Logical operator

AND,OR,NOT

```
In [4]: a = 5
        b = 4
```

```
In [6]: a<5 and b<5
```

```
Out[6]: False
```

```
In [8]: a<5 & b>4
```

```
Out[8]: False
```

```
In [10]: x = False
         x
```

```
Out[10]: False
```

```
In [12]: not x
```

```
Out[12]: True
```

```
In [14]: x = not x
         x
```

```
Out[14]: True
```

```
In [ ]:
```

NUmber system conversion

binary:base(0,1),and divide by number/2 & count in reverse order,octal base:(0,7),

hexadecimal:base(0,9 and a =10,b=11,c=13..till f ,(A,F/a,f)),

in real time we use in ip config..when we want to knoe about ip address

```
In [19]: 25
```

```
Out[19]: 25
```

```
In [21]: bin(25)
```

```
Out[21]: '0b11001'
```

```
In [27]: 0b11001

Out[27]: 25

In [29]: bin(45)

Out[29]: '0b101101'

In [33]: 0b101101

Out[33]: 45

In [35]: int(0b110011)

Out[35]: 51

In [37]: oct(13)

Out[37]: '0o15'

In [39]: oct(67)

Out[39]: '0o103'

In [43]: int(0o103)

Out[43]: 67

In [45]: hex(6)

Out[45]: '0x6'

In [47]: hex(70)

Out[47]: '0x46'

In [49]: hex(10)

Out[49]: '0xa'

In [51]: int(0x43)

Out[51]: 67

In [53]: 0xa

Out[53]: 10
```

Swap variable betwwn two numbers in different methods

In [56]:

```
x= 5
y=4
```

In [58]:

```
x,y = y,x
```

In [60]:

```
x
```

Out[60]:

```
4
```

In [62]:

```
y
```

Out[62]:

```
5
```

In [78]:

```
x1 = 67
x2 = 43
```

In [81]:

```
temp =x1
x1 = x2
x2= temp
```

In [83]:

```
print(x1)
print(x2)
```

```
43
67
```

In [85]:

```
# Using addition, sub method
a = 67
b = 45
```

In [87]:

```
a = a+b # 67+45 = 112
b = a-b # 112 - 45 = 67
a = a -b # 112 - 67 =45
print(a)
print(b)
```

```
45
67
```

In [89]:

```
print(0b101) # 101, 110 both are 3 bit
print(0b110)
```

```
5
6
```

In [91]:

```
print(bin(11))
print(0b1011)
```

```
0b1011
11
```

In [93]:

```
a = 7
b = 8
```

In [97]:

```
# another way of swap variable using  xor\
a = a^ b
b = a^b
a = a^b
```

In [100…

```
print(a)
print(b)
```

```
7
8
```

Bitwise opearator

6 Opeartor

1. Complement(~)

2. And(&)

3.OR(|)

4.XOR(^)

5.Left shift(<<)

6.Right Shift(>>)

Complement(~)

complement --> you will get this key below esc character

12 ==> 1100 ||

first thing we need to understand what is mean by complement.

complement means it will do reverse of the binary format i.e. - ~0 it will give you 1 ~1 it will give 0

12 binary format is 00001100 (complement of ~00001100 reverse the number - 11110011 which is (-13)

but the question is why we got -13

to understand this concept (we have concept of 2's complement

2's complement mean (1's complemen## complement means it will do reverse of the binary format i.e. - ~0 it will give you 1 ~1 it will give 0t + 1)

in the system we can store +Ve number but how to store -ve number

lets understand binary form of 13 - 00001101 + 1

```
In [112…  ~45

Out[112…  -46
```

```
In [114…  ~786

Out[114…  -787
```

```
In [116…  ~-56 #minus value

Out[116…  55
```

```
In [118…  ~9.78 #error(float)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[118], line 1
----> 1 ~9.78

TypeError: bad operand type for unary ~: 'float'
```

```
In [120…  ~6+10j #complex

Out[120…  (-7+10j)
```

BITWISE OPERATOR

bit wise and operator

AND - LOGICAL OPERATOR ||| & - BITWISE AND OPERATOR

(we know that 1 & 1 is 1)

12 - 00001100

13 - 00001101

when we are add both then then outut we will get as 12

```
In [136…  12 & 13
```

```
Out[136…  12
```

```
In [139…  12 & 13
```

```
Out[139…  12
```

```
In [141…  1 & 1
```

```
Out[141…  1
```

```
In [143…  1 | 1
```

```
Out[143…  1
```

```
In [145…  1 & 0
```

```
Out[145…  0
```

```
In [147…  # in XOR if the both number are different then we will get 1 or else we will get

          12 ^ 13
```

```
Out[147…  1
```

```
In [149…  25 ^ 30
```

```
Out[149…  7
```

```
In [151…  bin(25)
```

```
Out[151…  '0b11001'
```

```
In [153…  bin(30)
```

```
Out[153…  '0b11110'
```

```
In [155…  int(0b11110)
```

```
Out[155…  30
```

BIT WISE LEFT OPERATOR

bit wise left operator bydefault you will take 2 zeros ( )

10 binary operator is 1010 | also i can say 1010

```
In [163…   10<<2

Out[163…   40

In [165…   50<<3

Out[165…   400
```

BITWISE RIGHT SHIFT OPERATOR

```
In [170…   10>>2

Out[170…   2

In [172…   bin(20)

Out[172…   '0b10100'

In [174…   50>>2

Out[174…   12
```

Import Math Module

https://docs.python.org/3/library/math.html

```
In [178…   x = sqrt(625) # here sqrt is inbult fuc
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[178], line 1
----> 1 x = sqrt(625)

NameError: name 'sqrt' is not defined
```

```
In [180…   import math # importing math module
In [186…  x =  math.sqrt(625)
          x

Out[186…  25.0

In [188…  print(math.floor(2.9)) #minmum or least value

        2

In [190…  print(math.ceil(2.9)) #maximum or highest value

        3

In [192…  print(math.pow(6,2))

        36.0

In [194…  print(math.pi) #constant value

        3.141592653589793

In [196…  print(math.e)

        2.718281828459045

In [198…  import math as m
          m.sqrt(1225)

Out[198…  35.0

In [200…  import math as m
          m.pow(9,7)

Out[200…  4782969.0

In [202…  from math import pow
          pow(2,3)

Out[202…  8.0

In [204…  from math import ceil
          ceil(8.97)

Out[204…  9

In [208…  from math import *
          print(pow(4,6))
          print(ceil(5.5))

        4096.0
        6

In [210…  round(pow(9,2))

Out[210…  81
```

User input Function || comand line input

In [215…

```
r = input()
z = input()
c = r + z
print(c) #console is waiting for user to enter input
# also if you work in idle
```

```
56
```

In [217…

```
z1= input('first number')
z2 = input('Second number')
z3 = z1+z2
z3
```

Out[217…

```
'4567'
```

In [219…

```
type(z1)
type(z3)
```

Out[219…

```
str
```

In [223…

```
x1 = input("Enter number")
a = int(x1)
x2 = input("Enter a number")
b  = int(x2)
c = a +b
print(c)
```

```
124
```

from above code notice we are using many lines because fo that wasting some memory spaces as well

In [226…

```
a = int(input("1st number"))
b = int(input("2nd number"))
c = a+b
c
```

Out[226…

```
410
```

lets take input from the user in char format, but we dont have char format in python

In [229…

```
ch = input("Enter a char")
print(ch)
```

```
ratna
```

In [231…

```
print(ch[0])
```

```
r
```

In [233…

```
print(ch[-1])
```

```
a
```

In [235…

```
ch = input("Entera character")[0]
ch
```

Out[235…

```
't'
```

In [237…

```
ch = input("Enter ")[1:3]
ch
```

Out[237…

```
'di'
```

In [239…

```
ch = input("Enter")
ch
```

Out[239…

```
'45+67-0'
```

Eval fuc using input

In [244…

```
result = eval(input('enter a expr'))
print(result)
```

```
45
```

In [ ]: