

### **Q1. Explain the concept of broadcasting in NumPy with suitable examples.**

Broadcasting allows NumPy to perform arithmetic operations on arrays of different shapes. The smaller array is stretched automatically to match the larger array.

```
a = np.array([1, 2, 3])
b = 5
print(a + b) # [6 7 8]

x = np.array([[1],[2],[3]])
y = np.array([10,20,30])
print(x + y)
# [[11 21 31]
#  [12 22 32]
#  [13 23 33]]
```

### **Q2. Differentiate between a Python list and a NumPy array.**

**Python List:** General purpose, slower, more memory usage.

**NumPy Array:** Numerical, faster, less memory.

```
lst = list(range(1,1000000))
arr = np.array(lst)
np.sum(arr) # Faster than sum(lst)
```

### **Q3. Describe the various indexing techniques in NumPy arrays.**

**Basic Indexing:** arr[2]

**Slicing:** arr[1:3]

**Boolean Indexing:** arr[arr > 20]

**Fancy Indexing:** arr[[0,2,3]]

**Multi-dim Indexing:** mat[1,0]

### **Q4. Write a Python program to create a NumPy array and perform operations.**

```
arr = np.array([1,2,3,4,5,6])
print(arr + 10) # Element-wise addition
print(arr.mean()) # Mean
print(arr.std()) # Standard Deviation
print(arr.reshape(2,3)) # Reshaped to 2x3
```

### **Q5. What is a Pandas Series? How is it different from a NumPy array?**

A Pandas Series is a one-dimensional labeled array. It stores values with an index.

**NumPy Array:** only values.

**Pandas Series:** values + labels.

```
arr = np.array([10,20,30])
series = pd.Series(arr, index=["a","b","c"])
print(series)
# a 10
```

# b 20

# c 30