1. **NumPy.shape()**

NumPy arrays have an attribute called shape that returns a tuple with each index having the number of corresponding elements.

The shape of an array is the number of elements in each dimension.

**Example:**

#program to print shape of an array

import numpy as np

arr = np.array([[1, 2, 3, 4],[5, 6,7,8]])

print(arr.shape)

**Output**

(2,4)

**NumPy.size()**

Numpy.size() function count the number of elements along a given axis.

***Syntax:****numpy.size(arr, axis=None)*

**Parameters:**

***arr:****[array\_like] Input data.****axis:****[int, optional] Axis(x,y,z) along which the elements(rows or columns) are counted. By default, give the total number of elements in a array****Returns:****[int] Return the number of elements along a given axis.*

1. **IDEA OF BROADCASTING IN NUMPY**

The term ***broadcasting***refers to how numpy treats arrays with different Dimension during arithmetic operations which lead to certain constraints, the smaller array is broadcast across the larger array so that they have compatible shapes.   
Broadcasting provides a means of vectorizing array operations so that looping occurs in C instead of Python as we know that Numpy implemented in C. It does this without making needless copies of data and which leads to efficient algorithm implementations. There are cases where broadcasting is a bad idea because it leads to inefficient use of memory that slow down the computation.

**EXAMPLE:**

import numpy as np

a = np.array([5, 7, 3, 1])

b = np.array([90, 50, 0, 30])

# array are compatible because of same Dimension

c = a \* b

print (c)

1. There are some libraries in python with specific applications and those makes python better than other libraries for numerical computation. And also one of the key features of python is numerous libraries and packages.

Let us see some libraries and its specific functions.

1. **SciPy(Scientific Numeric Library):**

SciPy is free and open source libray used for scientific computing and technical computing.

The library consists of modules of optimisation, image processing, FFT, special functions and signal processing.

The SciPy package includes Algorithms and functions like io,lib,signal,sparse,spatial,optimise,linals,interpolate,integrate and so on.

1. **Pandas(Data Analytical Library):**

Pandas is the most important data analysis library of python. Being open source it is used for data analysis with python. It can take formats of CSV or TSV files, or a SQL database and convert in to python data frames with rows and columns.

some functions of pandas are:

pd.DataFrame()

df.to\_filetype(filename)

df.median()

df.mean()

df.max()

df.min()

df.std() and so on.

Since we have many libraries like this available in python, python for numerical computations.

1. Python too supports file handling and allows users to handle files i.e., to read and write files, along with many other file handling options, to operate on files. The concept of file handling has stretched over various other languages, but the implementation is either complicated or lengthy, but like other concepts of Python, this concept here is also easy and short. Python treats files differently as text or binary and this is important. Each line of code includes a sequence of characters and they form a text file. Each line of a file is terminated with a special character, called the EOL or End of Line characters like comma {,} or newline character. It ends the current line and tells the interpreter a new one has begun. This is how python deals with files.

Let us consider some operations:

**Working of open() function:**

Before performing any operation on the file like read or write, first we have to open that file. For this, we should use Python’s inbuilt function open()

But at the time of opening, we have to specify the mode, which represents the purpose of the opening file.

f = open(filename, mode)

**Working of read() function:**

There is more than one way to read a file in Python. If you need to extract a string that contains all characters in the file then we can use **file.read()**.

# Python code to illustrate read() mode

file = open("file.txt", "r")

print (file.read())

### ****Creating a file using write() mode:****

Let’s see how to create a file and how write mode works:   
To manipulate the file, write the following in your Python environment:

# Python code to create a file

file = open('geek.txt','w')

file.write("This is the write command")

file.write("It allows us to write in a particular file")

file.close()

Like this we have many other operations in python to handle Files.

1. **Importances of numpy.empty():**

The numpy module of Python provides a function called **numpy.empty()**. This function is used to create an array without initializing the entries of given shape and type.

Just like **numpy.zeros()**, the **numpy.empty()** function doesn't set the array values to zero, and it is quite faster than the **numpy.zeros()**. This function requires the user to set all the values in the array manually and should be used with caution.

Main importance of numpy.empty() is to check whether a numpy array is emoty or not.

We can check it in any methods. Let us consider some methods.

1. numpy.any() to check if the NumPy array is empty in Python
2. numpy.size() method
3. By Converting it to list
4. Using arr.size()
5. By using arr.shape() method.