

```
In [19]: import math  
math.sqrt(25)
```

Out[19]: 5.0

```
In [21]: math.ceil(3.45)
```

Out[21]: 4

```
In [23]: math.floor(3.45)
```

Out[23]: 3

```
In [25]: math.pow(3,2)
```

Out[25]: 9.0

```
In [31]: math.pi()
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[31], line 1  
----> 1 math.pi()  
  
TypeError: 'float' object is not callable
```

```
In [13]: print(math.e)
```

2.718281828459045

```
In [15]: import math as m  
m.sqrt(25)
```

Out[15]: 5.0

```
In [33]: from math import *
```

```
In [35]: pow(3,2)
```

Out[35]: 9.0

```
In [37]: round(pow(3,2))
```

Out[37]: 9

```
In [43]: #input
```

```
In [47]: x=input()  
y=input()  
z=x+y  
z
```

Out[47]: '8899'

```
In [49]: x1=int(input('enter first number'))
y1=int(input('enter second number'))
z1=x1+y1
z1
```

Out[49]: 6

```
In [53]: ch=input('enter a character')
print(ch)
```

hello

```
In [55]: ch3=int(input('enter an expression'))
print(ch3)
```

```
-----
ValueError                                Traceback (most recent call last)
Cell In[55], line 1
----> 1 ch3=int(input('enter an expression'))
      2 print(ch3)

ValueError: invalid literal for int() with base 10: '2+3'
```

```
In [57]: ch4=eval(input('enter the expression'))
print(ch4)
```

4

SET

```
In [8]: s={}
s
```

Out[8]: {}

```
In [10]: type(s)
```

Out[10]: dict

```
In [12]: s1=set()
s1
```

Out[12]: set()

```
In [14]: type(s1)
```

Out[14]: set

```
In [16]: s1
```

Out[16]: set()

```
In [18]: s1.add(10)
s1
```

```
Out[18]: {10}
```

```
In [20]: s2={200,3,1,20,10,10}#duplicate not
```

```
In [22]: s2
```

```
Out[22]: {1, 3, 10, 20, 200}
```

```
In [24]: s4={1,'f',3.3}
```

## set

```
In [27]: s4
```

```
Out[27]: {1, 3.3, 'f'}
```

```
In [29]: s4.add(1000)
```

```
In [31]: s4
```

```
Out[31]: {1, 1000, 3.3, 'f'}
```

```
In [33]: s4
```

```
Out[33]: {1, 1000, 3.3, 'f'}
```

```
In [35]: s4.pop()
```

```
Out[35]: 1000
```

```
In [37]: s2.remove(100)
```

```
-----
KeyError                                Traceback (most recent call last)
Cell In[37], line 1
----> 1 s2.remove(100)

KeyError: 100
```

```
In [39]: s2
```

```
Out[39]: {1, 3, 10, 20, 200}
```

```
In [41]: s2.discard(50000)
```

```
In [43]: s2
```

```
Out[43]: {1, 3, 10, 20, 200}
```

```
In [45]: s5=s2.copy()
```

```
In [47]: s5
```

```
Out[47]: {1, 3, 10, 20, 200}
```

```
In [49]: import sys
x=int(sys.argv[1])
y=int(sys.argv[2])
z=x+y
z
```

```
-----
ValueError                                Traceback (most recent call last)
Cell In[49], line 2
      1 import sys
----> 2 x=int(sys.argv[1])
      3 y=int(sys.argv[2])
      4 z=x+y

ValueError: invalid literal for int() with base 10: '-f'
```

```
In [1]: import sys
```

```
In [23]: sys.argv[0]
```

```
Out[23]: 'C:\\Users\\nandh\\anaconda3\\Lib\\site-packages\\ipykernel_launcher.py'
```

```
In [17]: x = 5
y = 6
z = x+y
z
```

```
Out[17]: 11
```

```
In [ ]:
```

```
In [15]: int('-f')
```

```
-----
ValueError                                Traceback (most recent call last)
Cell In[15], line 1
----> 1 int('-f')

ValueError: invalid literal for int() with base 10: '-f'
```

1ST FEB

```
In [52]: s5
```

```
Out[52]: {1, 3, 10, 20, 200}
```

```
In [54]: for i in enumerate(s5):  
         print(i)
```

```
(0, 1)  
(1, 3)  
(2, 20)  
(3, 200)  
(4, 10)
```

```
In [56]: for i in s5:  
         print (i)
```

```
1  
3  
20  
200  
10
```

```
In [60]: 1 in s5
```

```
Out[60]: True
```

```
In [62]: s5
```

```
Out[62]: {1, 3, 10, 20, 200}
```

```
In [66]: s5.update([1,2,3])
```

```
In [68]: s5
```

```
Out[68]: {1, 2, 3, 10, 20, 200}
```

```
In [70]: s6=s5.copy()
```

```
In [72]: s6
```

```
Out[72]: {1, 2, 3, 10, 20, 200}
```

```
In [138...
```

```
In [140... A.union(B)
```

```
Out[140... {1, 2, 3, 4, 5, 6, 7, 8}
```

```
In [142... A.union(B,C,)
```

```
Out[142... {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

```
In [144... A.union(B,C,45)
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[144], line 1  
----> 1 A.union(B,C,45)  
  
TypeError: 'int' object is not iterable
```

```
In [146... print(A)  
           print(B)  
           print(C)
```

```
{1, 2, 3, 4, 5}  
{4, 5, 6, 7, 8}  
{8, 9, 10}
```

```
In [148... A|B
```

```
Out[148... {1, 2, 3, 4, 5, 6, 7, 8}
```

```
In [150... A|B|C
```

```
Out[150... {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

```
In [152... A|B|C|D
```

```
-----  
NameError                                Traceback (most recent call last)  
Cell In[152], line 1  
----> 1 A|B|C|D  
  
NameError: name 'D' is not defined
```

## intersection

```
In [155... A&B
```

```
Out[155... {4, 5}
```

```
In [157... A.intersection(B)
```

```
Out[157... {4, 5}
```

```
In [159... A.intersection(C)
```

```
Out[159... set()
```

```
In [161... A.intersection(B,C)
```

```
Out[161... set()
```

```
In [163... A-B
```

Out[163... {1, 2, 3}

In [165... `A-C`

Out[165... {1, 2, 3, 4, 5}

In [167... `C-A`

Out[167... {8, 9, 10}

In [171... `A.symmetric_difference(B)`

Out[171... {1, 2, 3, 6, 7, 8}

In [173... `A.symmetric_difference(C)`

Out[173... {1, 2, 3, 4, 5, 8, 9, 10}

In [175... `A^C`

Out[175... {1, 2, 3, 4, 5, 8, 9, 10}

In [177... `B^C`

Out[177... {4, 5, 6, 7, 9, 10}

In [179... `A`

Out[179... {1, 2, 3, 4, 5}

In [183... `A1={1,2,3,4,5,6,7,8,9}`  
`B1={3,4,5,6,7,8}`  
`C1={10,20,30}`

In [185... `C1.isdisjoint(A1)`

Out[185... True

In [187... `C1.isdisjoint(B1)`

Out[187... True

In [189... `A2={1,2,3,4,5,6,7,8,9}`  
`B2={13,14,15,16,17,18}`  
`C2={10,20,30,40}`

In [191... `A2.issubset(B2)`

Out[191... False

In [193... `A`

Out[193... {1, 2, 3, 4, 5}

In [195... `sum(A)`

Out[195... 15

In [197... `max(A)`

Out[197... 5

In [199... `min(A)`

Out[199... 1

In [201... `for i in enumerate(A):`  
`print(i)`

(0, 1)

(1, 2)

(2, 3)

(3, 4)

(4, 5)

DICTIONARY

KEYS AND VALUES ARE PRESENT KEY:KEY\_VALUE it is mutable

In [212... `d={}`

In [214... `type(d)`

Out[214... dict

In [216... `d1={1:'one',2:'two',3:'three'}`

In [218... `d1`

Out[218... {1: 'one', 2: 'two', 3: 'three'}

In [220... `d1.keys()`

Out[220... dict\_keys([1, 2, 3])

In [222... `d1.values()`

Out[222... dict\_values(['one', 'two', 'three'])

In [224... `d1.items()`

Out[224... dict\_items([(1, 'one'), (2, 'two'), (3, 'three')])

In [226... `len(d1.items())`



Out[226... 3

In [228... `d1[1]`

Out[228... 'one'

In [230... `d1['one']`

```
-----  
KeyError                                Traceback (most recent call last)  
Cell In[230], line 1  
----> 1 d1['one']  
  
KeyError: 'one'
```

In [236... `print(d1.get(1))`

one

In [ ]: