Agenda

- **Installing and Importing Numpy**✅
- **Introduction to use case**✅
- **Motivation: Why to use Numpy? - How is it different from Python Lists?**✅
- **How numpy works under the hood?**✅
- **Creating a Basic Numpy Array**✅
  - From a List - `array()`, `shape`, `ndim` ✅
  - From a range and stepsize - `arange()` ✅
  - `type()` ndarray✅
- **Indexing and Slicing on 1D**
  - `Indexing`
  - `Slicing`
  - `Masking` (Fancy Indexing)
- **Operation on array**
  - `np.any`, `np.all`
- **Universal Functions (ufunc) on 1D array**
  - Aggregate Function/ Reduction functions - `sum()`, `mean()`, `min()`, `max()`
- **Usecase: calculate NPS**
  - loading data: `np.loadtxt()`

Airbnb Biz Case

Installing and Importing Numpy

In [1]:
```
!pip install numpy
```

```
Requirement already satisfied: numpy in /Users/nikhilsanghi/opt/anaconda3/lib/python3.9/site-packages (1.20.3)

[notice] A new release of pip is available: 23.0 -> 23.1.2
[notice] To update, run: pip install --upgrade pip
```

In [2]:
```
import numpy as np
```

In [3]:
```
a=[1,2,3,4,5]
a
```

Out[3]:
```
[1, 2, 3, 4, 5]
```

In [7]:
```
b=[i**2 for i in a]
b
```

Out[7]:
```
[1, 4, 9, 16, 25]
```

In [8]:
```
c=[]
for i in a:
    c.append(i**2)
c
```

Out[8]:
```
[1, 4, 9, 16, 25]
```

In [9]:
```
d=np.array([1,2,3,4,5])
d
```

Out[9]:
```
array([1, 2, 3, 4, 5])
```

In [11]:
```
d**2
```

Out[11]:
```
array([ 1,  4,  9, 16, 25])
```

In [12]:
```
d+2
```

Out[12]:
```
array([3, 4, 5, 6, 7])
```

In [13]:
```
d
```

Out[13]:
```
array([1, 2, 3, 4, 5])
```

In [14]:
```
d*10
```

Out[14]:
```
array([10, 20, 30, 40, 50])
```

Numpy arrays are faster

In [19]:
```
r=range(100000)
r
```

Out[19]:
```
range(0, 100000)
```

In [23]:
```
%timeit [i**2 for i in r]
```

```
26 ms ± 103 µs per loop (mean ± std. dev. of 7 runs, 10 loops each)
```

In [24]:
```
t=np.array(r)
t
```

Out[24]:
```
array([    0,     1,     2, ..., 99997, 99998, 99999])
```

In [25]:
```
%timeit t**2
```

```
29.5 µs ± 248 ns per loop (mean ± std. dev. of 7 runs, 10000 loops each)
```

In [ ]:
```

```

From a List - `array()`, `shape`, `ndim`

In [26]:
```
a=np.array([1,2,3,4,5,6])
a
```

Out[26]:
```
array([1, 2, 3, 4, 5, 6])
```

In [27]:
```
a.shape
```

Out[27]:
```
(6,)
```

In [28]:
```
a.shape[0]
```

Out[28]:
```
6
```

In [29]:
```
len(a)
```

Out[29]:
```
6
```

In [30]:
```
a.size
```

Out[30]:
```
6
```

In [32]:
```
a.ndim
```

Out[32]:
```
1
```

From a range and stepsize - `arange()`

In [34]:
```
for i in range(1,10):
    print(i)
```

```
1
2
3
4
5
6
7
8
9
```

In [35]:
```
for i in range(1,10,2):
    print(i)
```

```
1
3
5
7
9
```

In [36]:
```
for i in range(1,10,0.5):
    print(i)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/var/folders/hd/0z4dczbs56dj541b7q8w7s42w0000gn/T/ipykernel_81266/3064558270.py in <module>
----> 1 for i in range(1,10,0.5):
      2     print(i)

TypeError: 'float' object cannot be interpreted as an integer
```

In [37]:
```
np.arange(1,10,0.5)
```

Out[37]:
```
array([1. , 1.5, 2. , 2.5, 3. , 3.5, 4. , 4.5, 5. , 5.5, 6. , 6.5, 7. ,
       7.5, 8. , 8.5, 9. , 9.5])
```

In [38]:
```
np.arange(1,10,0.67)
```

Out[38]:
```
array([1.  , 1.67, 2.34, 3.01, 3.68, 4.35, 5.02, 5.69, 6.36, 7.03, 7.7 ,
       8.37, 9.04, 9.71])
```

`type()` ndarray

In [39]:
```
a=np.array([1,2,3,4,5])
a
```

Out[39]:
```
array([1, 2, 3, 4, 5])
```

In [40]:
```
type(a)
```

Out[40]:
```
numpy.ndarray
```

In [41]:
```
a.dtype
```

Out[41]:
```
dtype('int64')
```

In [42]:
```
a=np.array([1.0,2.0,3.0,4.0,5.0])
a
```

Out[42]:
```
array([1., 2., 3., 4., 5.])
```

In [43]:
```
type(a)
```

Out[43]:
```
numpy.ndarray
```

In [44]:
```
a.dtype
```

Out[44]:
```
dtype('float64')
```

In [45]:
```
a=np.array([True,False])
print(type(a))
print(a.dtype)
```

```
<class 'numpy.ndarray'>
bool
```

In [46]:
```
a=np.array([True,False,2])
print(a)
print(type(a))
print(a.dtype)
```

```
[1 0 2]
<class 'numpy.ndarray'>
int64
```

In [47]:
```
a=np.array([True,False,2,3.0])
print(a)
print(a.dtype)
```

```
[1. 0. 2. 3.]
float64
```

In [51]:
```
a=np.array(["Scaler",False,2,3.0])
print(a)
print(a.dtype)
```

```
['Scaler' 'False' '2' '3.0']
<U32
```

In [50]:
```
a
```

Out[50]:
```
array(['True', 'False', '2', '3.0', 'Scaler'], dtype='<U32')
```

In [57]:
```
a=np.array(["Scaler","Academy","Learner"],dtype="U7")
print(a)
print(a.dtype)
```

```
['Scaler' 'Academy' 'Learner']
<U7
```