

Python Bootcamp & Masterclass

Operators & Expressions

Operators

Arithmetic

+

Comparison

>

Assignment

=

Logical

or

Bitwise

Membership

in

Identity

is

Arithmetic Operators

$3 + 2 = 5$

$3 \% 2 = 1$

$3 - 2 = 1$

$3 ** 2 = 9$

$3 * 2 = 6$

$3 // 2 = 1$

$3 / 2 = 1.5$

Comparison Operators

$3 < 3$

F

$3 > 3$

F

$3 \leq 3$

T

$3 \geq 3$

$3 == 3$

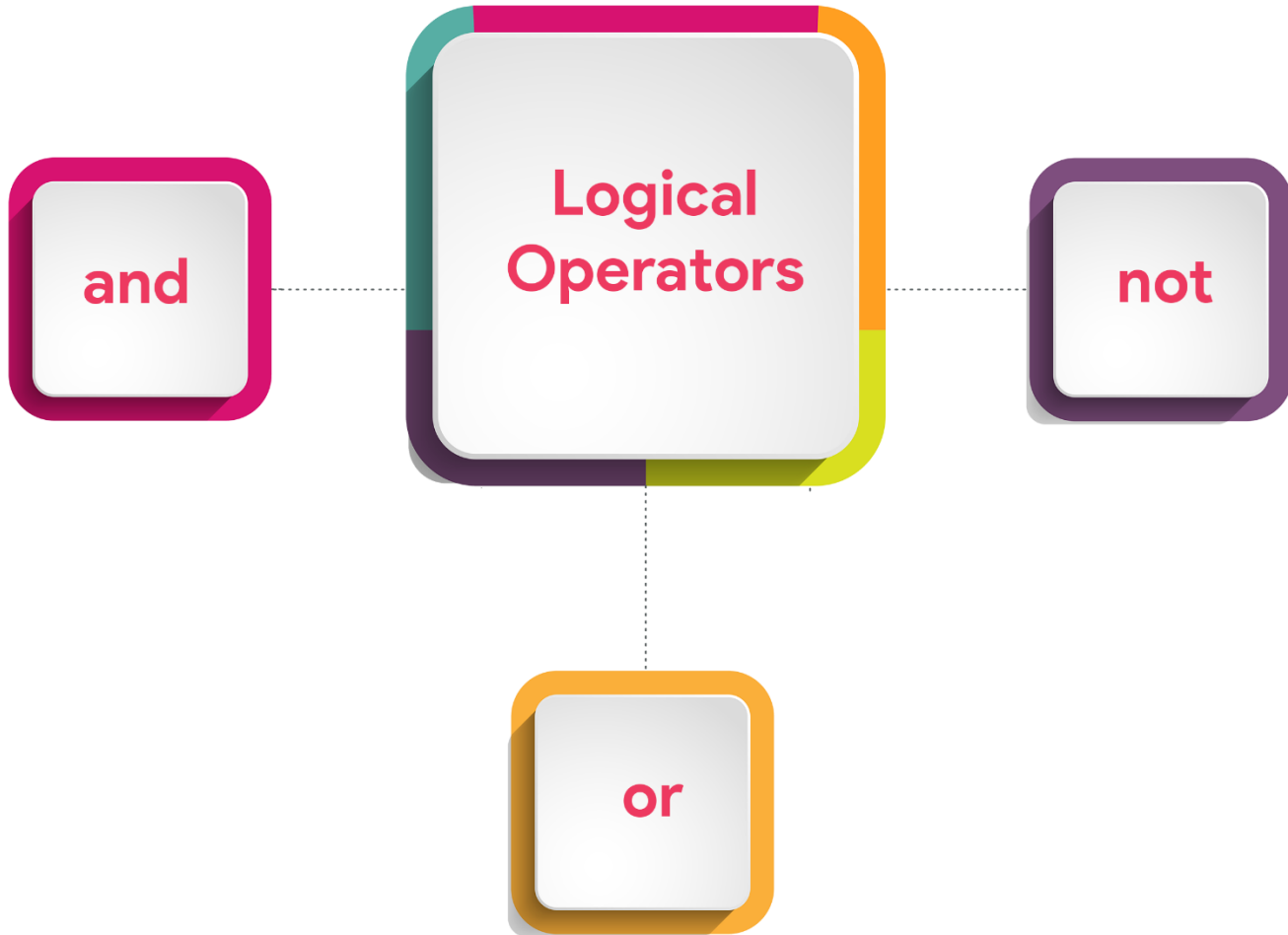
T

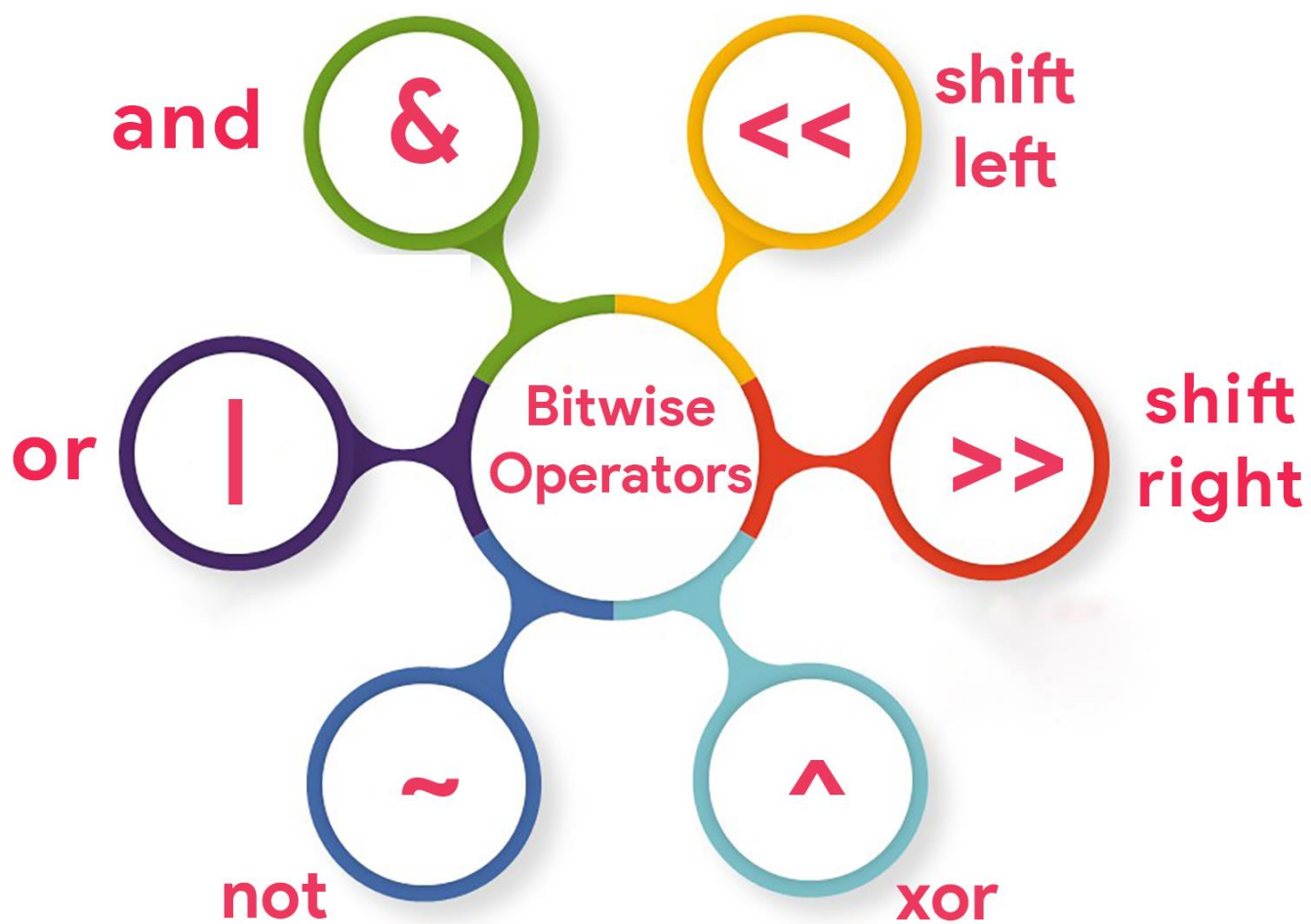
$3 != 3$

F

Assignment Operators

OPERATOR	USAGE	EXECUTION	RESULT (X = 5)
=	x = 5	x = 5	5
+=	x += 3	x = x + 3	8
-=	x -= 3	x = x - 3	2
*=	x *= 3	x = x * 3	15
/=	x /= 3	x = x / 3	1.667
%=	x %= 3	x = x % 3	2
//=	x //= 3	x = x // 3	1
**=	x **= 3	x = x ** 3	125
&=	x &= 3	x = x & 3	1
=	x = 3	x = x 3	7
^=	x ^= 3	x = x ^ 3	6
>>=	x >>= 3	x = x >> 3	0
<<=	x <<= 3	x = x << 3	40





in

**Membership
Operators**

not in

is

**Identity
Operators**

is not

Operator Precedence

OPERATOR	PRECEDENCE	DESCRIPTION	ASSOCIATIVITY	OPERATOR	PRECEDENCE	DESCRIPTION	ASSOCIATIVITY
()	1	Parentheses	left to right	^	8	Bitwise XOR	left to right
**	2	Exponent	right to left		9	Bitwise OR	left to right
+X	3	Unary plus	left to right	<	10	less than	left to right
-X	3	Unary minus	left to right	>	10	greater than	left to right
~X	3	Bitwise NOT	left to right	<=	10	less than or equal to	left to right
*	4	Multiplication	left to right	>=	10	greater than or equal to	left to right
/	4	Division	left to right	==	10	equal to	left to right
//	4	Floor division	left to right	!=	10	not equal to	left to right
%	4	Modulus	left to right	is	11	identity	left to right
+	5	Addition	left to right	is not	11	non identity	left to right
-	5	Subtraction	left to right	in	11	membership	left to right
<<	6	Bitwise shift left	left to right	not in	11	non membership	left to right
>>	6	Bitwise shift right	left to right	not	12	logical NOT	left to right
&	7	Bitwise AND	left to right	and	12	logical AND	left to right
				or	12	logical OR	left to right

Expression

An expression is a combination of variables, operators, objects, parentheses and calls to functions that Python can compute or evaluate to return the result. For example:

$x = 2 + 4 * 5 - 2 ** 3$ is an expression. Python uses precedence rules to evaluate this expression. Of all the operators present in this expression, exponentiation operator ($**$) has higher precedence than multiplication and division which have higher precedence than addition and subtraction. Python evaluates the expression progressively as

$$x = 2 + 4 * 5 - 8$$

$$x = 2 + 20 - 8$$

$$x = 22 - 8$$

$$x = 14$$

A boolean expression (or logical expression) evaluates to one of two states - **True** or **False**.

For example:

$5 > 3$ evaluates to **True** as the value of 5 is greater than the value of 3

```
x = 2 + 4 * 5 - 2 ** 3
```

```
x
```

14

```
5 > 3
```

True



Online Resources

For best python resources, please visit:



gknxt.com/python/

Python Bootcamp & Masterclass

Thank You
for your Rating & Review

