Department of Engineering Sciences and Technology,

Second Year Btech in Computer Science

Project Based Learning-Python

Assignment - 09

Name - Paritosh kolwadkar

SRN – 31231313

Roll no – 39

Batch – D2

Problem statement : **Write a program to create a NumPy array and reshape it into different dimensions. Transpose of the reshaped array and demonstrate swapping axes.**

Pre-requisites :Knowledge of NumPy library functions such as `reshape`, `transpose`, and `swapaxes`.

Understanding of array dimensions and matrix operations.

Installed NumPy library (`pip install numpy`).

Code:

```
# Import NumPy library

import numpy as np



# Create a 1D NumPy array

original_array = np.arange(1, 13)

print("Original Array (1D):")

print(original_array)

print(f"Shape: {original_array.shape}, Dimensions: {original_array.ndim}\n")
```

```python
# Reshape the array into 2D (3x4)
reshaped_2d = original_array.reshape(3, 4)
print("Reshaped Array (2D - 3x4):")
print(reshaped_2d)
print(f"Shape: {reshaped_2d.shape}, Dimensions: {reshaped_2d.ndim}\n")

# Reshape the array into 3D (2x2x3)
reshaped_3d = original_array.reshape(2, 2, 3)
print("Reshaped Array (3D - 2x2x3):")
print(reshaped_3d)
print(f"Shape: {reshaped_3d.shape}, Dimensions: {reshaped_3d.ndim}\n")

# Transpose the 2D array
transposed_2d = reshaped_2d.T
print("Transpose of the Reshaped 2D Array:")
print(transposed_2d)
print(f"Shape: {transposed_2d.shape}, Dimensions: {transposed_2d.ndim}\n")

# Swap axes of the 3D array (swap axes 0 and 2)
swapped_axes = reshaped_3d.swapaxes(0, 2)
print("3D Array with Swapped Axes (0 and 2):")
print(swapped_axes)
print(f"Shape: {swapped_axes.shape}, Dimensions: {swapped_axes.ndim}")
```

Explanation :

- **Creating a NumPy Array**:
  - `np.arange(1, 13)`: Creates a 1D array with values from 1 to 12.
- **Reshaping Arrays**:
  - `reshape(3, 4)`: Converts the array into a 2D array with 3 rows and 4 columns.
  - `reshape(2, 2, 3)`: Converts the array into a 3D array with dimensions 2x2x3.
- **Transpose of a 2D Array**:
  - `T`: Transposes the 2D array, flipping rows and columns.
- **Swapping Axes of a 3D Array**:
  - `swapaxes(0, 2)`: Swaps the 0th axis (depth) with the 2nd axis (columns).

Output:

Original Array (1D):

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

Shape: (12,), Dimensions: 1

Reshaped Array (2D - 3x4):

[[1 2 3 4]

[5 6 7 8]

[9 10 11 12]]

Shape: (3, 4), Dimensions: 2

Reshaped Array (3D - 2x2x3):

[[[1 2 3]

[4 5 6]]

[[7 8 9]

[10 11 12]]]

Shape: (2, 2, 3), Dimensions: 3

Transpose of the Reshaped 2D Array:

[[1 5 9]

[2 6 10]

[3 7 11]

```
 [ 4  8 12]]
```

Shape: (4, 3), Dimensions: 2

3D Array with Swapped Axes (0 and 2):

```
[[[ 1  7]
  [ 4 10]]

 [[ 2  8]
  [ 5 11]]

 [[ 3  9]
  [ 6 12]]]
```

Shape: (3, 2, 2), Dimensions: 3

Output Explained:

Original Array:

- A 1D array with 12 elements.

Reshaped Arrays:

- 2D (3x4): The array is converted into 3 rows and 4 columns.
- 3D (2x2x3): The array is split into two 2x3 matrices stacked on top of each other.

Transpose:

- Rows and columns of the 2D array are swapped, changing its shape to (4, 3).

Swapped Axes:

- Axes 0 and 2 of the 3D array are swapped, rearranging the structure of the array and altering its shape to (3, 2, 2).