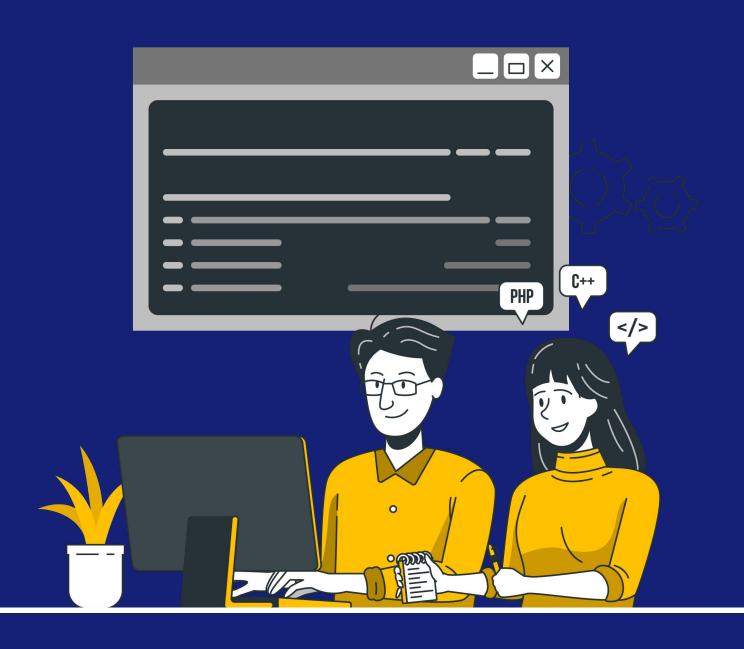


Assignments

Numpy





Theoretical Questions:

- 1. Explain the purpose and advantages of NumPy in scientific computing and data analysis. How does it enhance Python's capabilities for numerical operations?
- 2. Compare and contrast np.mean() and np.average() functions in NumPy. When would you use one over the other?
- 3. Describe the methods for reversing a NumPy array along different axes. Provide examples for 1D and 2D arrays.
- 4. How can you determine the data type of elements in a NumPy array? Discuss the importance of data types in memory management and performance.
- 5. Define ndarrays in NumPy and explain their key features. How do they differ from standard Python lists?
- 6. Analyze the performance benefits of NumPy arrays over Python lists for large-scale numerical operations.
- 7. Compare vstack() and hstack() functions in NumPy. Provide examples demonstrating their usage and output.
- 8. Explain the differences between flipIr() and flipud() methods in NumPy, including their effects on various array dimensions.
- 9. Discuss the functionality of the array_split() method in NumPy. How does it handle uneven splits?
- 10. Explain the concepts of vectorization and broadcasting in NumPy. How do they contribute to efficient array operations?



Practical Questions:

- 1. Create a 3x3 NumPy array with random integers between 1 and 100. Then, interchange its rows and columns.
- 2. Generate a 1D NumPy array with 10 elements. Reshape it into a 2x5 array, then into a 5x2 array.
- 3. Create a 4x4 NumPy array with random float values. Add a border of zeros around it, resulting in a 6x6 array.
- 4. Using NumPy, create an array of integers from 10 to 60 with a step of 5.
- 5. Create a NumPy array of strings ['python', 'numpy', 'pandas']. Apply different case transformations (uppercase, lowercase, title case, etc.) to each element.
- 6. Generate a NumPy array of words. Insert a space between each character of every word in the array.
- 7. Create two 2D NumPy arrays and perform element-wise addition, subtraction, multiplication, and division.
- 8. Use NumPy to create a 5x5 identity matrix, then extract its diagonal elements.
- 9. Generate a NumPy array of 100 random integers between 0 and 1000. Find and display all prime numbers in this array.
- 10. Create a NumPy array representing daily temperatures for a month. Calculate and display the weekly averages.