Numpy

NumPy is a Python library used for working with arrays.It also has functions for working in domain of linear algebra, fourier transform, and matrices. In Python we have lists that serve the purpose of arrays, but they are slow to process.

NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.

The array object in NumPy is called **ndarray**, it provides a lot of supporting functions that make working with ndarray very easy. NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently.

To use Numpy, you need to install it first. Open your terminal or command prompt and type the following command:

```
pip install numpy
```

Once NumPy is installed, import it in your applications by adding the import keyword:

```
import numpy
```

—---

Example

```
import numpy

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

print(arr)
```

—--

NumPy is usually imported under the np alias.

```
import numpy as np
```

Now the NumPy package can be referred to as np instead of numpy.

Example

```
import numpy as np
arr = np.array([1, 2, 3, 4, 5])
print(arr)
```

……………………………………………..

Example

Create a 2-D array containing two arrays with the values 1,2,3 and 4,5,6:

```
import numpy as np
arr = np.array([[1, 2, 3], [4, 5, 6]])
print(arr)
```

Example

Create an array with 5 dimensions and verify that it has 5 dimensions:

```
import numpy as np
arr = np.array([1, 2, 3, 4], ndmin=5)
print(arr)
print('number of dimensions :', arr.ndim)
```

…………………………………………………….

Data Types in NumPy

NumPy has some extra data types, and refer to data types with one character, like i for integers, u for unsigned integers etc.Below is a list of all data types in NumPy and the characters used to represent them.

- i - integer
- b - boolean
- u - unsigned integer
- f - float
- c - complex float
- m - timedelta
- M - datetime
- O - object
- S - string
- U - unicode string
- V - fixed chunk of memory for other type (void

Checking the Data Type of an Array

The NumPy array object has a property called dtype that returns the data type of the array:

Example

```
import numpy as np
arr = np.array([1, 2, 3, 4])
print(arr.dtype)
```

Example

```
import numpy as np
arr = np.array([1, 2, 3, 4])
print(arr.dtype)
```

Example

```
import numpy as np
arr = np.array([1, 2, 3, 4])
print(arr.dtype)
```

Pandas

Pandas is a Python library used for working with data sets.It has functions for analyzing, cleaning, exploring, and manipulating data.A Pandas DataFrame is a 2 dimensional data structure, like a 2 dimensional array, or a table with rows and columns.

Example

```
import pandas

mydataset = {

  'cars': ["BMW", "Volvo", "Ford"],

  'passings': [3, 7, 2]

}

myvar = pandas.DataFrame(mydataset)

print(myvar)
```

A Pandas DataFrame is a 2 dimensional data structure, like a 2 dimensional array, or a table with rows and columns.

```
import pandas as pd

data = {

  "calories": [420, 380, 390],

  "duration": [50, 40, 45]

}

#load data into a DataFrame object:

df = pd.DataFrame(data)

print(df)
```

Load Files Into a DataFrame

If your data sets are stored in a file, Pandas can load them into a DataFrame.

```
 import pandas as pd
df = pd.read_csv('data.csv')
print(df.to_string())


 #load data into a DataFrame object:


df = pd.DataFrame(data)
print(df)
```