**Assignment-3**

**Name: Vijay Kumar**

**ID:700737910**

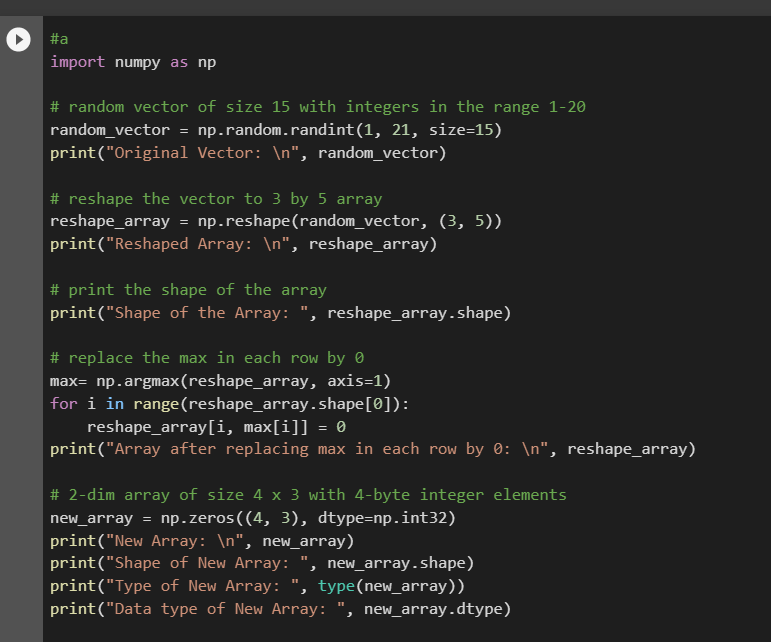
Git hub:

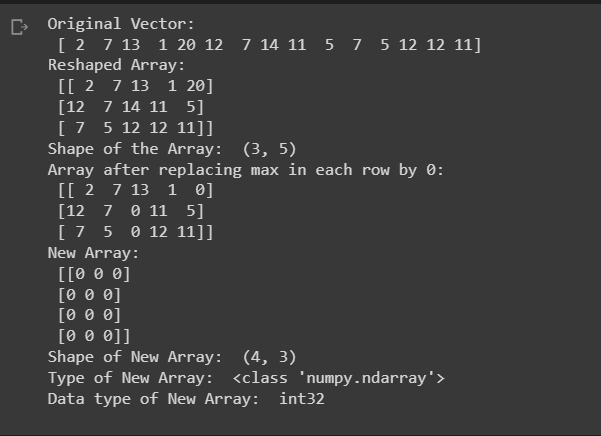
https://github.com/Vijaykamsani/ML\_Assignment\_-70037910/blob/main/Assignment\_3.ipynb

Video link:

https://drive.google.com/file/d/1GDAbgOvYuQa3z8f9JzG2P1sSpi6Y5sx5/view?usp=share\_link

**a>**

****

****

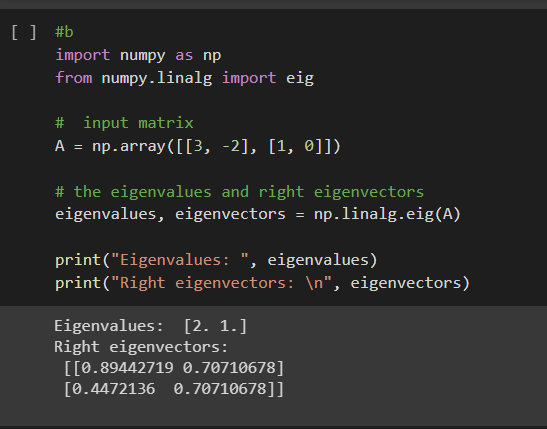
**Comment:**

First, we import the NumPy library as np. Then, we use the np.random.randint() function to create a random vector for a random\_vector of size 15 with integers in the range 1-20.Next, we use np.reshape() to reshape the random\_vector into a 3x5 array reshape\_array.

After that, we use np.argmax() with axis=1 to get the column of the maximum elements in each row of reshape\_array and store them in max. We then use a for loop to iterate over the rows of reshape\_array and set the maximum element in each row to 0 by using the row index and the corresponding value in max.

Finally, we create a new 4x3 array new\_array of integers with default value 0 using np.zeros() with a dtype of np.int32. We then print the original random\_vector, the reshape\_array, and the new\_array, along with their respective shapes, types, and data types.

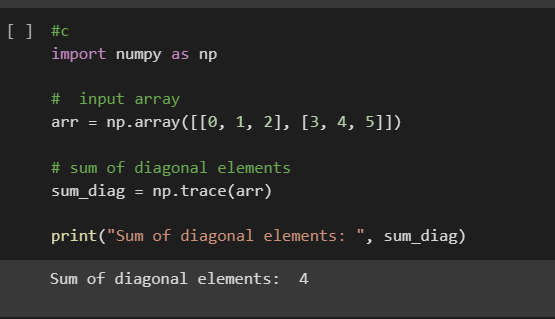
**b>**

****

**Comment:**

In this code, we first create the input matrix A using NumPy array. Then, we compute the eigenvalues and right eigenvectors of the matrix using np.linalg.eig() function. The function returns two values, the first value is an array containing the eigenvalues of the matrix, and the second value is a matrix containing the right eigenvectors of the matrix. Finally, we print the eigenvalues and right eigenvectors using the print function.

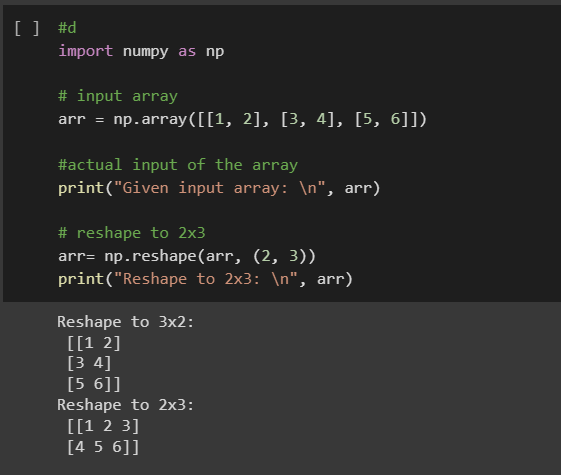
C>



**Comment:**

In this code, we first create the input array arr using NumPy array. Then, we compute the sum of diagonal elements using the np.trace() function. The function returns the sum of diagonal elements of the input array. Finally, we print the result using the print() function.

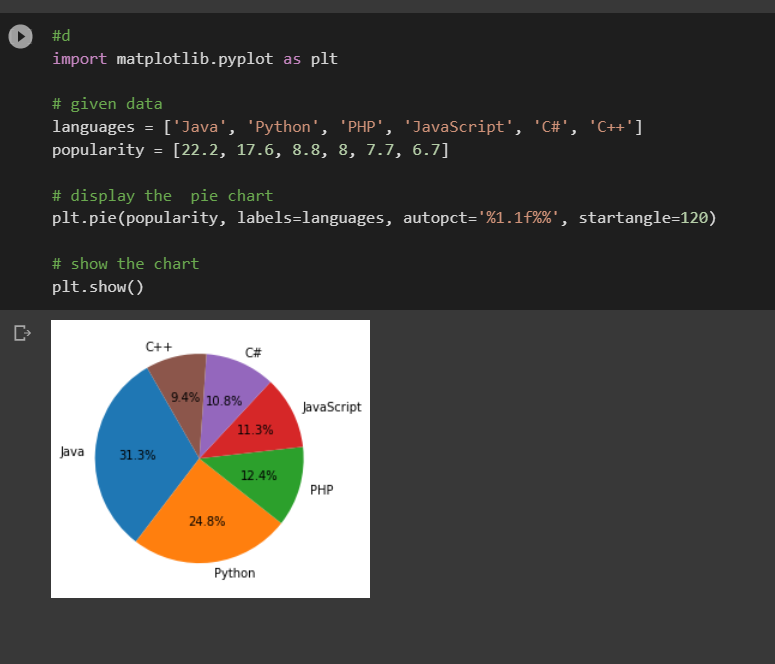
d>



**Comment:**

In this code, we first create the input array arr using NumPy array. Then, we reshape the array to a new shape without changing its data using the np.reshape() function. We pass the new shape as a tuple to the function. Finally, we print the new reshaped arrays using the print() function.

2>



**Comment:**

In this code, we first define the sample data languages and popularity. Then, we create a pie chart using the plt.pie() function, passing the popularity as the data to be charted and languages as the labels for each slice. We use the autopct parameter to display the percentage values for each slice and the startangle parameter to rotate the starting angle of the chart. Finally, we show the chart using the plt.show() function