

Logical operator

- and , or , not are logical operators
- Logical operators works upon compound conditional statement by using and , or , not

and

A	B	A and B
True	True	True
True	False	False
False	True	False
False	False	False

- They will work upon non bool type ????? Lets see

a = 5 b = 10 c = 15
if a<b and a<c
5<10 and 5<15

- And will return true if all are true . The conditions given in if are true

0 - False other than 0 is - True
if 5 and 6 returns True
T T

if -5 and 7 returns True # anything other than is True

If 0 and 6 returns False # 0 - False

- They will work on non bool type . It will not give the result T but it will give the second number (Anything other than 0 is True)

```
[Clang 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>> 5 and 6
6
>>> -5 and 6
6
>>> 5 or 6
5
>>> 1.2 and 1.3
1.3
>>> 0.0 and 4.0
0.0
>>> 0 and 9
0
>>> 5+2j and 3+1j
(3+1j)
>>> 5+2j and 0+0j
0j
>>> 0+0j and 5+2j
0j
>>> 'hi' or 'bye'
'hi'
>>> 'hi' and 'bye'
'bye'
>>> '' and 'bye'
''
>>>
```

•Empty string is treated as false

Short circuit : In **and** when the first one is false we don't have to check the second value cause it will be false cause and will return true if both the conditions are true otherwise false . If the first is True then it will check the second statement

OR Truth table

A	B	A or B
True	True	True
True	False	True
False	True	True
False	False	False

In OR it will check the first statement If it true than it will not check the second statement cause it will return true.

```
>>> 5 and 6
6
>>> 0 and 6
0
>>> 5 and 0
0
>>> 5 or 6
5
>>> 0 or 6
6
>>> 0 or 0
0
>>> 5 or 9
5
```

