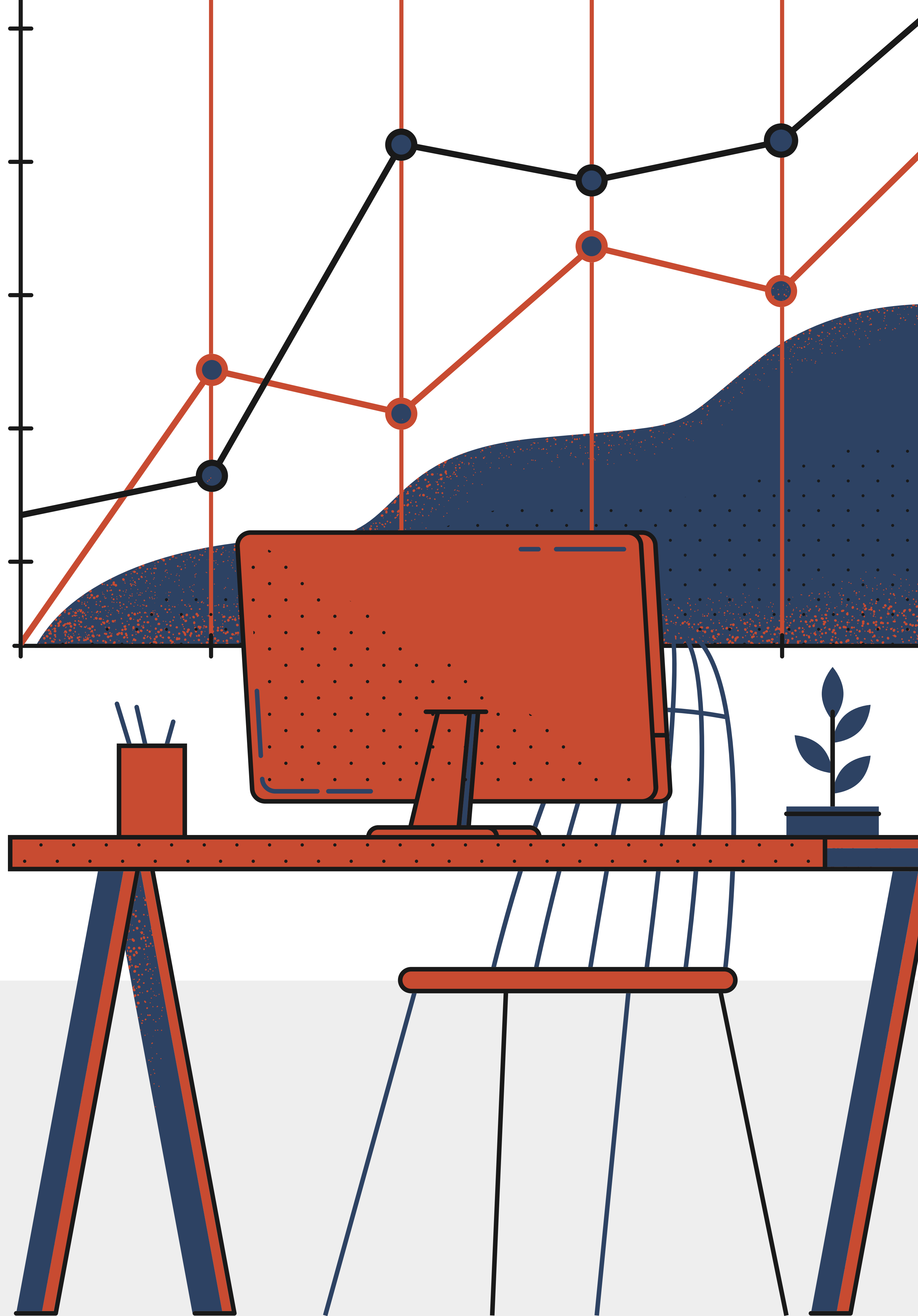
Wednesday, 6th July

CFI , IIT Madras

Intro to Numpy



SRINIVASAN KIDAMBI

ANIRUDH KALYAN

Table of Contents

Numpy is an open source library concerned with array manipulation. It has various functions for mathematical operations such as arithmetic, trigonometric, fourier analysis, matrices .... the list just goes on !

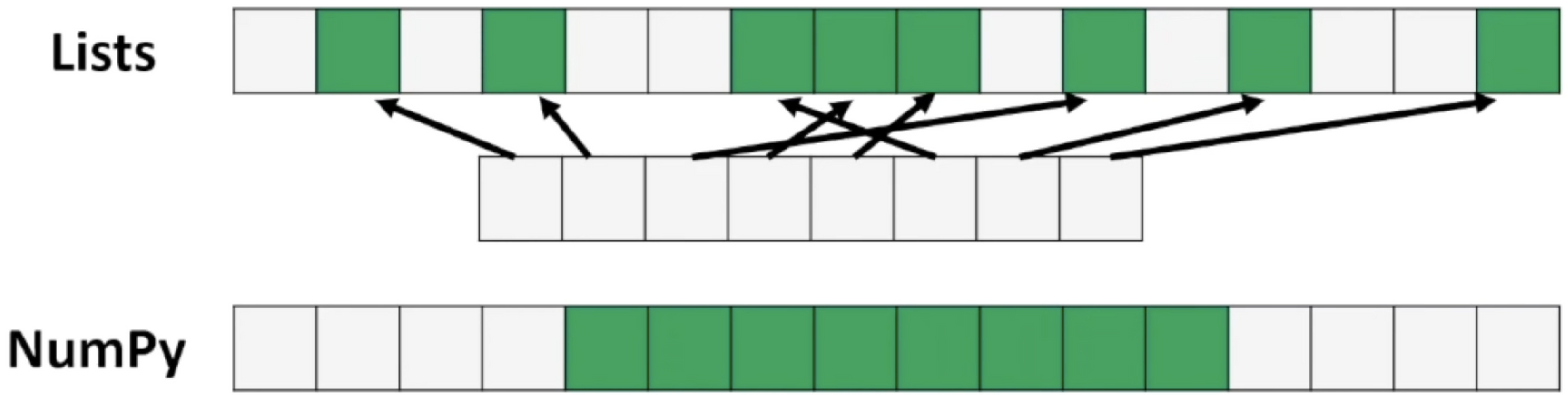
So what makes it so useful?

Numpy is a very important, useful and famous library as it offers multi-dimensional array capabilities. Lists store values, numpy arrays have built in functions that we can use to manipulate data on a large scale.

Much faster than lists --

- Lists store a lot more values than NumPy for the same value.
- You don't have to do type checking any time you read objects.
- Numpy uses contiguous memory, so it's easier to go through them and faster.

Why is NumPy Faster? - Contiguous Memory

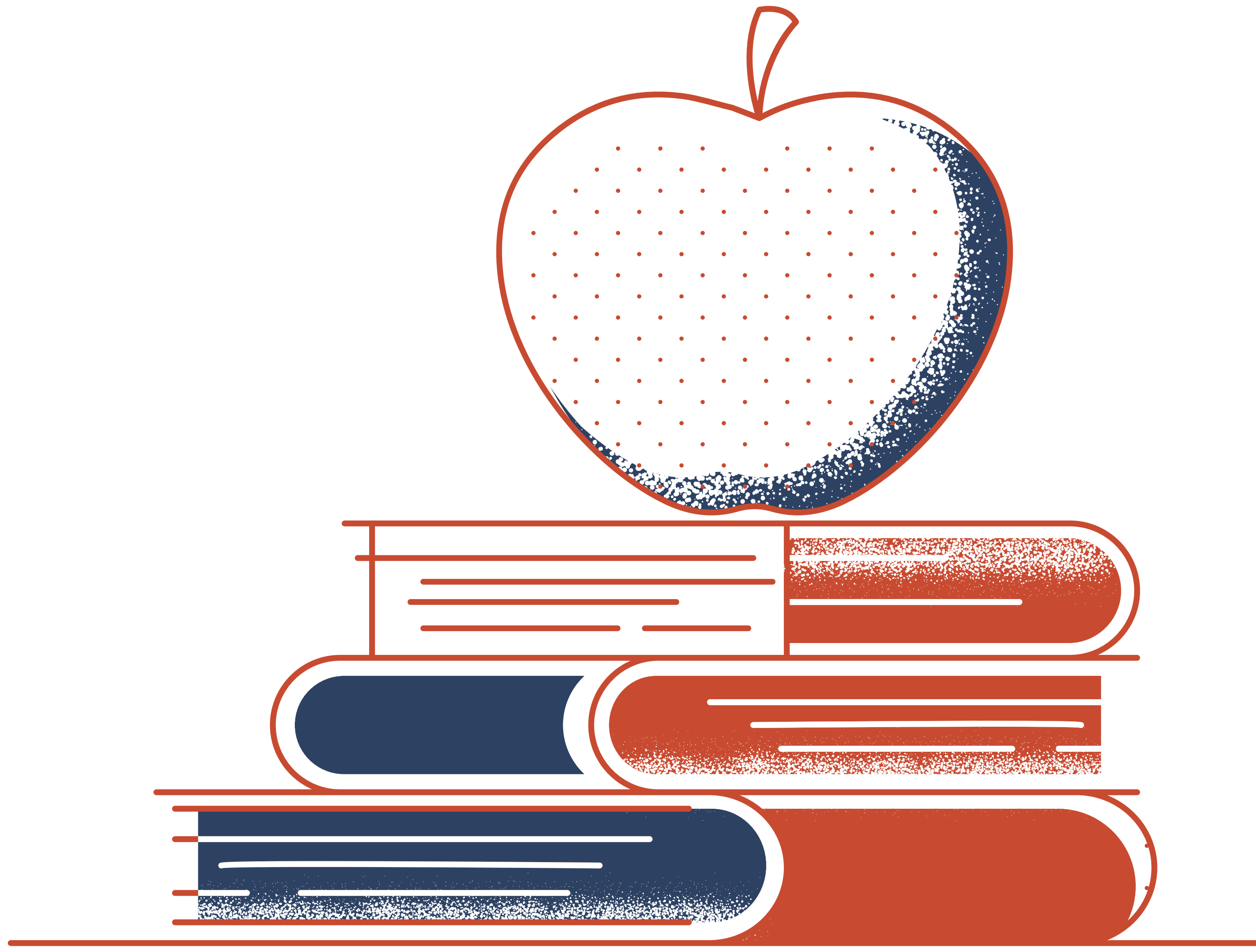


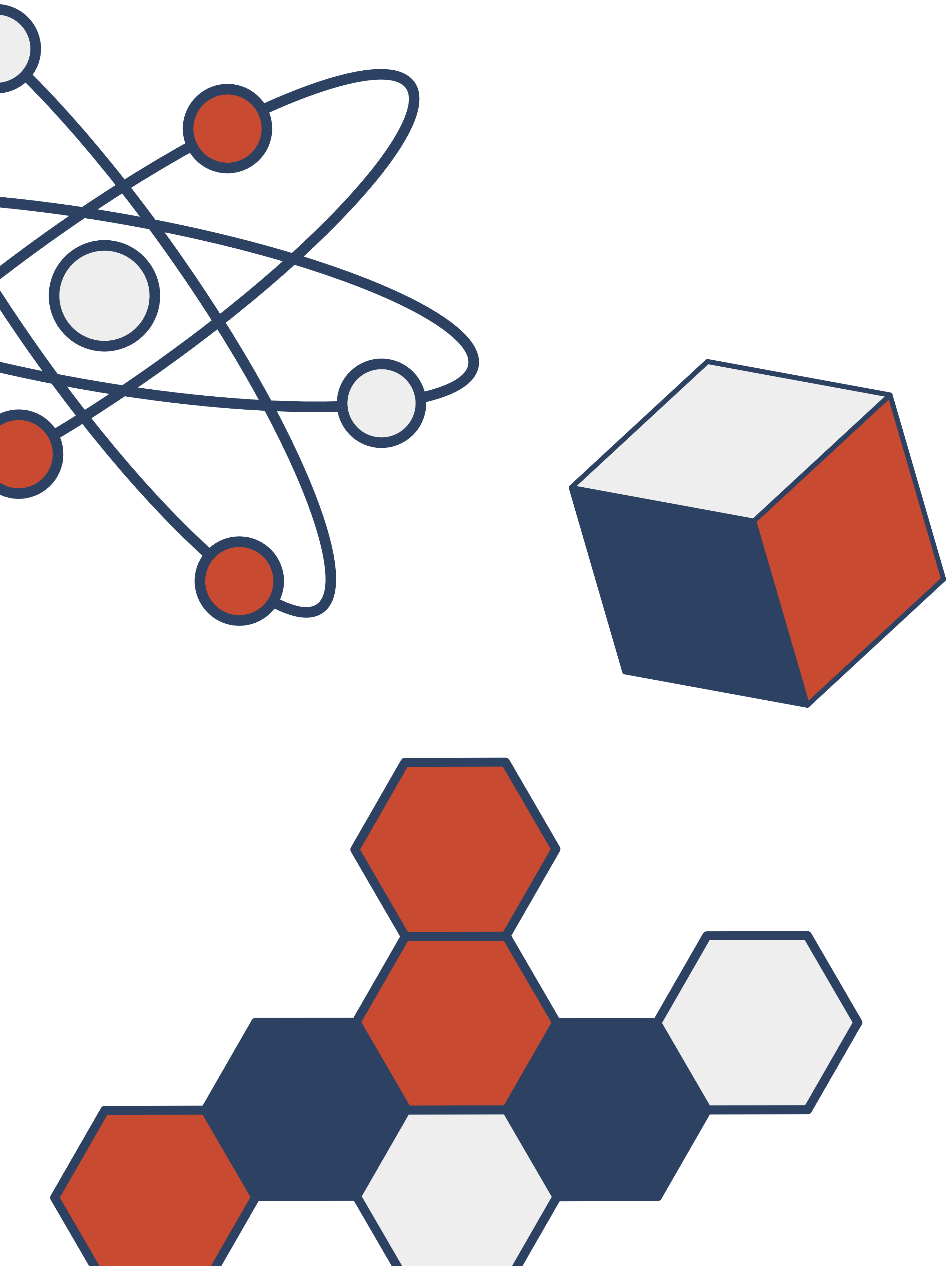
Indexing -

Array indexing is the same as accessing an array element. You can access an array element by referring to its index number. The indices in NumPy arrays start with 0, meaning that the first element has index 0, and the second has index 1 etc.

Slicing -

Slicing in python means taking elements from one given index to another given index. We pass slice instead of index like this: [start:end]





Numpy offers several built in arithmetic operations. Some of these are listed below

- **Raising the power**- numpy.power() is a function that allows the user to raise the value of every single element to a power n. This is useful for large scale data manipulation
- **Trigonometric** - Apply trigonometric functions to all elements
- **Add, subtract, multiply** - numpy.add, numpy.subtract, numpy.multiply etc

Shape Manipulation and Stacking

Shape-

Shape of a numpy array is determined by the number of elements along each 'axis' .

eg- [[2, 3],[3,4],[4,5]] has a shape (3,2)

Stacking-

Stacking is basically arranging arrays along axes to form bigger new arrays.

eg- a = [[2,3], [3,4]] and b = [[1,8], [0,4]]

stacking along the vertical axis,

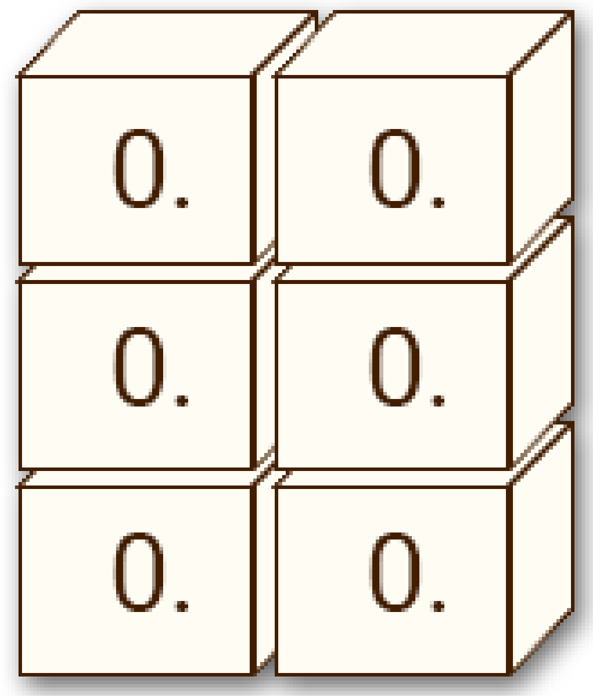
```
c = [[2,3],
      [3,4],
     [1,8],
     [0,4]]
```

Shape manipulation refers to modififying the shape of an array.

There are different way of doing this as per the situation. Some of the commonly used functions are:

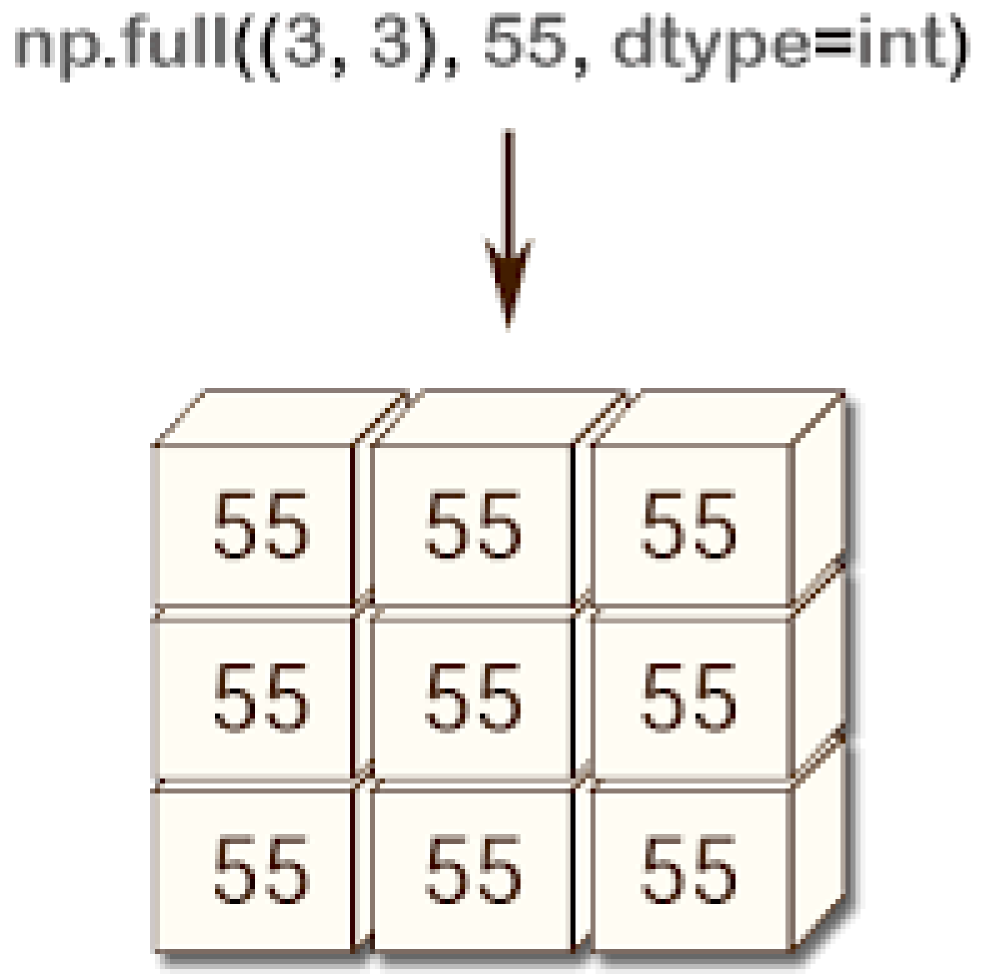
- array.ravel() - This method unravels a multidimensional array essentially flattening it. This functions creates a new array without affecting the original
- array.reshape() - This method reshapes the array without affecting the original array
- array.resize() - This method resizes/reshapes the array thereby affecting the original array itself.

Ones and Zeroes



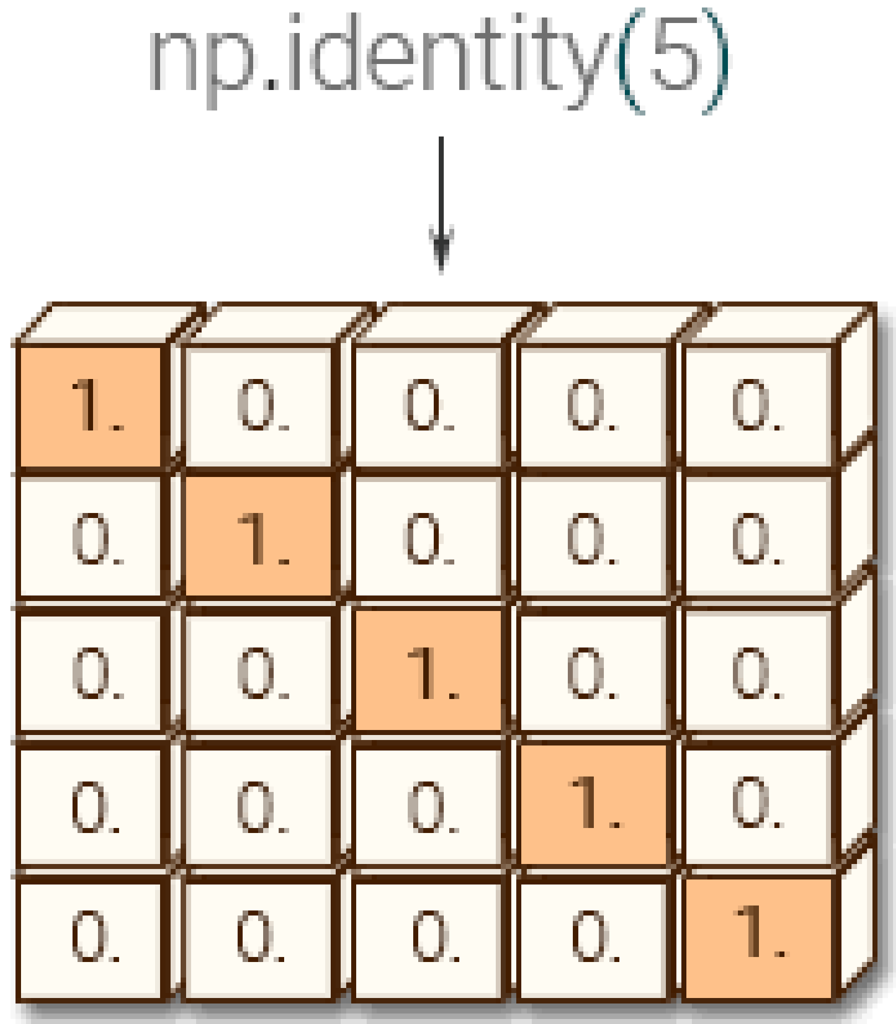
numpy.zeroes()

This is a function that allows the user to create an array of specific size with all 0 values. It is useful for initialization purposes.



numpy.full()

SImilarly , this function initializes an array with all entries as a number of the user's choice.



numpy.identity

An identity matrix is generated. This is especially useful when dealing with matrix multiplication and other matrix operations.

Numpy can also be used to read directly from a file :

```
filedata = np.genfromtxt('data.txt', delimiter=',')
filedata = filedata.astype('int32')
print(filedata)
```

Now supposing we want to track the elements that were, say greater than 100 in this array, we just need to do :

```
print(filedata>100)
```

The issue in dealing with this function is that reading csv files is not feasible. This is where Pandas comes in.....