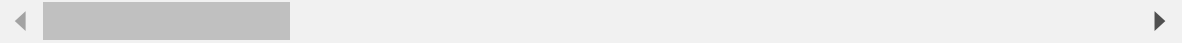


# BASICS OF PYTHON

## PART 1 - INTRODUCTION

Video link: <https://youtu.be/FggJNXN68fs> (<https://youtu.be/FggJNXN68fs>)

Codes: [https://www.youtube.com/redirect?event=video\\_description&redir\\_token=QUFFLUhqa0YtYXBuTXVFWWNPR1gxZ3Y1U2xQdUFVektBd3xB](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa0YtYXBuTXVFWWNPR1gxZ3Y1U2xQdUFVektBd3xB)  
([https://www.youtube.com/redirect?event=video\\_description&redir\\_token=QUFFLUhqa0YtYXBuTXVFWWNPR1gxZ3Y1U2xQdUFVektBd3xB](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa0YtYXBuTXVFWWNPR1gxZ3Y1U2xQdUFVektBd3xB))



See the commands to work fast.

Plugging in *numerical* expressions

```
In [1]: 255+45*10-100
```

```
Out[1]: 605
```

```
In [2]: 91%10
```

```
Out[2]: 1
```

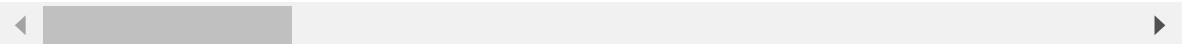
```
In [3]: 4**1.5
```

```
Out[3]: 8.0
```

## PART 2 - LISTS, TUPLES, ARRAYS

Video link: [https://youtu.be/xrLi9qq\\_wK8](https://youtu.be/xrLi9qq_wK8) ([https://youtu.be/xrLi9qq\\_wK8](https://youtu.be/xrLi9qq_wK8))

Codes: [https://www.youtube.com/redirect?event=video\\_description&redir\\_token=QUFFLUhqbEMxQmtyUWRDT0pjZmZzckZ1RFRBZXlneTU3Z3xB](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbEMxQmtyUWRDT0pjZmZzckZ1RFRBZXlneTU3Z3xB)  
([https://www.youtube.com/redirect?event=video\\_description&redir\\_token=QUFFLUhqbEMxQmtyUWRDT0pjZmZzckZ1RFRBZXlneTU3Z3xB](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbEMxQmtyUWRDT0pjZmZzckZ1RFRBZXlneTU3Z3xB))



### Variables

```
In [4]: x=46
```

```
In [5]: y=54
```

```
In [6]: (x+y)**0.5
```

```
Out[6]: 10.0
```

```
In [7]: age= 19.5554
```

```
In [8]: z= 'My age is {}'.format(age)
```

```
In [9]: z
```

```
Out[9]: 'My age is 19.5554.'
```

watch the video again and complete this part.

## Lists, Tuples, Dictionaries & Arrays

### Lists

```
In [10]: l1= [8,42,6,636,[7,89],'It\'s a list']
```

```
In [11]: l1[2], l1[4], l1[4][1], l1[5]
```

```
Out[11]: (6, [7, 89], 89, "It's a list")
```

### Tuple

lists can be edited but we can't edit tuples.

```
In [12]: t1= (2,[12,8,9],2,4,'It\'s a tuple')
```

```
In [13]: t1[1], t1[4]
```

```
Out[13]: ([12, 8, 9], "It's a tuple")
```

### Dictionaries

```
In [14]: d1= {'Barca':'Messi', 'Real':'Ronaldo'}
```

```
In [15]: d1['Barca']
```

```
Out[15]: 'Messi'
```

### Arrays

```
In [16]: import numpy as np
```

```
In [17]: np
```

```
Out[17]: <module 'numpy' from 'C:\\ProgramData\\Anaconda3\\lib\\site-packages\\numpy\\__init___.py'>
```

```
In [18]: x= [4,6,5,8,6,6]
        y= [45,1,68,1,0,7]
```

```
In [19]: X, Y = np.array(x), np.array(y)
```

```
In [20]: X, Y
```

```
Out[20]: (array([4, 6, 5, 8, 6, 6]), array([45,  1, 68,  1,  0,  7]))
```

```
In [21]: X.shape, Y.shape
```

```
Out[21]: ((6,), (6,))
```

```
In [22]: x+y
```

```
Out[22]: [4, 6, 5, 8, 6, 6, 45, 1, 68, 1, 0, 7]
```

```
In [23]: X+Y
```

```
Out[23]: array([49,  7, 73,  9,  6, 13])
```

```
In [24]: 2*x
```

```
Out[24]: [4, 6, 5, 8, 6, 6, 4, 6, 5, 8, 6, 6]
```

```
In [25]: 2*X
```

```
Out[25]: array([ 8, 12, 10, 16, 12, 12])
```

```
In [26]: X*Y
```

```
Out[26]: array([180,   6, 340,   8,   0, 42])
```

```
In [27]: linsp= np.linspace(0,1,10)
```

```
In [28]: linsp
```

```
Out[28]: array([0.          , 0.11111111, 0.22222222, 0.33333333, 0.44444444,
               0.55555556, 0.66666667, 0.77777778, 0.88888889, 1.          ])
```

```
In [29]: ar= np.arange(0,25,2)
```

```
In [30]: ar
```

```
Out[30]: array([ 0,  2,  4,  6,  8, 10, 12, 14, 16, 18, 20, 22, 24])
```

```
In [31]: ar**2
```

```
Out[31]: array([ 0,  4, 16, 36, 64, 100, 144, 196, 256, 324, 400, 484, 576],
              dtype=int32)
```

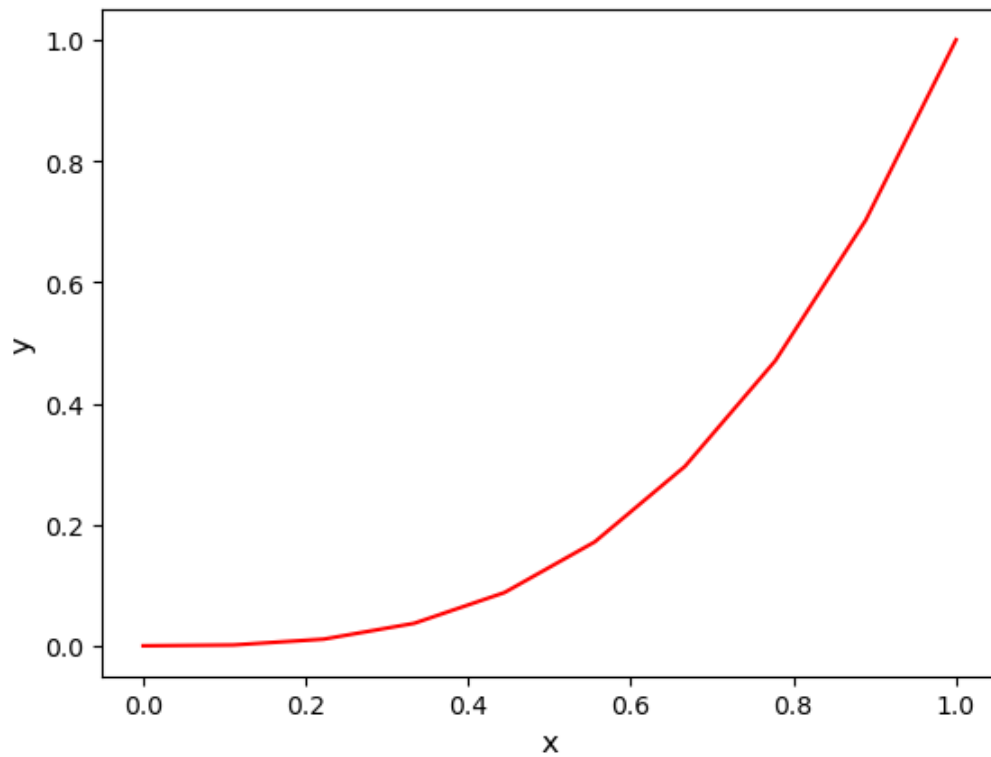
## Plotting

```
In [32]: import matplotlib.pyplot as plt
```

```
In [33]: x= linspace  
y= x**3
```

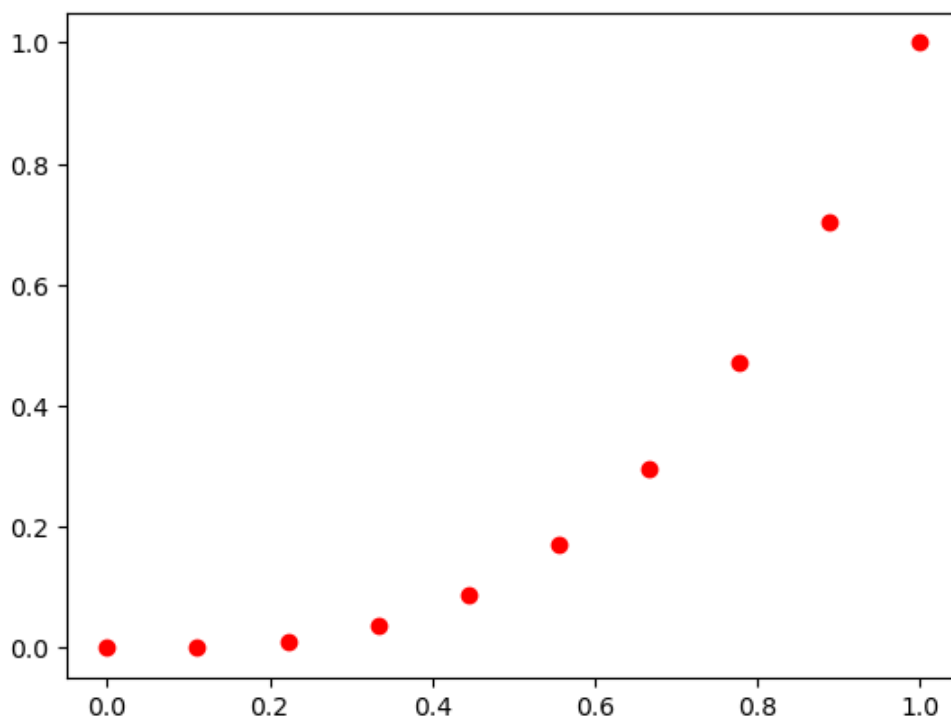
```
In [34]: plt.plot(x,y,'r')  
plt.xlabel('x', fontsize=12)  
plt.ylabel('y', fontsize=12)
```

```
Out[34]: Text(0, 0.5, 'y')
```



```
In [35]: plt.plot(x,y,'ro')
```

```
Out[35]: [matplotlib.lines.Line2D at 0x12bd5ab9df0]
```

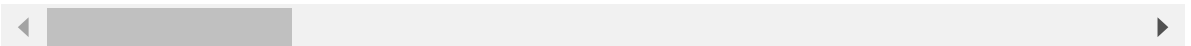


to know more google '*matplotlib plot*'.

## PART 3 - LOOPS, IF STATEMENTS

Video link: <https://youtu.be/3i3k9YgBROk> (<https://youtu.be/3i3k9YgBROk>)

Codes: [https://www.youtube.com/redirect?event=video\\_description&redir\\_token=QUFFLUhqBgtTmppb2EwYktBc3RxVGtDUEZfcGJNYUY4d3xBQ;](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqBgtTmppb2EwYktBc3RxVGtDUEZfcGJNYUY4d3xBQ;)  
([https://www.youtube.com/redirect?event=video\\_description&redir\\_token=QUFFLUhqBgtTmppb2EwYktBc3RxVGtDUEZfcGJNYUY4d3xBQ;](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqBgtTmppb2EwYktBc3RxVGtDUEZfcGJNYUY4d3xBQ;)



### forloops

#### 1. Regular loop

```
In [36]: for i in range(5):  
         print(i)
```

```
0  
1  
2  
3  
4
```

```
In [37]: tencube1st= []  
        for i in range (11):  
            tencube1st.append(i**3)
```

```
In [38]: tencube1st
```

```
Out[38]: [0, 1, 8, 27, 64, 125, 216, 343, 512, 729, 1000]
```

## 2. List loop

```
In [39]: l1= ['Jupyter', 'Pycharm', 'Replit']  
        for item in l1:  
            print(item)
```

```
Jupyter  
Pycharm  
Replit
```

## 3. Enumeration loop

```
In [40]: for i, item in enumerate(l1):  
        print('index {} contains {}'.format(i, item))
```

```
index 0 contains Jupyter  
index 1 contains Pycharm  
index 2 contains Replit
```

## 4. List comprehension loop

```
In [41]: l2= [i**3 for i in range(11)]
```

```
In [42]: l2
```

```
Out[42]: [0, 1, 8, 27, 64, 125, 216, 343, 512, 729, 1000]
```

## 5. Double forloop

```
In [43]: for i in range(4):  
        for j in range(3):  
            print('i={} and j={}'.format(i,j))
```

```
i=0 and j=0  
i=0 and j=1  
i=0 and j=2  
i=1 and j=0  
i=1 and j=1  
i=1 and j=2  
i=2 and j=0  
i=2 and j=1  
i=2 and j=2  
i=3 and j=0  
i=3 and j=1  
i=3 and j=2
```

## booleans and if statements

```
In [44]: a= True  
        b= False
```

```
In [45]: a or b
```

```
Out[45]: True
```

```
In [46]: a and b
```

```
Out[46]: False
```

```
In [47]: not(a)
```

```
Out[47]: False
```

```
In [48]: x= 43
```

```
In [49]: x==40, x==43, x!= 0
```

```
Out[49]: (False, True, True)
```

```
In [50]: checkit= 'esm' in 'desmos'
```

```
In [51]: checkit
```

```
Out[51]: True
```

```
In [52]: if (a or b)==True:  
        print('high')  
        else:  
        print('low')
```

high

```
In [53]: if (a or b)==True and (a and b)==True:  
        print('yes for both')  
        elif (a or b)==True and (a and b)!=True:  
        print('yes for condition 1 only')  
        elif (a or b)!=True and (a and b)==True:  
        print('yes for condition 2 only')  
        else:  
        print('no for both')
```

yes for condition 1 only

```
In [54]: if 0:  
        print('yes')  
        else:  
        print('no')
```

no

```
In [55]: if 12:
          print('yes')
        else:
          print('no')
```

yes

## PART 4 - SAMPLE PROBLEMS

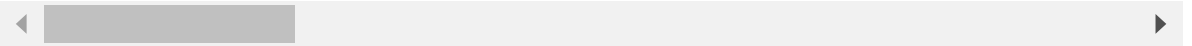
Video link: <https://youtu.be/X5YDtmg7q5o> (<https://youtu.be/X5YDtmg7q5o>)

Codes: [https://www.youtube.com/redirect?](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqBUpnYzZldzFOV2xnSV9maHp0djN5bkhaLTBqd3xBQ3)

[event=video\\_description&redir\\_token=QUFFLUhqBUpnYzZldzFOV2xnSV9maHp0djN5bkhaLTBqd3xBQ3](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqBUpnYzZldzFOV2xnSV9maHp0djN5bkhaLTBqd3xBQ3)

[/https://www.youtube.com/redirect?](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqBUpnYzZldzFOV2xnSV9maHp0djN5bkhaLTBqd3xBQ3)

[event=video\\_description&redir\\_token=QUFFLUhqBUpnYzZldzFOV2xnSV9maHp0djN5bkhaLTBqd3xBQ3](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqBUpnYzZldzFOV2xnSV9maHp0djN5bkhaLTBqd3xBQ3)



Problem 3: Add up every number from 1 to 999999 except for those that can be divided by 4 and 6.

```
In [56]: s= 0
          for i in range(1000000):
            if not(i%4==0) and not(i%6==0):
              s=s+i
```

```
In [57]: s
```

```
Out[57]: 333333666666
```

```
In [ ]:
```