

## PRACTICAL NO. 6

**AIM:** Explain numpy. What are 1D,2D,3D arrays? Illustrate with examples the following functions:

Shape, dtype, empty, zeros, linspace, argmax, sort.

---

### THEORY:

NumPy is a Python library used for numerical computing. It provides support for arrays and matrices, as well as a range of mathematical operations that can be performed on them. NumPy is designed to be very efficient, with support for vectorization and broadcasting, making it ideal for large-scale data analysis.

1D, 2D, and 3D arrays are multi-dimensional arrays supported by NumPy:

1D arrays are arrays with a single dimension, similar to a list or vector. They are created using the `numpy.array()` function, and can be accessed using an index.

2D arrays are arrays with two dimensions, similar to a matrix. They are created using the `numpy.array()` function with nested lists, and can be accessed using two indices.

3D arrays are arrays with three dimensions, similar to a cube. They are created using the `numpy.array()` function with nested lists, and can be accessed using three indices.

`shape`: returns the dimensions of a NumPy array.

### Code:

```
import numpy as np
```

```
a = np.array([1, 2, 3])  
print(a.shape) # Output: (3,)
```

---

dtype: returns the data type of a NumPy array.

**Code:**

```
import numpy as np  
a = np.array([1, 2, 3])  
print(a.dtype) # Output: int64
```

---

empty: returns an array with uninitialized values.

**Code:**

```
import numpy as np  
  
a = np.empty((2, 2))  
print(a) # Output: [[2.68156159e+154 2.68156159e+154]  
#      [2.68156159e+154 2.68156159e+154]]
```

---

zeros: returns an array with all values initialized to zero.

**Code:**

```
import numpy as np
```

```
a = np.zeros((2, 2))
```

```
print(a) # Output: [[0. 0.]
```

```
         #      [0. 0.]]
```

---

linspace: returns an array with evenly spaced values between a start and end point.

**Code:**

```
import numpy as np
```

```
a = np.linspace(0, 10, 5)
```

```
print(a) # Output: [ 0.  2.5  5.  7.5 10.]
```

---

sort: sorts the values in an array.

**Code:**

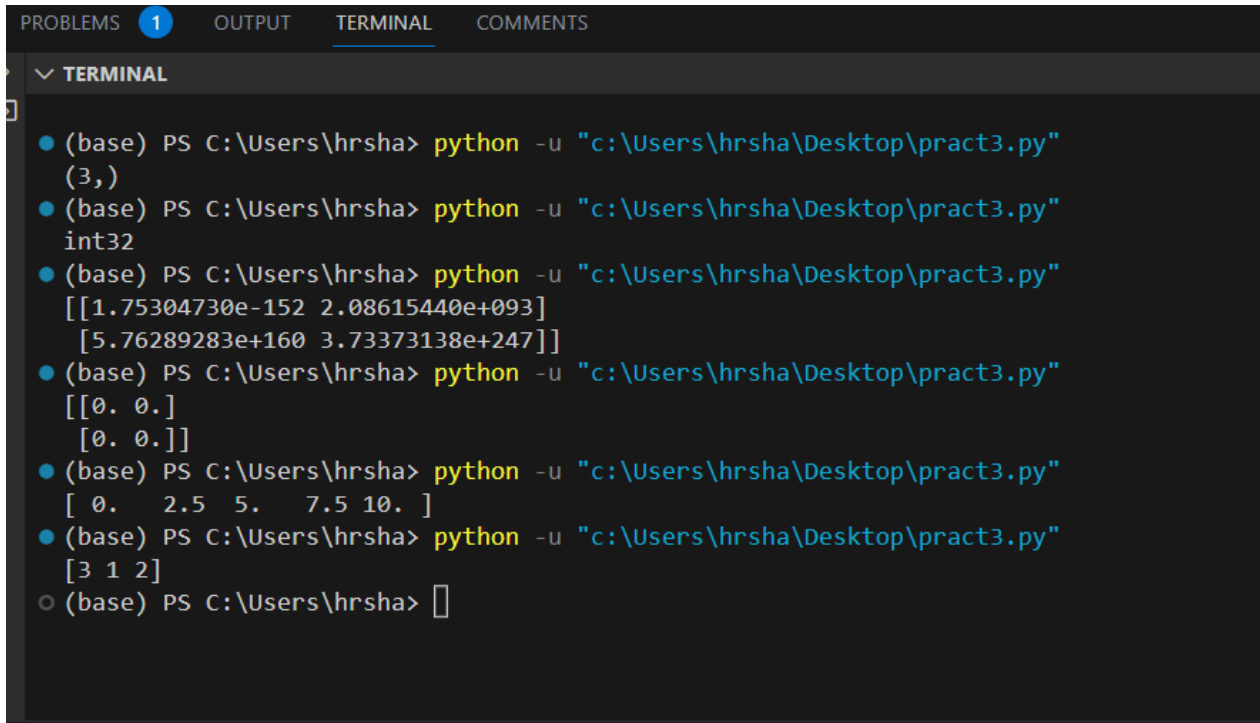
```
import numpy as np
```

```
a = np.array([3, 1, 2])
```

```
np.sort(a)
```

```
print(a) # Output: [1 2 3]
```

---



The screenshot shows a Jupyter Notebook interface with a terminal window open. The terminal displays the execution of a Python script named `pract3.py`. The output of the script is as follows:

```
(base) PS C:\Users\hrsha> python -u "c:\Users\hrsha\Desktop\pract3.py"
(3,)
(int32)
[[1.75304730e-152 2.08615440e+093]
 [5.76289283e+160 3.73373138e+247]]
[[0. 0.]
 [0. 0.]]
[ 0.  2.5  5.  7.5 10.]
[3 1 2]
```

## CONCLUSION:

Thus we have successfully executed programs .