Assignment 3

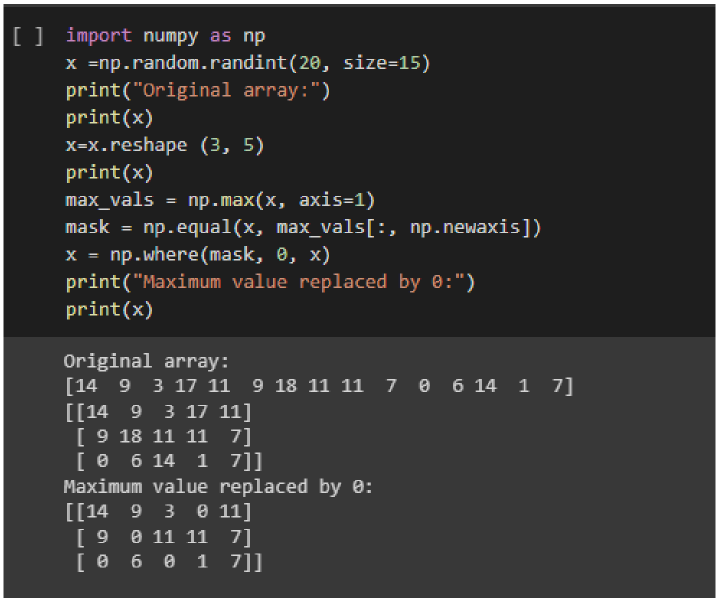
1. Numpy: a. Using NumPy create random vector of size 15 having only Integers in the range 1-20.

1. Reshape the array to 3 by 5

2. Print array shape.

3. Replace the max in each row by 0

```
import numpy as np

x =np.random.randint(20, size=15)

print("Original array:")

print(x)


x=x.reshape (3, 5)

print(x)


max_vals = np.max(x, axis=1)

mask = np.equal(x, max_vals[:, np.newaxis])

x = np.where(mask, 0, x)

print("Maximum value replaced by 0:")

print(x)
```



```
[ ]  import numpy as np
     x =np.random.randint(20, size=15)
     print("Original array:")
     print(x)
     x=x.reshape (3, 5)
     print(x)
     max_vals = np.max(x, axis=1)
     mask = np.equal(x, max_vals[:, np.newaxis])
     x = np.where(mask, 0, x)
     print("Maximum value replaced by 0:")
     print(x)
```

```
Original array:
[14  9  3 17 11  9 18 11 11  7  0  6 14  1  7]
[[14  9  3 17 11]
 [ 9 18 11 11  7]
 [ 0  6 14  1  7]]
Maximum value replaced by 0:
[[14  9  3  0 11]
 [ 9  0 11 11  7]
 [ 0  6  0  1  7]]
```

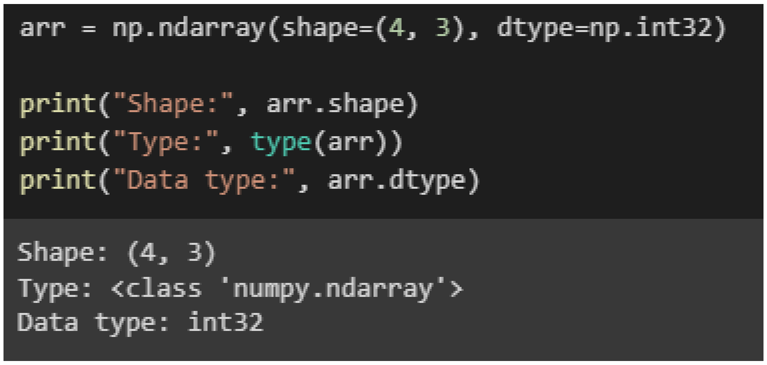
2 Create a 2-dimensional array of size 4 x 3 (composed of 4-byte integer elements), also print the shape, type and data type of the array

arr = np.ndarray(shape=(4, 3), dtype=np.int32)

print("Shape:", arr.shape)

print("Type:", type(arr))

print("Data type:", arr.dtype)



```
arr = np.ndarray(shape=(4, 3), dtype=np.int32)

print("Shape:", arr.shape)
print("Type:", type(arr))
print("Data type:", arr.dtype)
```

```
Shape: (4, 3)
Type: <class 'numpy.ndarray'>
Data type: int32
```

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

arr = np.array([[3, -2], [1, 0]])

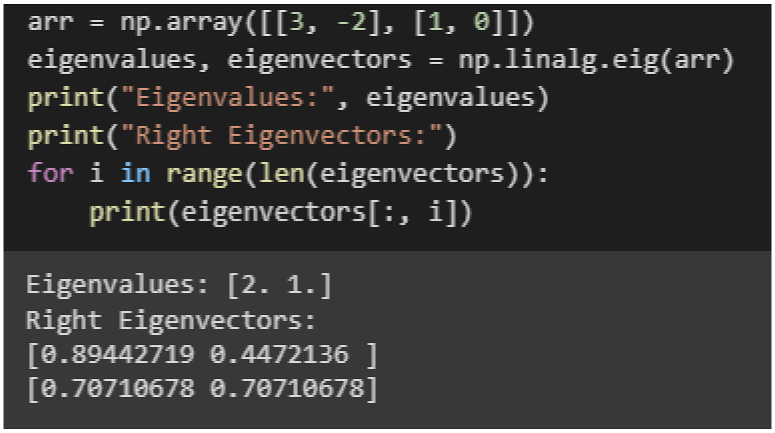
eigenvalues, eigenvectors = np.linalg.eig(arr)

print("Eigenvalues:", eigenvalues)

print("Right Eigenvectors:")

for i in range(len(eigenvectors)):

    print(eigenvectors[:, i])



```
arr = np.array([[3, -2], [1, 0]])
eigenvalues, eigenvectors = np.linalg.eig(arr)
print("Eigenvalues:", eigenvalues)
print("Right Eigenvectors:")
for i in range(len(eigenvectors)):
    print(eigenvectors[:, i])
```

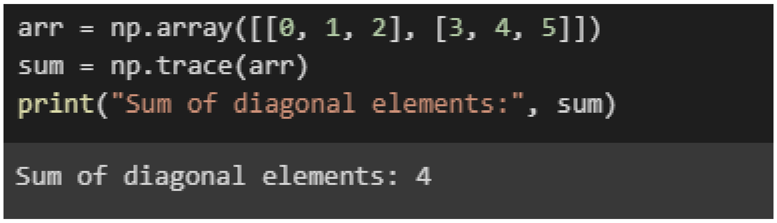
```
Eigenvalues: [2. 1.]
Right Eigenvectors:
[0.89442719 0.4472136 ]
[0.70710678 0.70710678]
```

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

arr = np.array([[0, 1, 2], [3, 4, 5]])

sum = np.trace(arr)

print("Sum of diagonal elements:", sum)



```
arr = np.array([[0, 1, 2], [3, 4, 5]])
sum = np.trace(arr)
print("Sum of diagonal elements:", sum)
```

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

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]

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

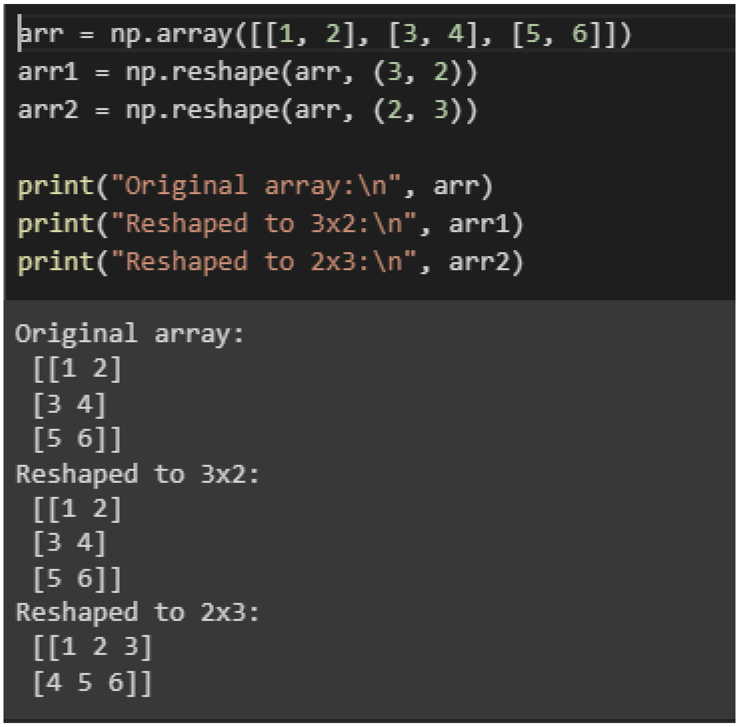
arr1 = np.reshape(arr, (3, 2))

arr2 = np.reshape(arr, (2, 3))

print("Original array:\n", arr)

print("Reshaped to 3x2:\n", arr1)

print("Reshaped to 2x3:\n", arr2)

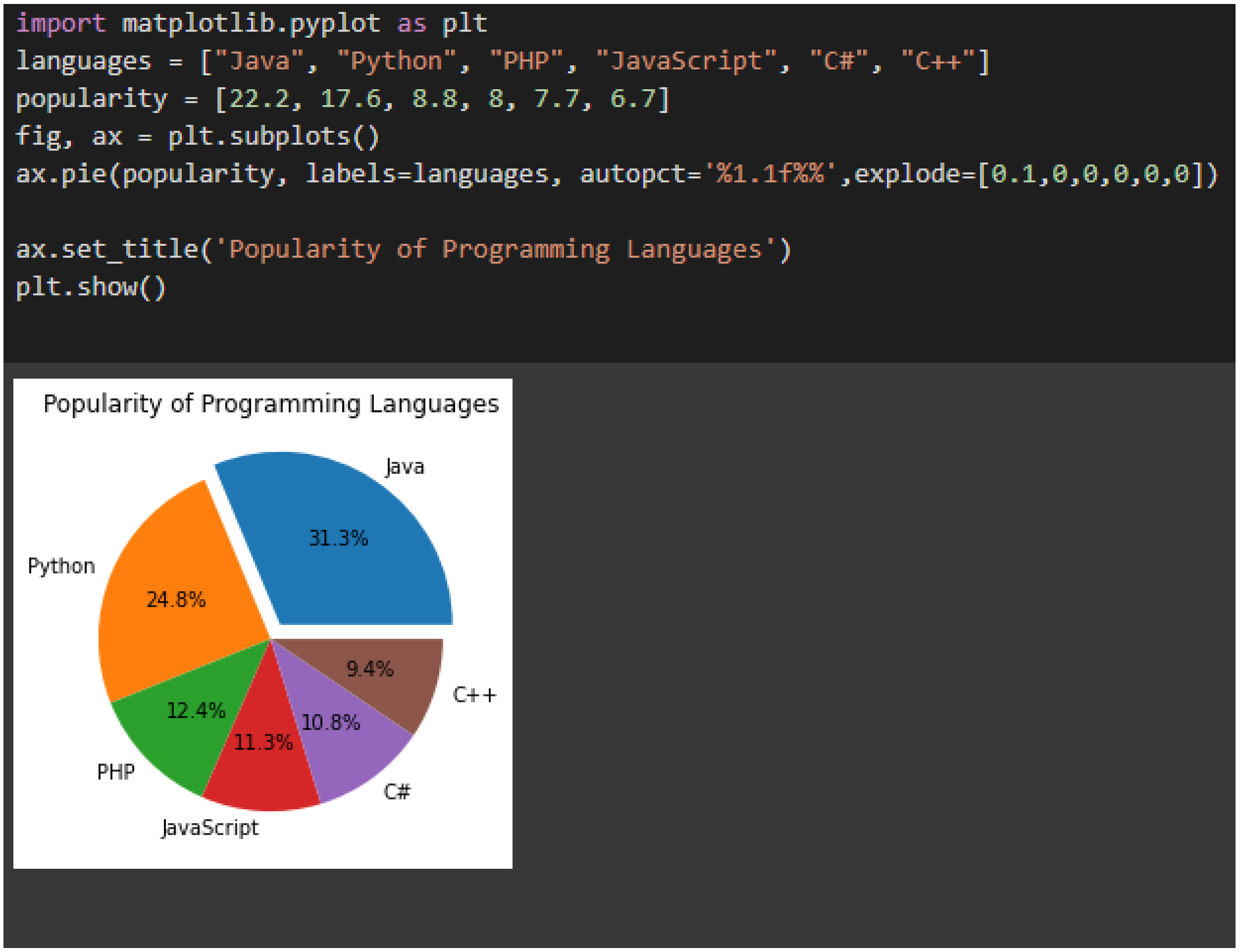


```
arr = np.array([[1, 2], [3, 4], [5, 6]])
arr1 = np.reshape(arr, (3, 2))
arr2 = np.reshape(arr, (2, 3))

print("Original array:\n", arr)
print("Reshaped to 3x2:\n", arr1)
print("Reshaped to 2x3:\n", arr2)
```

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

2. Matplotlib 1. Write a Python programming to create a below chart of the popularity of programming Languages. 2. Sample data: Programming languages: Java, Python, PHP, JavaScript, C#, C++ Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7

```
import matplotlib.pyplot as plt

languages = ["Java", "Python", "PHP", "JavaScript", "C#", "C++"]

popularity = [22.2, 17.6, 8.8, 8, 7.7, 6.7]

fig, ax = plt.subplots()

ax.pie(popularity, labels=languages, autopct='%1.1f%%',explode=[0.1,0,0,0,0,0])


ax.set_title('Popularity of Programming Languages')

plt.show()
```



```
import matplotlib.pyplot as plt
languages = ["Java", "Python", "PHP", "JavaScript", "C#", "C++"]
popularity = [22.2, 17.6, 8.8, 8, 7.7, 6.7]
fig, ax = plt.subplots()
ax.pie(popularity, labels=languages, autopct='%1.1f%%',explode=[0.1,0,0,0,0,0])

ax.set_title('Popularity of Programming Languages')
plt.show()
```