NumPy Array Indexing

Negative Indexing

```
import numpy as np
arr = np.array([[1,2,3,4,5],[6,7,8,9,10]])
print('Last element from 2nd dim: ' arr[0,-1])
```

Output

10

NumPy Array Slicing

```
[start : end]
[start : end : step]
```

If we don't pass start its considered 0
" " " " " step " " 1

```
import numpy as np
arr = np.array([1,2,3,4,5,6,7])
print(arr[1:5])
```

Output

[2, 3, 4, 5]

5 will not be there because we know the rule of slicing which is the last value will not print out.

2) import numpy as np
arr = np.array([1,2,3,4,5,6,7])
print(arr[4:])

Output
[5,6,7]

3) import numpy as np
arr = np.array([1,2,3,4,5,6,7])
print(arr[:4])

Output
[1,2,3,4]

Negative slicing

import numpy as np
arr = np.array([1,2,3,4,5,6,7])
print(arr[-3:-1])

Output
[5,6]

STEP

Return every other element from index 1 to index 5:

1) import numpy as np
arr = np.array([1,2,3,4,5,6,7])
print(arr[1:5:2])

Output
[2,4]

2) import numpy as np
arr = np.array([1,2,3,4,5,6,7])
print(arr[::2])

Output
[1,3,5,7]

Slicing 2-D Arrays

1) import numpy as np
arr = np.array([[1,2,3,4,5],[6,7,8,9,10]])
print(arr[1,1:4])

Output

[7,8,9]

2) import numpy as np
arr = np.array([[1,2,3,4,5],[6,7,8,9,10]])
print(arr[0:2,2]

Output

[3,8]

3) import numpy as np
arr = np.array([[1,2,3,4,5],[6,7,8,9,10]])
print(arr[0:2,1:4])