SAI NIKITH REDDY YARRAM (700742917)

MACHINE LEARNING

CRN: 30521

ASSIGNMENT 1

JUNE 20th 2023

GITHUB LINK: https://github.com/nikithreddy30/ML.Assginment1

VIDEO LINK: https://drive.google.com/file/d/1bTfPaydGDy7F0I39N-Q_nKlivIL5ZXzh/view?usp=sharing

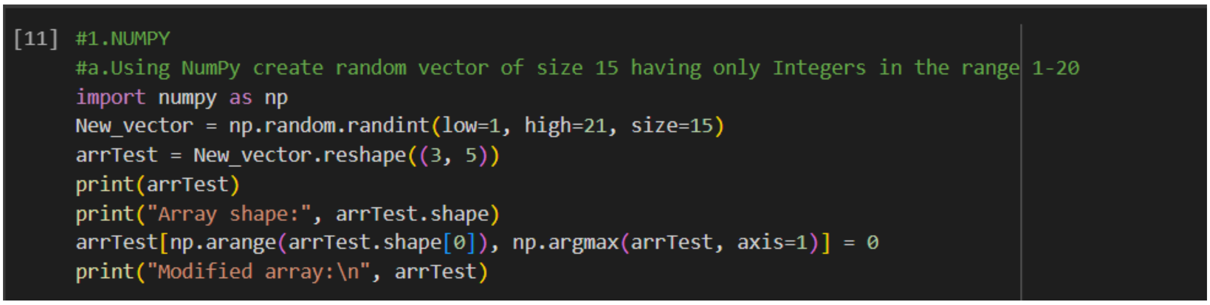
1. NUMPY:

a. Using NumPy creates a random vector of size 15 having only Integers in the range 1-20.

1. Reshape the array to 3 by 5

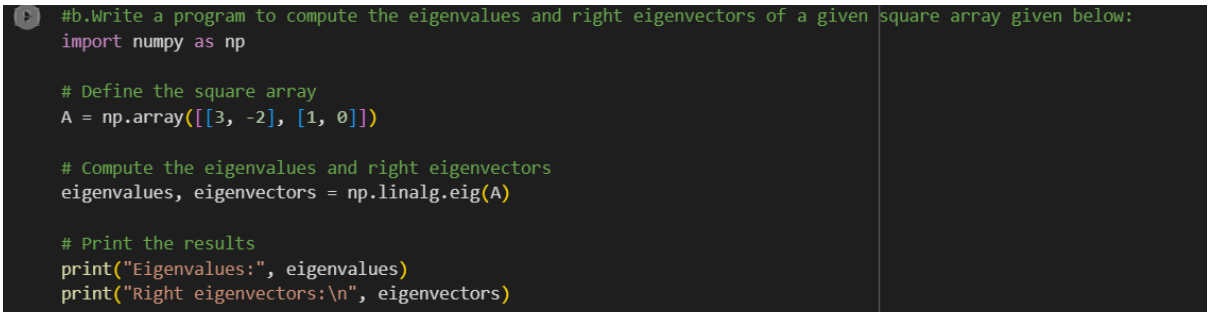
2. Print array shape.

3. Replace the max in each row by 0 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.



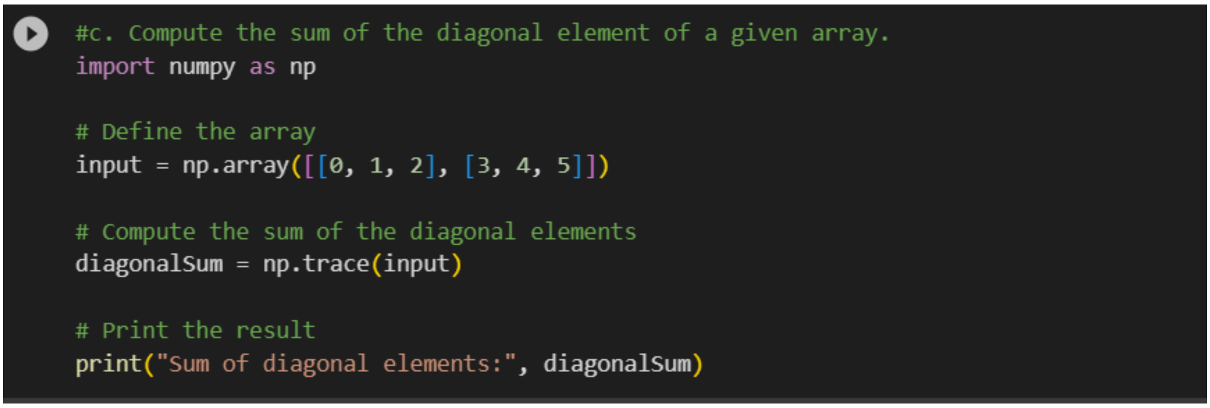
```
[11] #1.NUMPY
     #a.Using NumPy create random vector of size 15 having only Integers in the range 1-20
     import numpy as np
     New_vector = np.random.randint(low=1, high=21, size=15)
     arrTest = New_vector.reshape((3, 5))
     print(arrTest)
     print("Array shape:", arrTest.shape)
     arrTest[np.arange(arrTest.shape[0]), np.argmax(arrTest, axis=1)] = 0
     print("Modified array:\n", arrTest)
```

b. Write a program to compute the eigenvalues and right eigenvectors of a given square array given below: [[3 -2] [1 0]]



```
#b.Write a program to compute the eigenvalues and right eigenvectors of a given square array given below:
import numpy as np

# Define the square array
A = np.array([[3, -2], [1, 0]])

# Compute the eigenvalues and right eigenvectors
eigenvalues, eigenvectors = np.linalg.eig(A)

# Print the results
print("Eigenvalues:", eigenvalues)
print("Right eigenvectors:\n", eigenvectors)
```

c. Compute the sum of the diagonal element of a given array. [[0 1 2] [3 4 5]]

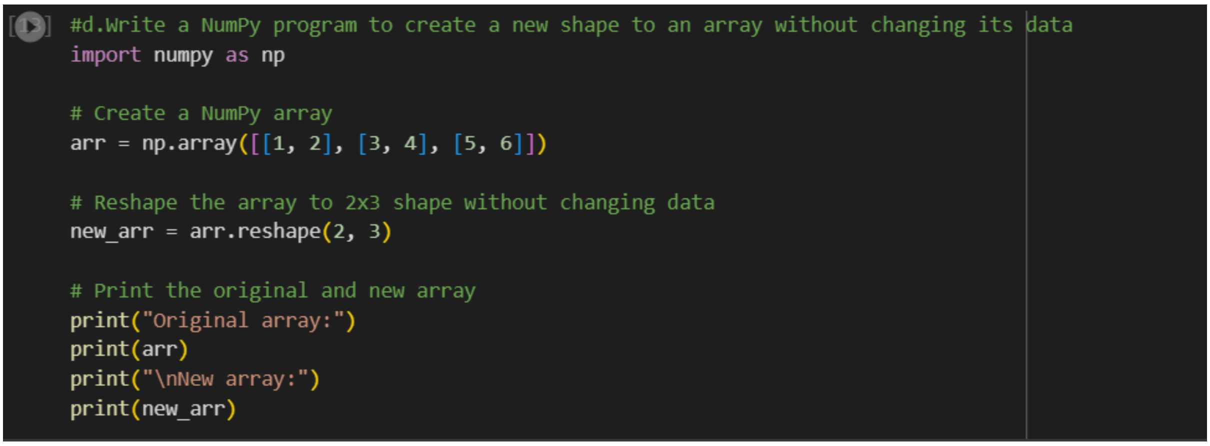


```
#c. Compute the sum of the diagonal element of a given array.
import numpy as np

# Define the array
input = np.array([[0, 1, 2], [3, 4, 5]])

# Compute the sum of the diagonal elements
diagonalSum = np.trace(input)

# Print the result
print("Sum of diagonal elements:", diagonalSum)
```

d. Write a NumPy program to create a new shape to an array without changing its data.

Reshape 3x2: [[1 2] [3 4] [5 6]]

Reshape 2x3: [[1 2 3] [4 5 6]]



```
#d.Write a NumPy program to create a new shape to an array without changing its data
import numpy as np

# Create a NumPy array
arr = np.array([[1, 2], [3, 4], [5, 6]])

# Reshape the array to 2x3 shape without changing data
new_arr = arr.reshape(2, 3)

# Print the original and new array
print("Original array:")
print(arr)
print("\nNew array:")
print(new_arr)
```

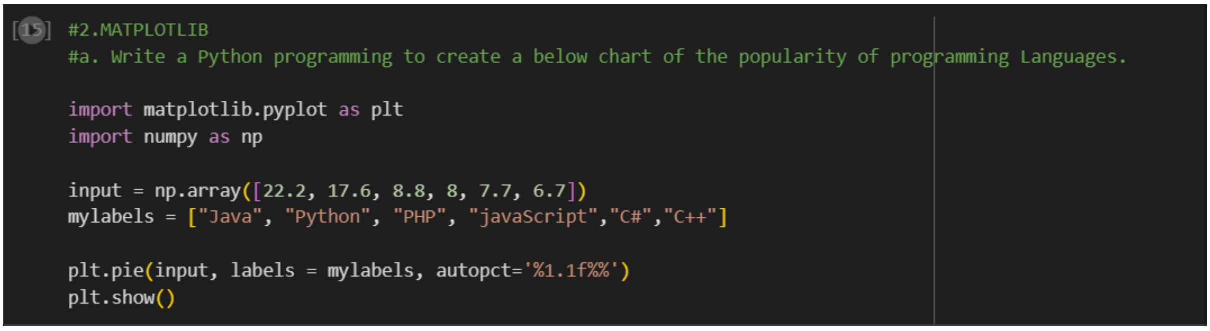
2. MATPLOTLIB

a. Write a Python programming to create a below chart of the popularity of programming Languages.

b. Sample data:

Programming languages: Java, Python, PHP, JavaScript, C#, C++

Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7



[15]

```
#2.MATPLOTLIB
#a. Write a Python programming to create a below chart of the popularity of programming Languages.

import matplotlib.pyplot as plt
import numpy as np

input = np.array([22.2, 17.6, 8.8, 8, 7.7, 6.7])
mylabels = ["Java", "Python", "PHP", "javaScript","C#","C++"]

plt.pie(input, labels = mylabels, autopct='%1.1f%%')
plt.show()
```