identity =\n',arr_identity)
```

```
'''
```

```
#performing array operations
```

```
print('\nPerforming Array Operations:')
```

```
print('Addition of 2D arrays\n',arr2d + arr_identity)
```

```
print('Substraction of 2D arrays\n',arr2d - arr_identity)
```

```
print('Multiplication of 2D arrays\n',arr2d * arr_identity)
```

```
print('Matrix Multiplication of 2D arrays\n',arr2d @ arr_identity)
```

```
print('Transpose of 2D arrays\n',arr2d.transpose())
```

```
#performing slice operations
```

```
print('Elements of Second Row and First 2 Columns',arr2d[1,:2])
```

```
print('Elements of Last Row and Second Column onwards',arr2d[-1,1:])
```

```
print(arr2d[arr1d])
```

```
#performing reshape on arrays
```

```
#Converting 1-D array into 2-D array
```

```
reshape_1darray = np.array([1,2,3,4,5,6,7,8,9])
```

```
converted = reshape_1darray.reshape(3,3)
```

```
print("Reshaped 2-D Array:")
```

```
print(converted)
```

```
#Converting 1-D array into 3-D array
```

```
one_darray = np.array([1,2,3,4,5,6,7,8,9,10,11,12])
```

```
arr3d = one_darray.reshape(2,2,3)
```

```
print("Reshaped 3-D Array:")
```

```
print(arr3d)
```

```
#Converting 2-D array into 1-D array
```

```
array2d = np.array([[1,2,3],[4,5,6],[7,8,9]])
```

```
array1d = array2d.reshape(9)
```

```
print("2-D Array:")
```

```
print(array2d)
print("Reshaped 1-D Array:")
print(array1d)
```

'''

OUTPUT:

Performing Array Operations:

Addition of 2D arrays

```
[[ 2.  2.  3.]
 [ 4.  6.  6.]
 [ 7.  8. 10.]]
```

Substraction of 2D arrays

```
[[0.  2.  3.]
 [4.  4.  6.]
 [7.  8.  8.]]
```

Multiplication of 2D arrays

```
[[1.  0.  0.]
 [0.  5.  0.]
 [0.  0.  9.]]
```

Matrix Multiplication of 2D arrays

```
[[1.  2.  3.]
 [4.  5.  6.]
 [7.  8.  9.]]
```

Transpose of 2D arrays

```
[[1.  4.  7.]
 [2.  5.  8.]
 [3.  6.  9.]]
```

Elements of Second Row and First 2 Columns [4. 5.]

Elements of Last Row and Second Column onwards [8. 9.]

```
[[4.  5.  6.]
```

```
[7. 8. 9.]
```

Reshaped 2-D Array:

```
[[1 2 3]
```

```
[4 5 6]
```

```
[7 8 9]]
```

Reshaped 3-D Array:

```
[[[ 1  2  3]
```

```
 [ 4  5  6]]
```

```
[[ 7  8  9]
```

```
 [10 11 12]]]
```

2-D Array:

```
[[1 2 3]
```

```
[4 5 6]
```

```
[7 8 9]]
```

Reshaped 1-D Array:

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

CONCLUSION: In this experiment we have successfully implemented numpy library and learned how to create numpy arrays and reshape the array from 1 dimension to another.

'''