**EX.NO: 6 DATE:**

**Manipulation of NumPy arrays- Indexing, Slicing, Reshaping, Joining and Splitting.**

**AIM:**

To demonstrate various manipulation techniques on NumPy arrays including indexing, slicing, reshaping, joining, and splitting.

**PROCEDURE:**

1. **Indexing:**

Access individual elements in a NumPy array using indices.

1. **Slicing:**

Extract a subarray from a NumPy array using slicing operations.

1. **Reshaping:**

Change the shape of a NumPy array without changing its data.

1. **Joining:**

Combine two or more arrays using functions like concatenate(), vstack(), and hstack().

1. **Splitting:**

Split an array into multiple subarrays using split(), vsplit(), or hsplit().

**PROGRAM:**

import numpy as np

**# 1. Indexing**

**# Creating a 2D array**

array\_2d = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]]) print("Original 2D array:\n", array\_2d)

**# Accessing individual elements**

element = array\_2d[1, 2] # Access element at row 1, column 2 (0-based index) print("Element at index [1, 2]:", element)

**# 2. Slicing**

**# Extracting a subarray**

subarray = array\_2d[0:2, 1:3] # Rows 0 and 1, Columns 1 and 2 print("Sliced subarray:\n", subarray)

**# 3. Reshaping**

**# Reshape a 1D array into a 2D array**

array\_1d = np.array([1, 2, 3, 4, 5, 6, 7, 8])

reshaped\_array = array\_1d.reshape(2, 4) # Reshaping into 2 rows and 4 columns print("Reshaped array (2x4):\n", reshaped\_array)

**# 4. Joining Arrays**

**# Joining two 1D arrays using concatenate**

array\_a = np.array([1, 2, 3])

array\_b = np.array([4, 5, 6])

joined\_array = np.concatenate((array\_a, array\_b)) print("Joined array:", joined\_array)

**# Joining two 2D arrays using vertical stack (vstack)**

array\_c = np.array([[1, 2], [3, 4]])

array\_d = np.array([[5, 6], [7, 8]]) vstacked\_array = np.vstack((array\_c, array\_d))

print("Vertically stacked array:\n", vstacked\_array)

**# Joining two 2D arrays using horizontal stack (hstack)** hstacked\_array = np.hstack((array\_c, array\_d)) print("Horizontally stacked array:\n", hstacked\_array)

**# 5. Splitting Arrays**

**# Splitting a 1D array into 3 parts** split\_array = np.split(array\_1d, 3) print("Split array into 3 parts:", split\_array)

**# Splitting a 2D array vertically**

vsplit\_array = np.vsplit(array\_2d, 3) # Split into 3 subarrays row-wise print("Vertically split 2D array:")

for sub in vsplit\_array: print(sub)

**# Splitting a 2D array horizontally**

hsplit\_array = np.hsplit(array\_2d, 3) # Split into 3 subarrays column-wise print("Horizontally split 2D array:")

for sub in hsplit\_array: print(sub)

**OUTPUT:**

**Original 2D array:**

[[1 2 3]

[4 5 6]

[7 8 9]]

**Element at index [1, 2]: 6 Sliced subarray:**

[[2 3]

[5 6]]

**Reshaped array (2x4):**

[[1 2 3 4]

[5 6 7 8]]

**Joined array: [1 2 3 4 5 6] Vertically stacked array:**

[[1 2]

[3 4]

[5 6]

[7 8]]

**Horizontally stacked array:**

[[1 2 5 6]

[3 4 7 8]]

**Split array into 3 parts: [array([1, 2, 3]), array([4, 5, 6]), array([7, 8])] Vertically split 2D array:**

[[1 2 3]]

[[4 5 6]]

[[7 8 9]]

**Horizontally split 2D array:**

[[1]

[4]

[7]]

[[2]

[5]

[8]]

[[3]

[6]

[9]]

**Result:**

The various manipulation techniques, including indexing, slicing, reshaping, joining, and splitting of NumPy arrays, were successfully demonstrated