Indexing in NumPy

This lesson will help you learn indexing in NumPy.

Indexing means to refer to any value in an array. Each item in a numpy array is stored at a specific index. To access value at a specific index write:

```
Z=np.arrange(9)
Z[0] #get the value at index 0
```

Get the First Value

To get the first value of a matrix, write: `Z(0,0)`.

```
   | 0 | 1 | 2 | → | 0 | (scalar)
Z  | 3 | 4 | 5 |
   | 6 | 7 | 8 |
```

```python
import numpy as np
Z = np.arange(9).reshape(3,3)
print(Z[0,0])
```

RUN SAVE RESET

Output 0.739s

```
0
```

Get the Last Value

To get the last value of a matrix, write: `Z[-1,-1]`.

```
   | 0 | 1 | 2 |
Z  | 3 | 4 | 5 |
   | 6 | 7 | 8 | → | 8 | (scalar)
```

```python
import numpy as np
Z = np.arange(9).reshape(3,3)
print(Z[-1,-1])
```

RUN SAVE RESET

Output 0.701s

```
8
```

Get a row from a Grid

To get a row from a grid, write: `Z[row_index]`.

To get the first row from a grid, write: `Z[1]`.

```
   | 0 | 1 | 2 |
Z  | 3 | 4 | 5 | → | 3 | 4 | 5 |
   | 6 | 7 | 8 |      (view)
```

```python
import numpy as np
Z = np.arange(9).reshape(3,3)
print(Z[1])
```

RUN SAVE RESET

Output 0.756s

```
[3 4 5]
```

Get a Column from a Grid

To get the column from a grid, use `Z[:,column_index]`

To get the second column from a grid, use `Z[:,2]`

```
   | 0 | 1 | 2 |     | 2 |
Z  | 3 | 4 | 5 | →   | 5 | (view)
   | 6 | 7 | 8 |     | 8 |
```

```python
import numpy as np
Z = np.arange(9).reshape(3,3)
print(Z[:,2])


```

RUN SAVE RESET

Output 0.645s

```
[2 5 8]
```

Get a Mini-grid from a Grid

To get a subset of a grid, write: `Z[row_index:,column_index:]`.

To get a subset of a grid containing the first row onwards up to the size and first column onwards up to the size, write: `Z[1:,1:]`.

```
   | 0 | 1 | 2 |    (view)
Z  | 3 | 4 | 5 |   | 4 | 5 |
   | 6 | 7 | 8 | → | 7 | 8 |
```

```python
import numpy as np
Z = np.arange(9).reshape(3,3)
print(Z[1:,1:])
```

RUN SAVE RESET

Output 0.649s

```
[[4 5]
 [7 8]]
```

Arrange Values from a Grid in a Mini-grid

To get the values from corners of a grid and arrange them in a grid format write: `Z[::row_size-1,::column_size-1]`

To get the values at index (0,0),(0,2),(2,0),(2,2) and arrange them in a grid format write: `(Z[::2,::2])`

```
Z
| 0 | 1 | 2 |   | 0 | 2 |
| 3 | 4 | 5 | → | 6 | 8 |
| 6 | 7 | 8 |   (view)
```

```python
import numpy as np
Z = np.arange(9).reshape(3,3)
print(Z[::2,::2])
```

RUN SAVE RESET

Output 0.639s

```
[[0 2]
 [6 8]]
```

Get Specific Indices from a Grid

To get specific indices values such as (0,0) and (0,2) write: `(Z[[0,1],[0,2]])`.

```
   | 0 | 1 | 2 |
Z  | 3 | 4 | 5 | → | 0 | 5 |
   | 6 | 7 | 8 |     (copy)
```

```python
import numpy as np
Z = np.arange(9).reshape(3,3)
print(Z[[0,1],[0,2]])
```

RUN SAVE RESET

Output 0.648s

```
[0 5]
```

Solve this Quiz!

Q Given an np array `Z` .How would you get the following values from Z?

```
   | 1 | 1 | 2 |
Z  | 3 | 4 | 9 | → | 1 | 9 |
   | 6 | 7 | 8 |     (copy)
```

A) `print(Z[[0,1],[0,2]])`

B) `print(Z[[0,1],[0,1]])`

COMPLETED 0%

1 of 1

Now that you have learned about indexing in NumPy, let's move on to the next lesson "Broadcasting in NumPy".