

1. Basic Array Manipulation:

Create a 1D array with values ranging from 0 to 9.

```
import numpy as np
#Create 1D array
np_array1d = np.arange(10)
print(np_array1d)

[0  1  2  3  4  5  6  7  8  9]
```

Reshape the array into a 3x3 matrix.

Explore Reshape Function

```
np_array1d = np.arange(9)
#Reshaping into 3x3 matrix
reshaped_array = np_array1d.reshape(3, 3)
print(reshaped_array)

[[0 1 2]
 [3 4 5]
 [6 7 8]]
```

Access the element at the second row, second column.

```
np_array1d = np.arange(9)

reshaped_array = np_array1d.reshape(3, 3)
#Access the elemnet
element = reshaped_array[1, 1]
print(element)

4
```

Perform element-wise addition, subtraction, multiplication, and division on two arrays [1, 2, 3] and [4, 5, 6].

```
array1 = np.array([1, 2, 3])
array2 = np.array([4, 5, 6])

#Perform element-wise addition
add = array1 + array2
print(add)

#Perform element-wise subtraction
sub = array1 - array2
```

```

print(sub)

#Perform element-wise multiplication
mul = array1 * array2
print(mul)

#Perform element-wise division
div = array1 / array2
print(div)

[5 7 9]
[-3 -3 -3]
[ 4 10 18]
[0.25 0.4 0.5 ]

```

Find the sum, mean, and standard deviation of the array [1, 2, 3, 4, 5].

```

array = np.array([1, 2, 3, 4, 5])

#find sum
sum = array.sum()
print(sum)

#find mean
mean = array.mean()
print(mean)

#find standard deviation
std = np.round(array.std(), 2) #round method
print(std)

15
3.0
1.41

```

Reshape the array [1, 2, 3, 4, 5, 6] into a 2x3 array.

Explore Reshape

```

row = np.array([1, 2, 3, 4, 5, 6])

reshaped_row = row.reshape(2, 3)
print(reshaped_row)

[[1 2 3]
 [4 5 6]]

```

2. Indexing and Slicing:

Create a 5x5 array with random integers.

```
random_array = np.random.randint(0, 200, size=(5,5))
print(random_array)

[[ 34  77 114  22  17]
 [ 45 139 188  30 120]
 [121 119  22 167 107]
 [127 115  41  88 114]
 [ 88  96 151 182  32]]
```

Replace all values in the sub-array with a specific value.

```
random_array[1:4, 1:4] = 101
print(random_array)

[[ 34  77 114  22  17]
 [ 45 101 101 101 120]
 [121 101 101 101 107]
 [127 101 101 101 114]
 [ 88  96 151 182  32]]
```

Extract the subarray [[3, 4], [7, 8]] from the array [[1, 2], [3, 4], [5, 6], [7, 8]].

```
array_2d = np.array([[1, 2],
                     [3, 4],
                     [5, 6],
                     [7, 8]])

sub_array = array_2d[[1, 3], :]
print(sub_array)

[[3 4]
 [7 8]]
```

3. Broadcasting:

Create a 2D array of shape (3, 3) with values from 0 to 2.

```
values = np.array([0, 1, 2])

arr_2d = np.tile(values, (3, 1))
print(arr_2d)
```

```
[[0 1 2]
 [0 1 2]
 [0 1 2]]
```

Add a 1D array of shape (3,) to each row of the 2D array using broadcasting.

```
arr_1d = np.array([1, 2, 3])
arr_2d + arr_1d
array([[1, 3, 5],
       [1, 3, 5],
       [1, 3, 5]])
```

4. Concatenation and Splitting

1. Create two 2D arrays of shape (3, 3) with random integers.
2. Concatenate them horizontally and vertically.
3. Split the concatenated arrays back into the original arrays.

```
arr1_2d = np.random.randint(0, 50, size=(3, 3))
arr2_2d = np.random.randint(0, 50, size=(3, 3))
print('Original array 1')
print(arr1_2d)
print('Original array 2')
print(arr2_2d)

print('Horizontal Concatenated Array')
horizontal_concat_array = np.hstack((arr1_2d, arr2_2d))
print(horizontal_concat_array)

print('Vertical Concatenated Array')
vertical_concat_array = np.vstack((arr1_2d, arr2_2d))
print(vertical_concat_array)

print('Split Horizontal Array')
original_arr1_h, original_arr2_h = np.hsplit(horizontal_concat_array, 2)
print(original_arr1_h)
print(original_arr2_h)

print('Split Vertical Array')
original_arr1_v, original_arr2_v = np.vsplit(vertical_concat_array, 2)
print(original_arr1_v)
print(original_arr2_v)

Original array 1
[[42  2 12]
```

```

[21 19  8]
[17 22 43]]
Original array 2
[[ 2 19 41]
 [46 12 32]
 [37 23  6]]
Horizontal Concatenated Array
[[42  2 12  2 19 41]
 [21 19  8 46 12 32]
 [17 22 43 37 23  6]]
Vertical Concatenated Array
[[42  2 12]
 [21 19  8]
 [17 22 43]
 [ 2 19 41]
 [46 12 32]
 [37 23  6]]
Split Horizontal Array
[[42  2 12]
 [21 19  8]
 [17 22 43]]
[[ 2 19 41]
 [46 12 32]
 [37 23  6]]
Split Vertical Array
[[42  2 12]
 [21 19  8]
 [17 22 43]]
[[ 2 19 41]
 [46 12 32]
 [37 23  6]]

```

Bonus Question

Find mode of this array = [1,2,1,2,3,4,5,6,8,9,1,2,2,22,21,1,1,1,1,2,5]

Hint: Use count and also loop

```

rand_array = [1,2,1,2,3,4,5,6,8,9,1,2,2,22,21,1,1,1,1,2,5]

#occurrences of each element
count = {}
for value in rand_array:
    if value in count:
        count[value] += 1
    else:
        count[value] = 1

```

```
#Mode of the given array
max_count = 0
mode = None

for key,value in count.items():
    if value > max_count:
        max_count = value
        mode = key

print(f'The Mode of Given array is {mode}')
```

The Mode of Given array is 1

Notes

1. All the Questions carry Equal Points (2 points)
2. Bonus Question carry 5 points
3. Review lecture for any query

Good Luck!