

# Operators

July 20, 2025

**0.1 In Python, operators are special symbols used to perform operations on variables and values.**

```
[1]: x = 10
     y = 50
     z = x + y
     print("Total: ", z)
```

Total: 60

```
[2]: z = x * y
     print("Multiple values: ", z)
```

Multiple values: 500

```
[16]: # Initialize variables
a = 10
b = 3
c = [1, 2, 3]
d = a # identity test

print("---- Arithmetic Operators ----")
print("a + b =", a + b)
print("a - b =", a - b)
print("a * b =", a * b)
print("a / b =", a / b)
print("a // b =", a // b)
print("a % b =", a % b)
print("a ** b =", a ** b)

print("\n---- Comparison Operators ----")
print("a == b:", a == b)
print("a != b:", a != b)
print("a > b:", a > b)
print("a < b:", a < b)
print("a >= b:", a >= b)
print("a <= b:", a <= b)
```

```

print("\n---- Assignment Operators ----")
x = 5
print("Initial x =", x)
x += 2
print("x += 2:", x)
x *= 3
print("x *= 3:", x)
x -= 1
print("x -= 1:", x)
x /= 2
print("x /= 2:", x)

print("\n---- Logical Operators ----")
print("a > 5 and b < 5:", a > 5 and b < 5)
print("a < 5 or b < 5:", a < 5 or b < 5)
print("not(a > b):", not(a > b))

print("\n---- Bitwise Operators ----")
print("a & b:", a & b)
print("a | b:", a | b)
print("a ^ b:", a ^ b)
print("~a:", ~a)
print("a << 1:", a << 1)
print("a >> 1:", a >> 1)

print("\n---- Membership Operators ----")
print("2 in c:", 2 in c)
print("5 not in c:", 5 not in c)

print("\n---- Identity Operators ----")
print("a is d:", a is d)
print("a is not b:", a is not b)

```

---- Arithmetic Operators ----

```

a + b = 13
a - b = 7
a * b = 30
a / b = 3.3333333333333335
a // b = 3
a % b = 1
a ** b = 1000

```

---- Comparison Operators ----

```

a == b: False
a != b: True
a > b: True
a < b: False

```

```
a >= b: True
a <= b: False
```

---- Assignment Operators ----

```
Initial x = 5
x += 2: 7
x *= 3: 21
x -= 1: 20
x /= 2: 10.0
```

---- Logical Operators ----

```
a > 5 and b < 5: True
a < 5 or b < 5: True
not(a > b): False
```

---- Bitwise Operators ----

```
a & b: 2
a | b: 11
a ^ b: 9
~a: -11
a << 1: 20
a >> 1: 5
```

---- Membership Operators ----

```
2 in c: True
5 not in c: True
```

---- Identity Operators ----

```
a is d: True
a is not b: True
```

```
[1]: 5 in {1, 5, 41, 47, 7,}
```

```
[1]: True
```

```
[ ]:
```