EXERCISES

Topic 1: Array Indexing and Slicing

Exercises

1. **Basic**: Create a 1D array with values [5, 10, 15, 20, 25]. Slice the array to get the first three elements.
2. **Basic**: Create a 2D array with values [[1, 2], [3, 4]]. Access the element in the second row, first column.
3. **Basic**: Create a 1D array with values [10, 20, 30, 40, 50]. Use negative indexing to access the last element.
4. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Slice the array to get the second column.
5. **Intermediate**: Create a 2D array with values [[10, 20, 30], [40, 50, 60]]. Slice the array to get the first two rows and first two columns.
6. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get every second row.
7. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get the diagonal elements.
8. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get the subarray consisting of the last two rows and last two columns.
9. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get every second element in the first row.
10. **Basic**: Create a 1D array with values [5, 10, 15, 20, 25]. Use slicing to get the last three elements.
11. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get the first two rows and all columns.
12. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get the second and third columns.
13. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get the subarray consisting of the first and third rows.
14. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get every second element in the second column.

15. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6], [7, 8, 9]]. Use slicing to get the subarray consisting of the first and last elements of each row.

Topic 2: Universal Functions (ufuncs)

Exercises

1. **Basic**: Create a 1D array with values [1, 2, 3, 4, 5]. Compute the sum of all elements using np.sum().
2. **Basic**: Create a 1D array with values [10, 20, 30, 40, 50]. Compute the square root of each element using np.sqrt().
3. **Basic**: Create a 2D array with values [[1, 2], [3, 4]]. Compute the product of all elements using np.prod().
4. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the mean of each row using np.mean().
5. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the standard deviation of each column using np.std().
6. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the maximum value in the array using np.max().
7. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the sum of elements greater than 3 using np.sum() and boolean indexing.
8. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the mean of elements less than 5 using np.mean() and boolean indexing.
9. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the product of elements in the second column using np.prod().
10. **Basic**: Create a 1D array with values [1, 2, 3, 4, 5]. Compute the exponential of each element using np.exp().
11. **Intermediate**: Create a 2D array with values [[1, 2], [3, 4]]. Compute the natural logarithm of each element using np.log().
12. **Intermediate**: Create a 2D array with values [[1, 2], [3, 4]]. Compute the sine of each element using np.sin().
13. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the sum of elements along the rows using np.sum().

14. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the variance of the array using np.var().
15. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the minimum value along the columns using np.min().

Topic 3: Aggregates

Exercises

1. **Basic**: Create a 1D array with values [1, 2, 3, 4, 5]. Compute the sum of all elements using np.sum().
2. **Basic**: Create a 1D array with values [10, 20, 30, 40, 50]. Compute the mean of all elements using np.mean().
3. **Basic**: Create a 2D array with values [[1, 2], [3, 4]]. Compute the minimum value in the array using np.min().
4. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the sum of each column using np.sum().
5. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the mean of each row using np.mean().
6. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the maximum value along the columns using np.max().
7. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the standard deviation of the entire array using np.std().
8. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the variance of each column using np.var().
9. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the product of all elements using np.prod().
10. **Basic**: Create a 1D array with values [1, 2, 3, 4, 5]. Compute the maximum value using np.max().
11. **Intermediate**: Create a 2D array with values [[1, 2], [3, 4]]. Compute the sum of each row using np.sum().
12. **Intermediate**: Create a 2D array with values [[1, 2], [3, 4]]. Compute the mean of the entire array using np.mean().
13. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the sum of elements greater than 3 using np.sum() and boolean indexing.

14. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the mean of elements less than 5 using np.mean() and boolean indexing.
15. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Compute the product of elements in the second column using np.prod().

Topic 4: Broadcasting

Exercises

1. **Basic**: Create a 1D array with values [1, 2, 3]. Add a scalar value of 5 to the array using broadcasting.
2. **Basic**: Create a 2D array with values [[1, 2], [3, 4]]. Multiply the array by a scalar value of 2 using broadcasting.
3. **Basic**: Create a 1D array with values [10, 20, 30]. Subtract a scalar value of 5 from the array using broadcasting.
4. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Add a 1D array [10, 20, 30] to it using broadcasting.
5. **Intermediate**: Create a 2D array with values [[1, 2], [3, 4]]. Multiply it by a 1D array [10, 20] using broadcasting.
6. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Subtract a 1D array [1, 2, 3] from it using broadcasting.
7. **Advanced**: Create a 2D array with values [[1, 2], [3, 4]]. Add a scalar value of 10 and then multiply by a 1D array [2, 3] using broadcasting.
8. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Divide it by a 1D array [1, 2, 3] using broadcasting.
9. **Advanced**: Create a 2D array with values [[1, 2], [3, 4]]. Subtract a scalar value of 5 and then add a 1D array [10, 20] using broadcasting.
10. **Basic**: Create a 1D array with values [5, 10, 15]. Multiply it by a scalar value of 3 using broadcasting.
11. **Intermediate**: Create a 2D array with values [[1, 2], [3, 4]]. Add a scalar value of 10 and then subtract a 1D array [5, 10] using broadcasting.
12. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Multiply it by a scalar value of 2 and then add a 1D array [10, 20, 30] using broadcasting.

13. **Advanced**: Create a 2D array with values [[1, 2], [3, 4]]. Divide it by a scalar value of 2 and then multiply by a 1D array [5, 10] using broadcasting.
14. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Subtract a scalar value of 10 and then divide by a 1D array [2, 3, 4] using broadcasting.
15. **Advanced**: Create a 2D array with values [[1, 2], [3, 4]]. Add a scalar value of 5 and then multiply by a 1D array [2, 3] using broadcasting.

Topic 5: Boolean Arrays and Masks

Exercises

1. **Basic**: Create a 1D array with values [1, 2, 3, 4, 5]. Create a boolean array where elements greater than 3 are True.
2. **Basic**: Create a 2D array with values [[1, 2], [3, 4]]. Create a boolean array where elements less than 3 are True.
3. **Basic**: Create a 1D array with values [10, 20, 30, 40, 50]. Use boolean indexing to select elements greater than 25.
4. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Create a boolean array where elements between 2 and 5 (inclusive) are True.
5. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to select elements less than 4.
6. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to select elements that are even.
7. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to set all elements greater than 3 to 0.
8. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to set all elements less than 5 to -1.
9. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to select elements that are divisible by 2.
10. **Basic**: Create a 1D array with values [5, 10, 15, 20, 25]. Use boolean indexing to select elements less than 20.
11. **Intermediate**: Create a 2D array with values [[1, 2], [3, 4]]. Use boolean indexing to select elements greater than 2.
12. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to select elements that are odd.

13. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to set all elements greater than 4 to 10.
14. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to set all elements less than 3 to -5.
15. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use boolean indexing to select elements that are multiples of 3.

Topic 6: Fancy Indexing

Exercises

1. **Basic**: Create a 1D array with values [10, 20, 30, 40, 50]. Use integer array indexing to select the 2nd and 4th elements.
2. **Basic**: Create a 2D array with values [[1, 2], [3, 4]]. Use integer array indexing to select the first row.
3. **Basic**: Create a 1D array with values [5, 10, 15, 20, 25]. Use integer array indexing to select the 1st, 3rd, and 5th elements.
4. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select the elements at positions (0, 1) and (1, 2).
5. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select the second column.
6. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select the first and third rows.
7. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select the diagonal elements.
8. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select the subarray consisting of the first and last elements of each row.
9. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select every second element in the first row.
10. **Basic**: Create a 1D array with values [10, 20, 30, 40, 50]. Use integer array indexing to select the 3rd and 5th elements.
11. **Intermediate**: Create a 2D array with values [[1, 2], [3, 4]]. Use integer array indexing to select the second row.

12. **Intermediate**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select the first and second columns.
13. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select the subarray consisting of the first and last rows.
14. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select every second element in the second column.
15. **Advanced**: Create a 2D array with values [[1, 2, 3], [4, 5, 6]]. Use integer array indexing to select the subarray consisting of the first and last elements of each column.