

Identity Operator

Operator	Meaning	Example
is	True if the operands are identical	X is true
is not	True if the operands are not identical	X is not true

```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
>>> x = 2
>>> y = 2
>>> z = 3
>>> print (x is y) #check if both point same memory location
True
>>> print (y is z)
False
>>> print (x is not y) #check if both point separate memory location
False
>>> print (y is not z)
True
Ln: 75 Col: 4
```

Bitwise Operator

Operator	Description
&	Bitwise AND
	Bitwise OR
^	Bitwise XOR
~	Bitwise NOT
>>	Bitwise right shift
<<	Bitwise left shift

Truth table

1 = TRUE 0: FALSE

A	B	A & B	A B	A ^ B
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

What truth table is saying?

- if True in both condition Then only true in & case. If true in any one condition atleast then True in | case. if one true and one false then true in XOR ^ case.

```
In [1]: #Bitwise operator work with binary value
#example
a=10
b=8
#now lets see binary of both values at first
print(bin(a))
```

0b1010

```
In [2]: print(bin(b))
```

0b1000

```
In [3]: #ob is binary so do not count, binary value of 10 is 1010 and binary value of 8 is 1000.
```

```
In [4]: #now lets look case of &
print(a&b)
```

8

it giving value 8, how? lets look

a=1010 b=1000

in case of & :1&1=1 ; 0&0=0 ; 1&0=0 and 0&0=0 means- 1000

lets check again:

```
In [5]: print(a&b, bin(a&b))
```

8 0b1000

```
In [6]: #Simialrly, we can perform for or and xor
```

```
In [ ]:
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js