NumPy Practice Tasks

Simple, Intermediate, and Hard Exercises

Simple Tasks

S1. Create a NumPy array with values from 1 to 20.

S2. Create a 3x3 matrix filled with zeros.

S3. Create a 5x5 identity matrix.

S4. Generate 10 random integers between 1 and 100.

S5. Find the maximum, minimum, and mean of a random array.

S6. Reshape a 1D array of size 12 into a 3x4 matrix.

S7. Create an array of 50 even numbers.

S8. Reverse a NumPy array.

S9. Extract all values greater than 50 from a random array.

S10. Generate an array of 100 random floats between 0 and 1.

Intermediate Tasks

I1. Create a 5x5 matrix where the border is 1 and the inside is 0.

I2. Normalize an array (convert values to range 0–1).

I3. Stack two arrays vertically and horizontally.

I4. Generate a 10x10 matrix of random numbers and compute row-wise and column-wise sums.

I5. Replace all negative values in an array with 0.

I6. Create a matrix and extract its diagonal.

I7. Generate a 15-element array and split it into 3 equal parts.

I8. Multiply two matrices using NumPy's dot product.

I9. Create a boolean mask array and filter the original array.

I10. Compute the Euclidean distance between two 1D arrays.

Hard Tasks

H1. Create a 10x10 random matrix and:

- Sort it row-wise
- Sort it column-wise

H2. Implement manual matrix multiplication using loops (no NumPy dot).

H3. Generate a 1000-element array and compute:

- Variance
- Standard deviation
- Percentiles (25th, 50th, 75th)

H4. Simulate 1000 coin flips (0 = tails, 1 = heads) and compute the probability of heads.

H5. Create a moving average function using NumPy.

H6. Create a 2D Gaussian matrix (heatmap values) using meshgrid.

H7. Implement one-hot encoding using NumPy only.

H8. Perform element-wise division between two arrays, handling divide-by-zero gracefully.

H9. Construct a 3D array (4x4x4) and extract slices in different directions.

H10. Build a function that flattens any N-dimensional NumPy array manually.