EXERCISE - 4. Indexing and Slicing of NumPy Array

********** **********

1) Slicing 1-D NumPy arrays

1.1)Basic Slicing

```
 import numpy as np

# Create a 1-D array
arr = np.array([10, 20, 30, 40, 50])

# Slice elements from index 1 to 3 (excluding 4)
sliced_arr = arr[1:4]
print(sliced_arr)
 # Output: [20 30 40]
```

1.2) Slicing with Step

```
 import numpy as np

# Create a 1-D array
arr = np.array([10, 20, 30, 40, 50, 60, 70])

# Slice elements from index 0 to 6 with a step of 2
sliced_arr = arr[0:7:2]
print(sliced_arr)
# Output: [10 30 50 70]
```

1.3) Slicing from the End

```
  import numpy as np

# Create a 1-D array
arr = np.array([10, 20, 30, 40, 50])

# Slice the last three elements
sliced_arr = arr[-3:]
print(sliced_arr)
# Output: [30 40 50]
```

****

2) Slicing 2-D NumPy arrays

2.1) Basic Slicing

```
 import numpy as np

# Create a 2-D array
arr = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])

# Slice elements from rows 0 to 1 and columns 1 to 2
sliced_arr = arr[0:2, 1:3]
print(sliced_arr)
# Output: [[2 3]
#          [6 7]]
```

2.2) Slicing with Step

```
  import numpy as np

# Create a 2-D array
arr = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])

# Slice every other element from rows and columns
sliced_arr = arr[::2, ::2]
```

```
print(sliced_arr)
# Output: [[ 1  3]
#          [ 9 11]]

--------------------------------

2.3) Slicing Rows or Columns
  import numpy as np

# Create a 2-D array
arr = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])

# Slice the second row
sliced_arr = arr[1, :]
print(sliced_arr)
# Output: [5 6 7 8]

# Slice the second column
sliced_arr = arr[:, 1]
print(sliced_arr)
# Output: [ 2  6 10]

**************************************************

    3)  Slicing 3-D NumPy arrays

    3.1) Basic Slicing

import numpy as np

# Create a 3-D array
arr = np.array([[[1, 2, 3], [4, 5, 6]],
                [[7, 8, 9], [10, 11, 12]],
                [[13, 14, 15], [16, 17, 18]]])

# Slice elements from 0 to 2 in the first dimension,
# 0 to 1 in the second dimension, and all elements in the third dimension
sliced_arr = arr[0:2, 0:1, :]
print(sliced_arr)
# Output: [[[1 2 3]]
#          [[7 8 9]]]

--------------------------------

3.2) Slicing with Step

  import numpy as np

# Create a 3-D array
arr = np.array([[[1, 2, 3], [4, 5, 6]],
                [[7, 8, 9], [10, 11, 12]],
                [[13, 14, 15], [16, 17, 18]]])

# Slice elements with a step of 2 in the first dimension,
# and all elements in the other dimensions
sliced_arr = arr[::2, :, :]
print(sliced_arr)
# Output: [[[ 1  2  3]
#           [ 4  5  6]]
#          [[13 14 15]
#           [16 17 18]]]

---------------------------------
```

3.3) Slicing Specific Rows or Columns

```
  import numpy as np

# Create a 3-D array
arr = np.array([[[1, 2, 3], [4, 5, 6]],
                [[7, 8, 9], [10, 11, 12]],
                [[13, 14, 15], [16, 17, 18]]])

# Slice the second row from each 2-D array
sliced_arr = arr[:, 1, :]
print(sliced_arr)
# Output: [[ 4  5  6]
#          [10 11 12]
#          [16 17 18]]

# Slice the second column from each 2-D array
sliced_arr = arr[:, :, 1]
print(sliced_arr)
# Output: [[ 2  5]
#          [ 8 11]
#          [14 17]]
```

****

4) Negative slicing of NumPy arrays

4.1) Negative Slicing in 1-D Arrays

```
import numpy as np

# Create a 1-D array
arr = np.array([10, 20, 30, 40, 50])

# Slice the last three elements
sliced_arr = arr[-3:]
print(sliced_arr)
# Output: [30 40 50]

# Slice the array excluding the last element
sliced_arr = arr[:-1]
print(sliced_arr)
# Output: [10 20 30 40]
```

4.2) Negative Slicing in 2-D Arrays

```
import numpy as np

# Create a 2-D array
arr = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])

# Slice the last two rows and last two columns
sliced_arr = arr[-2:, -2:]
print(sliced_arr)
# Output: [[ 7  8]
#          [11 12]]

# Slice the array excluding the last row
sliced_arr = arr[:-1, :]
print(sliced_arr)
# Output: [[1 2 3 4]
#          [5 6 7 8]]
```

4.3) Negative Slicing in 3-D Arrays

```
 import numpy as np

# Create a 3-D array
arr = np.array([[[1, 2, 3], [4, 5, 6]],
                [[7, 8, 9], [10, 11, 12]],
                [[13, 14, 15], [16, 17, 18]]])

# Slice the last two 2-D arrays and the last two columns
sliced_arr = arr[-2:, :, -2:]
print(sliced_arr)
# Output: [[[ 2  3]
#           [ 5  6]]
#          [[14 15]
#           [17 18]]]

# Slice the array excluding the last 2-D array
sliced_arr = arr[:-1, :, :]
print(sliced_arr)
# Output: [[[ 1  2  3]
#           [ 4  5  6]]
#          [[ 7  8  9]
#           [10 11 12]]]
```

****