

# Operators

## Python Operators

Operators are used to perform operations on variables and values.

In the example below, we use the **+** operator to add together two values:

```
print(10 + 5)
```

```
15
```

**Python divides the operators in the following groups:**

- Arithmetic operators
- Assignment operators
- Comparison operators
- Logical operators
- Identity operators
- Membership operators
- Bitwise operators

# Arithmetic Operators

Arithmetic operators are used with numeric values to perform common mathematical operations:

Operator	Name
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus
**	Exponentiation
//	Floor division

# Assignment Operators

Assignment operators are used to assign values to variables:

Operator	Example
=	x = 5
+=	x += 3
-=	x -= 3
*=	x *= 3
/=	x /= 3
%=	x %= 3
//=	x //= 3
**=	x **= 3
&=	x &= 3
=	x  = 3
^=	x ^= 3
>>=	x >>= 3
<<=	x <<= 3

# Comparison Operators

Comparison operators are used to compare two values:

Operator	Name
<code>==</code>	Equal
<code>!=</code>	Not equal
<code>&gt;</code>	Greater than
<code>&lt;</code>	Less than
<code>&gt;=</code>	Greater than or equal to
<code>&lt;=</code>	Less than or equal to

# Logical Operators

Logical operators are used to combine conditional statements:

`and` Returns True if both statements are true

`or` Returns True if one of the statements is true

`not` Reverse the result, returns False if the result is true

## Example

```
x < 5 and x < 10
```

```
x < 5 or x < 4
```

```
not(x < 5 and x < 10)
```

# Identity Operators

Identity operators are used to compare the objects, not if they are equal, but if they are actually the same object, with the same memory location:

is	Returns True if both variables are the same object
----	--

is not	Returns True if both variables are not the same object
--------	--

## Example

```
x is y
```

```
x is not y
```