NAME: MUHAMMAD HUSSAM (2303.KHI.DEG.020)

PAIRING WITH : MAVIA ALAM KHAN (2303.KHI.DEG.017)

&

AQSA TAUHEED(2303.KHI.DEG.011)

ASSIGNMENT N0 2.2

Build a 6x4 matrix of random numbers.
Using slicing, replace rows 5-6 of the matrix so that the 5th row becomes a sum of the 1st and the 3rd row, and the 6th row becomes a sum of the 2nd and the 4th one.

Solution:

Step1:


```
import numpy as np
matrix = np.random.randint(0, 101, size=(6, 4))
print(matrix)
```

```
[[ 78  40  41  11]
 [ 89  88   0  21]
 [ 68 100  90  70]
 [ 70  46  41  46]
 [ 72  33  55  23]
 [  2  30  11  55]]
```

To build a 6 x 4 matrix with random integer values ranging from 0 to 100 (inclusive), we can use the **np.random.randint()** function from the NumPy library. After importing NumPy as **np**, we can generate the random matrix by calling **np.random.randint(0, 101, size=(6, 4))**, where the first argument specifies the inclusive lower bound of the range of integers, the second argument specifies the exclusive upper bound of the range of integers, and **size=(6, 4)** specifies the dimensions of the matrix.After that we printed matrix which we build .

Step2 :

```
[5]: matrix[4:5] = matrix[0] + matrix[2]

[6]: print(matrix[4:5])

[[146 140 131  81]]
```

To replace the fifth row of the matrix as the sum of the first and third rows, we can use NumPy's slicing functionality. To select the fifth row, we can use Matrix[4:5], where the index 4 refers to the fifth row since indexing starts from 0 and the range is exclusive of the ending index. To compute the sum of the first and third rows, we can simply add Matrix[0] + Matrix[2]. Therefore, to replace the fifth row with the sum of the first and third rows, we use Matrix[4:5] = Matrix[0] + Matrix[2]

So 5^{th} row = [[146,140,131,81]]

Step3:

```
[8]: matrix[5:6]= matrix[1] + matrix[3]
```

```
[9]: print(matrix[5:6])
```

```
[[159 134  41  67]]
```

To replace the sixth row of the matrix as the sum of the second and fourth rows, we can use NumPy's slicing functionality. To select the sixth row, we can use Matrix[5:6], where the index 5 refers to the sixth row since indexing starts from 0 and the range is exclusive of the ending index. To compute the sum of the second and fourth rows, we can simply add Matrix[1] + Matrix[3]. Therefore, to replace the sixth row with the sum of the second and fourth rows, we use Matrix[5:6] = Matrix[1] + Matrix[3] , so 6th row = [[159,134,41,67]].

<u>Step4:</u>

```
[[159 134  41  67]]
```

```
[17]: print(matrix)
```

```
[[ 78  40  41  11]
 [ 89  88   0  21]
 [ 68 100  90  70]
 [ 70  46  41  46]
 [146 140 131  81]
 [159 134  41  67]]
```

Now we printed matrix , as you can above the 5th and 6th rows are changed from the original matrix , so the ,modified matrix would be like as follow:

Modified matix :

```
[[ 78  40  41  11]
 [ 89  88   0  21]
 [ 68 100  90  70]
 [ 70  46  41  46]
 [146 140 131  81]
 [159 134  41  67]]
```

Whole Code:

nt2.2.ipynb

Code

Python 3 (ipykernel)

```python
import numpy as np
matrix = np.random.randint(0, 101, size=(6, 4))
print(matrix)
```

```
[[ 78  40  41  11]
 [ 89  88   0  21]
 [ 68 100  90  70]
 [ 70  46  41  46]
 [ 72  33  55  23]
 [  2  30  11  55]]
```

```python
matrix[4:5] = matrix[0] + matrix[2]
```

```python
print(matrix[4:5])
```

```
[[146 140 131  81]]
```

```python
print(matrix)
```

```
[[ 78  40  41  11]
 [ 89  88   0  21]
 [ 68 100  90  70]
 [ 70  46  41  46]
 [146 140 131  81]
 [  2  30  11  55]]
```

```python
matrix[5:6]= matrix[1] + matrix[3]
```

```python
print(matrix[5:6])
```

```
[[159 134  41  67]]
```

```python
print(matrix)
```

```
[[ 78  40  41  11]
 [ 89  88   0  21]
 [ 68 100  90  70]
 [ 70  46  41  46]
 [146 140 131  81]
 [159 134  41  67]]
```

Original matrix =

```
 [[ 78  40  41  11]
 [ 89  88   0  21]
 [ 68 100  90  70]
 [ 70  46  41  46]
 [ 72  33  55  23]
 [  2  30  11  55]]
```

Modified matrix=

```
[[ 78  40  41  11]
 [ 89  88   0  21]
 [ 68 100  90  70]
 [ 70  46  41  46]
 [146 140 131  81]
 [159 134  41  67]]
```