Ram Mohan Rao Yeduri

700740253

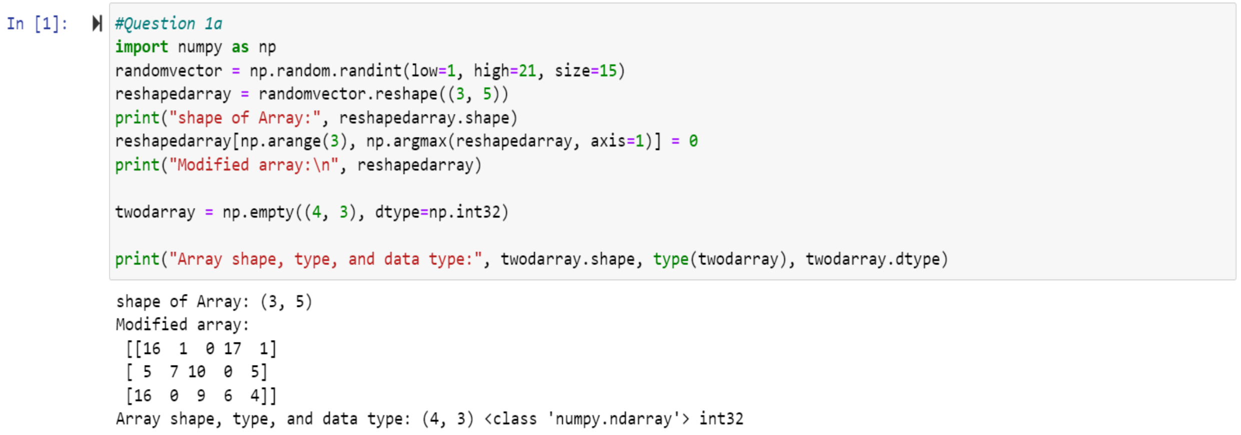
Machine Learning – Assignment1

GitHub: https://github.com/rxy02530/ASSIGNMENT_1.git

Video link: https://drive.google.com/file/d/1dOEZz6g5QS4Ru52meLnR3aUcEoS6isEy/view?usp=drive_link

Task 1: Using Numpy

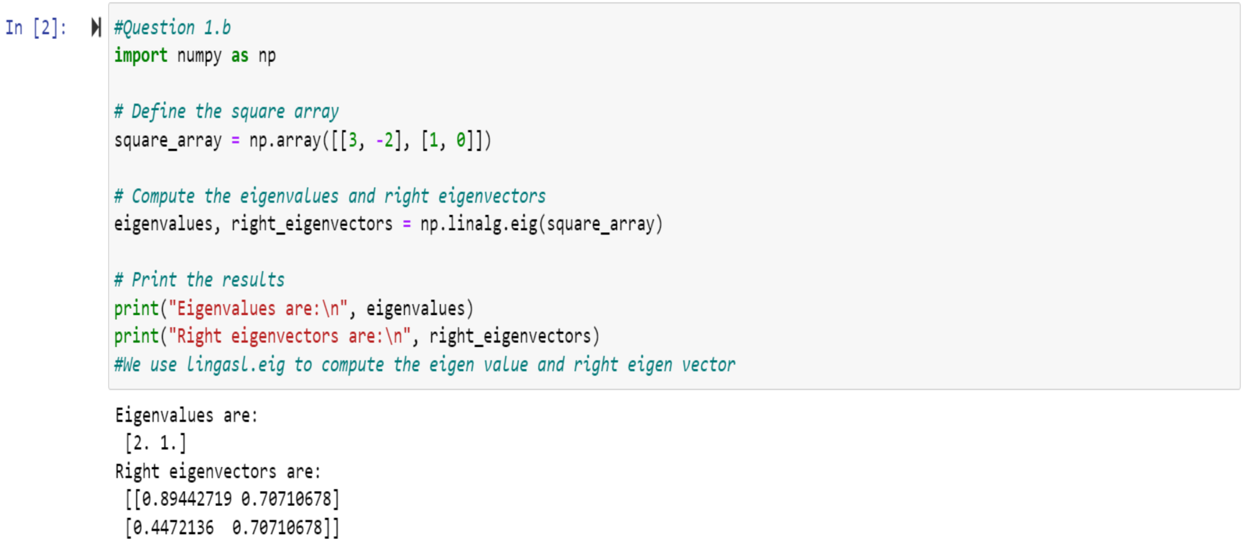
1.a:



```
In [1]:
#Question 1a
import numpy as np
randomvector = np.random.randint(low=1, high=21, size=15)
reshapedarray = randomvector.reshape((3, 5))
print("shape of Array:", reshapedarray.shape)
reshapedarray[np.arange(3), np.argmax(reshapedarray, axis=1)] = 0
print("Modified array:\n", reshapedarray)

twodarray = np.empty((4, 3), dtype=np.int32)

print("Array shape, type, and data type:", twodarray.shape, type(twodarray), twodarray.dtype)
```

```
shape of Array: (3, 5)
Modified array:
 [[16  1  0 17  1]
 [ 5  7 10  0  5]
 [16  0  9  6  4]]
Array shape, type, and data type: (4, 3) <class 'numpy.ndarray'> int32
```

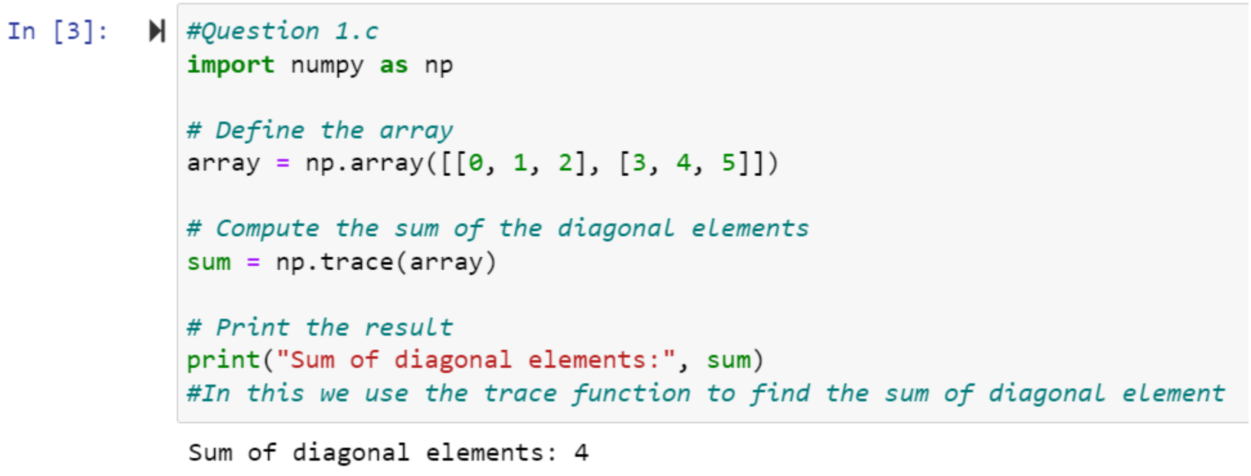
1.b: Write a program to compute the eigenvalues and right eigenvectors of a given square array



```
In [2]:
#Question 1.b
import numpy as np

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

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

# Print the results
print("Eigenvalues are:\n", eigenvalues)
print("Right eigenvectors are:\n", right_eigenvectors)
#We use lingasl.eig to compute the eigen value and right eigen vector
```

```
Eigenvalues are:
 [2. 1.]
Right eigenvectors are:
 [[0.89442719 0.70710678]
 [0.4472136  0.70710678]]
```

1.c: Compute the sum of the diagonal element

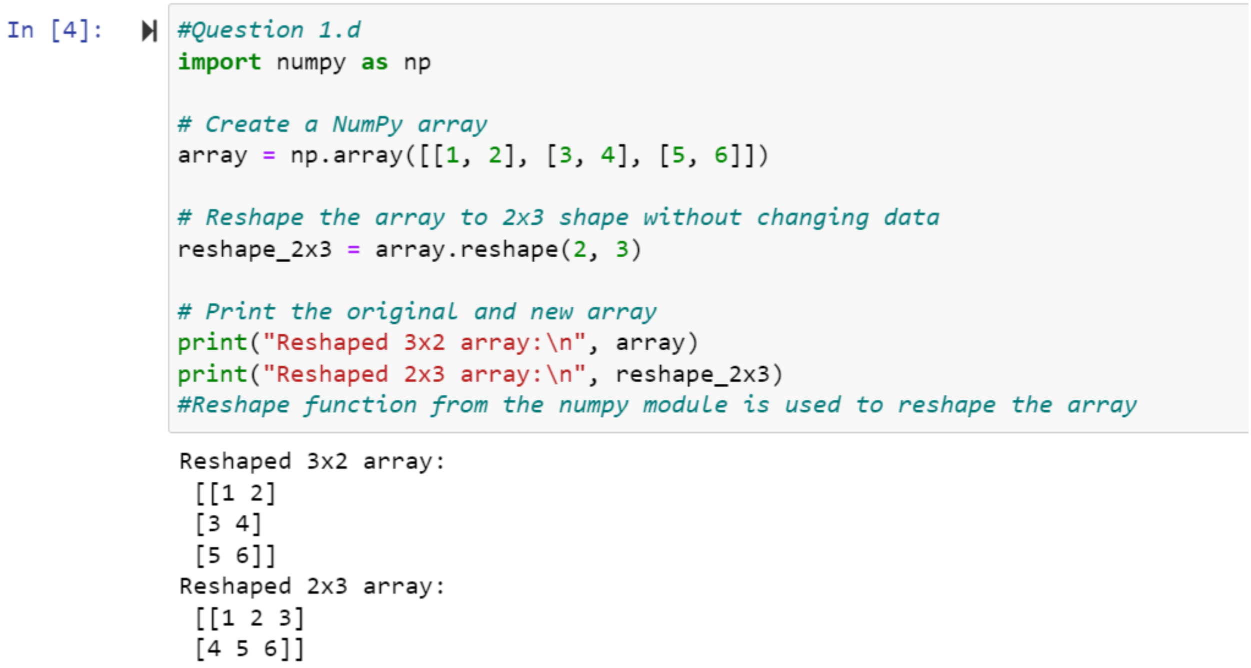


In [3]:

```python
#Question 1.c
import numpy as np

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

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

# Print the result
print("Sum of diagonal elements:", sum)
#In this we use the trace function to find the sum of diagonal element
```

```
Sum of diagonal elements: 4
```

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

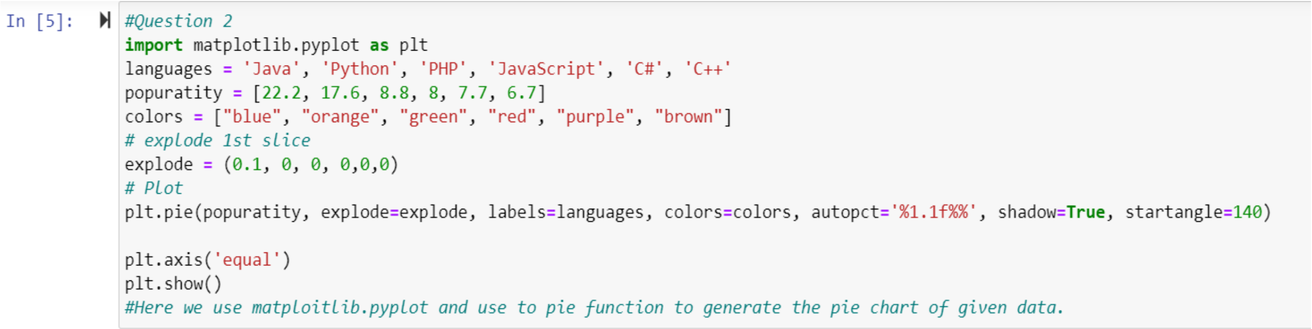


In [4]:

```python
#Question 1.d
import numpy as np

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

# Reshape the array to 2x3 shape without changing data
reshape_2x3 = array.reshape(2, 3)

# Print the original and new array
print("Reshaped 3x2 array:\n", array)
print("Reshaped 2x3 array:\n", reshape_2x3)
#Reshape function from the numpy module is used to reshape the array
```

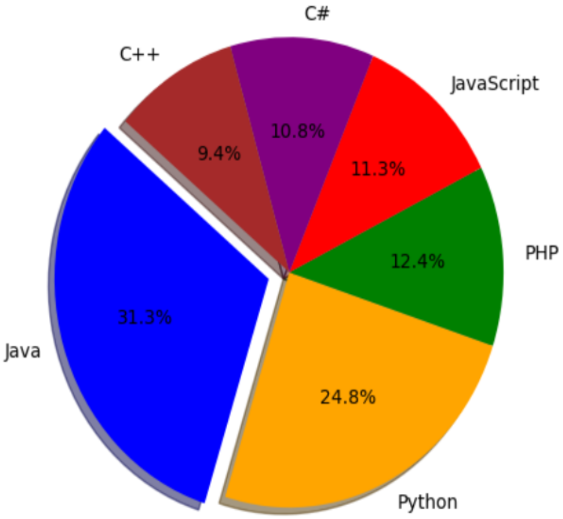
```
Reshaped 3x2 array:
 [[1 2]
 [3 4]
 [5 6]]
Reshaped 2x3 array:
 [[1 2 3]
 [4 5 6]]
```

Task 2: Using Matplotlib



In [5]:

```
#Question 2
import matplotlib.pyplot as plt
languages = 'Java', 'Python', 'PHP', 'JavaScript', 'C#', 'C++'
popuratity = [22.2, 17.6, 8.8, 8, 7.7, 6.7]
colors = ["blue", "orange", "green", "red", "purple", "brown"]
# explode 1st slice
explode = (0.1, 0, 0, 0,0,0)
# Plot
plt.pie(popuratity, explode=explode, labels=languages, colors=colors, autopct='%1.1f%%', shadow=True, startangle=140)

plt.axis('equal')
plt.show()
#Here we use matploitlib.pyplot and use to pie function to generate the pie chart of given data.
```


C#
C++
JavaScript
PHP
Java
Python