

# Summer 2023: CS5710\_Machine Learning

## In-Class Programming Assignment-1

JAIOSN VICTOR JOSEPH FRANCIS XAVIER- 700743397

```
Machine Learning Programming Assignment 1.ipynb ☆
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

[185] 1 #JAIOSN VICTOR JOSEPH FRANCIS XAVIER- 700743397
      2 #Summer 2023: CS5710_Machine Learning
      3 #In-Class Programming Assignment-1

[186] 1 #1. Numpy:
      2 #a)Using NumPy create random vector of size 15 having only Integers in the range 1-20.
      3 import numpy as np
      4 matrix_values = np.random.randint(1,20, size = 15)
      5 print(matrix_values)

[15  6 14  6  9  7 18  2  6  7  8 14 10 16 14]

[187] 1 # 1. Reshape the array to 3 by 5
      2 Reshape array= matrix_values.reshape(3,5)
      3 print(Reshape_array)

[[15  6 14  6  9]
 [ 7 18  2  6  7]
 [ 8 14 10 16 14]]
```

```
Machine Learning Programming Assignment 1.ipynb ☆
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

[188] 1 # 2. Print array shape.
      2 print("The Array is:\n",Reshape_array)
      3 print("\n The Array shape is:",Reshape_array.shape)

The Array is:
[[15  6 14  6  9]
 [ 7 18  2  6  7]
 [ 8 14 10 16 14]]

The Array shape is: (3, 5)

[189] 1 # 3. Replace the max in each row by 0
      2 Matrix_1 = np.where(Reshape_array == [[i] for i in np.amax(Reshape_array, axis = 1)], 0, Reshape_array)
      3 print(Matrix_1)

[[ 0  6 14  6  9]
 [ 7  0  2  6  7]
 [ 8 14 10  0 14]]
```

## Summer 2023: CS5710\_Machine Learning

```
Machine Learning Programming Assignment 1.ipynb ☆
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

[190] 1 # Create a 2-dimensional array of size 4 x 3 (composed of 4-byte integer elements), also print the shape, type and data type of the array.
      2 import numpy as np
      3
      4 # create a 2-dimensional array of size 4x3
      5 Array_1 = np.array([[5, 10, 15], [20, 25, 30], [35, 40, 45], [50, 55, 60]], dtype=np.int32)
      6 # print the array shape, array type & array data type:
      7 print("The Array shape:", Array_1.shape, "\nThe Array type:", type(Array_1), "\nThe Array data type:", Array_1.dtype)

The Array shape: (4, 3)
The Array type: <class 'numpy.ndarray'>
The Array data type: int32

[191] 1 #b)Write a program to compute the eigenvalues and right eigenvectors
      2 import numpy as np
      3 #Define the square array:
      4 A = np.array([[3, -2], [1, 0]])
      5
      6 #Compute the eigenvalues and right eigenvectors:
      7 Eigenvalues, Eigenvectors = np.linalg.eig(A)
      8
      9 #Print the eigenvalues and right eigenvectors:
      10 print("Eigenvalues:\n", Eigenvalues, "\nRight Eigenvectors:\n", Eigenvectors)
```

```
Machine Learning Programming Assignment 1.ipynb ☆
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Eigenvalues:
[2. 1.]
Right Eigenvectors:
[[0.89442719 0.70710678]
 [0.4472136  0.70710678]]

[192] 1 #c)Compute the sum of the diagonal element of a given array
      2 import numpy as np
      3 #Define the array:
      4 A = np.array([[0, 1, 2], [3, 4, 5]])
      5
      6 #Compute the sum of the diagonal elements:
      7 diagonal_sum = np.trace(A)
      8
      9 #Print the sum of the diagonal elements:
      10 print("Sum of diagonal elements:", diagonal_sum)

Sum of diagonal elements: 4
```

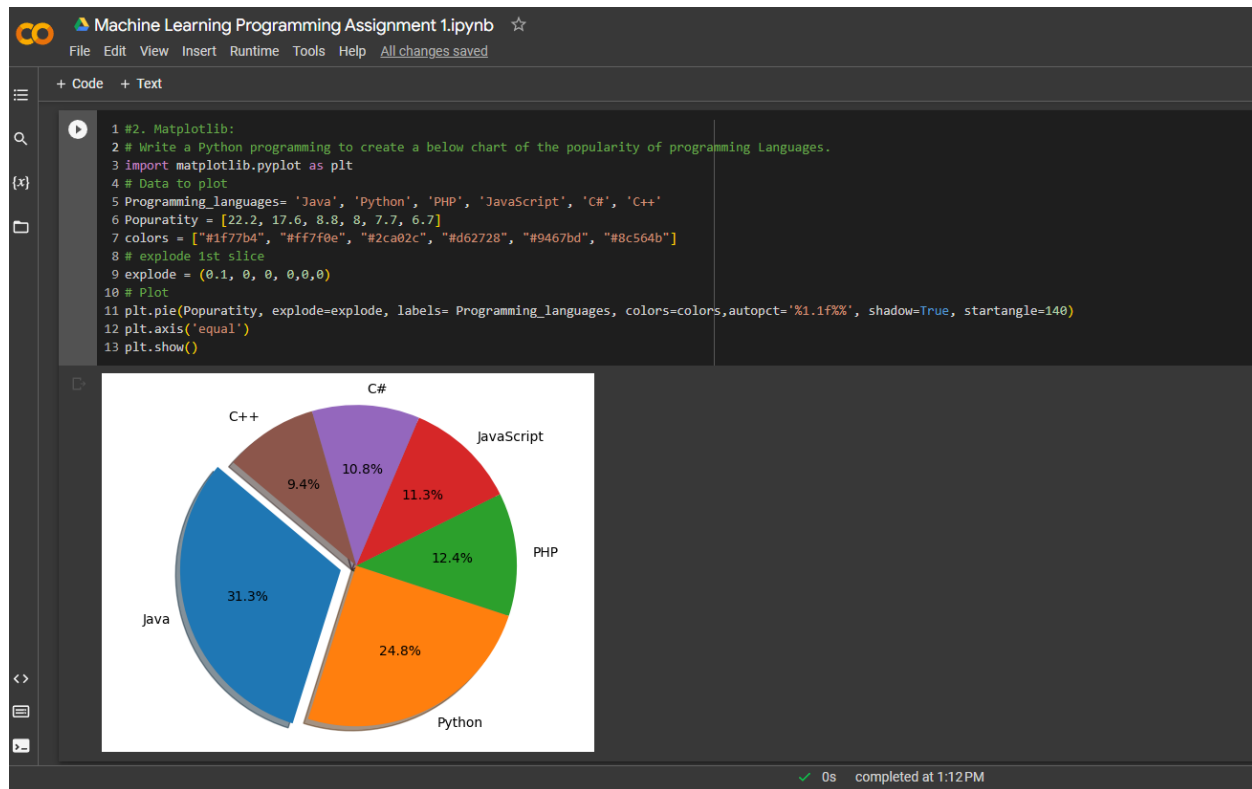
```
Machine Learning Programming Assignment 1.ipynb ☆
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

1 #d)Write a NumPy program to create a new shape to an array without changing its data.
2 import numpy as np
3
4 #Define the original array
5 Arr = np.array([[1, 2], [3, 4], [5, 6]])
6
7 #Reshape to 3x2
8 Arr_3x2 = Arr.reshape(3, 2)
9 # reshape to 2x3
10 Arr_2x3 = Arr.reshape(2, 3)
11
12 print("Before Reshape the matrix is:\n", Arr, "\nAfter Reshaped to 3x2:\n", Arr_3x2, "\nAfter Reshaped to 2x3:\n", Arr_2x3)
13

Before Reshape the matrix is:
[[1 2]
 [3 4]
 [5 6]]
After Reshaped to 3x2:
[[1 2]
 [3 4]
 [5 6]]
After Reshaped to 2x3:
[[1 2 3]
 [4 5 6]]
```

## Summer 2023: CS5710\_Machine Learning



Googlecolab link:

[https://drive.google.com/file/d/1PqmO1ZLSg8dLBZSm2p\\_T1HLq5qPkRmrV/view?usp=sharing](https://drive.google.com/file/d/1PqmO1ZLSg8dLBZSm2p_T1HLq5qPkRmrV/view?usp=sharing)