



Operator

Operator

Dalam pemrograman Python, **operator** adalah simbol khusus yang dapat memanipulasi nilai operand.

Bahasa programan Python mendukung jenis operator berikut:

1. Operator Aritmatika
2. Operator Perbandingan/Relasional
3. Operator Penugasan
4. Operator Bitwise
5. Operator Logika
6. Operator Keanggotaan
7. Operator Identitas

Operator

- Simbol yang memanipulasi nilai operand.

6 + 7 = 13

Operand

- Nilai di mana operator beroperasi.
- Nilai yang dimanipulasi oleh operator.

Operator Aritmatika



Operator Aritmatika

Operator **aritmatika** berguna untuk melakukan operasi matematika pada bilangan seperti **penambahan**, **pengurangan**, **perkalian**, **pembagian**, dll

Operator	Deskripsi	Contoh
+	Penjumlahan (Addition)	$x + y = 70$
-	Pengurangan (Subtraction)	$x - y = -30$
*	Perkalian (Multiplication)	$x * y = 400$
/	Pembagian (Division)	$y / x = 2$
%	Modulo (Modulus)	$y \% x = 2$
**	Perpangkatan (Exponent)	$x ** b = 100$
//	Floor Division	$9 // 2 = 4$ dan $9.0 // 2.0 = 4.0$

```
# Operator aritmatika
x = 7
y = 6

# Penjumlahan
print("Penjumlahan, x(7) + y(6) = ", x + y)

# Pengurangan
print("Pengurangan, x(7) - y(6) = ", x - y)

# Perkalian
print("Perkalian, x(7) * y(6) = ", x * y)

# Pembagian
print("Pembagian, x(7) / y(6) = ", x / y)

# Modulo (Sisa hasil bagi)
print("Modulo, x(7) % y(6) = ", x % y)

# Eksponen/Perpangkatan
print("Eksponen, x(7)**y(6) = ", x**y)

# Pembagian bilangan bulat yang dibulatkan
print("Floor Division, x(7)//y(6) = ", x//y)
```

```
# output
Penjumlahan, x(7) + y(6) = 13
Pengurangan, x(7) - y(6) = 1
Perkalian, x(7) * y(6) = 42
Pembagian, x(7) / y(6) = 1.1666666666666667
Modulo, x(7) % y(6) = 1
Eksponen, x(7)**y(6) = 117649
Floor Division, x(7)//y(6) = 1
```

Operator

Perbandingan/Relasional



Operator Perbandingan/Relasional

Operator **perbandingan** atau **relasional** berguna untuk membandingkan nilai. Operator ini akan mengembalikan **True** atau **False** sebagai hasil untuk kondisi tertentu.

Operator	Deskripsi	Contoh
==	Kondisi True, jika operan-1 sama dengan operan-2.	(x == y)
!=	Kondisi True, jika operan-1 tidak sama dengan operan-2.	(x != y)
<>	Kondisi True, jika operan-1 tidak sama dengan operan-2.	(x <> y)
>	Kondisi True, jika operan-1 lebih besar dari operan-2.	(x > y)
<	Kondisi True, jika operan-1 lebih kecil dari operan-2.	(x < y)
>=	Kondisi True, jika operan-1 lebih besar atau sama dengan operan-2.	(x >= y)
<=	Kondisi True, jika operan-1 lebih kecil atau sama dengan operan-2.	(x <= y)

- Python tidak menggunakan <>

```
# Operator perbandingan/relasional
x = 17
y = 13

# Operasi cek persamaan
print("Cek persamaan, x(17) == y(13) ", x == y)

# Operasi cek ketidaksamaan
print("Cek ketidaksamaan, x(17) != y(13) ", x != y)

# Operasi cek lebih besar
print("Cek lebih besar, x(17) >y(13) ", x > y)

# Operasi cek lebih kecil
print("Cek lebih kecil, x(17) <y(13) ", x < y)

# Operasi cek lebih besar atau sama dengan
print("Cek lebih besar atau sama dengan, x(17) >= y(13) ", x >= y)

# Operasi cek lebih kecil atau sama dengan
print("Cek lebih kecil atau sama dengan, x(17) <= y(13) ", x <= y)
```

```
# output
Cek persamaan, x(17) == y(13)  False
Cek ketidaksamaan, x(17) != y(13)  True
Cek lebih besar, x(17) >y(13)  True
Cek lebih kecil, x(17) <y(13)  False
Cek lebih besar atau sama dengan, x(17) >= y(13)  True
Cek lebih kecil atau sama dengan, x(17) <= y(13)  False
```


Operator Penugasan



Operator Penugasan (1)

Dalam Python, operator penugasan digunakan untuk menetapkan nilai ke dalam suatu variabel. Misalnya, `x = 7`. “=” adalah operator penugasan yang memberikan nilai numerik 7 yang berada di sisi kananya ke dalam variabel x yang berada di sisi kirinya.

Python juga mengizinkan penggunaan operator gabungan seperti `x += 7` yang menjumlah variabel dengan sebuah nilai numerik dan menyimpan hasil penjumlahannya ke dalam variable tersebut (operasi tersebut sama dengan operasi `x = x + 7`).

Operator Penugasan (2)

Operator	Deskripsi	Contoh
=	Menyimpan nilai operasi matematika operan kanan ke dalam operan kiri.	$z = x + y$
+=	Menjumlahkan operan kiri dengan operan kanan dan menyimpan hasilnya ke dalam operan kiri.	$z += x \Leftrightarrow z = z + x$
-=	Mengurangkan operan kiri dengan operan kanan dan menyimpan hasilnya ke dalam operan kiri.	$z -= x \Leftrightarrow z = z - x$
*=	Mengalikan operan kiri dengan operan kanan dan menyimpan hasilnya ke dalam operan kiri.	$z *= x \Leftrightarrow z = z * x$
/=	Membagi operan kiri dengan operan kanan dan menyimpan hasilnya ke dalam operan kiri.	$z /= x \Leftrightarrow z = z / x$
%=	Modulus operan kiri dengan operan kanan dan menyimpan hasilnya ke dalam operan kiri.	$z \% = x \Leftrightarrow z = z \% x$
**=	Menguadratkan operan kiri dengan operan kanan dan menyimpan hasilnya ke dalam operan kiri.	$z ** = x \Leftrightarrow z = z ** x$
//=	Menerapkan floor division pada operan kiri dengan operan kanan dan menyimpan hasilnya ke dalam operan kiri.	$z //= x \Leftrightarrow z = z // x$

```
# Operator penugasan (Assigns)
x = 3
y = 5

x += y
print("Hasil dari operasi x+=y adalah: ", x)

x *= y
print("Hasil dari operasi x*=y adalah: ", x)

x /= y
print("Hasil dari operasi x/=y adalah: ", x)

x %= y
print("Hasil dari operasi x%=y adalah: ", x)

x **= y
print("Hasil dari operasi x**=y adalah: ", x)

x // = y
print("Hasil dari operasi x// =y adalah: ", x)
```

```
# output
Hasil dari operasi x+=y adalah: 8
Hasil dari operasi x*=y adalah: 40
Hasil dari operasi x/=y adalah: 8.0
Hasil dari operasi x%=y adalah: 3.0
Hasil dari operasi x**=y adalah: 243.0
Hasil dari operasi x// =y adalah: 48.0
```

Operator Logika



Operator Logika

Operator **AND**, **OR**, **NOT** disebut sebagai operator logika. Operator logika berguna untuk memeriksa dua variabel terhadap kondisi yang diberikan dan hasilnya akan berupa kondisi benar (**TRUE**) atau salah (**FALSE**).

Operator	Deskripsi	Contoh
and	Operasi logika AND	Var1 and Var2
or	Operasi logika OR	Var1 or Var2
not	Operasi logika NOT	not Var1

Tabel Kebenaran (Truth Table)

Operator Logika **AND**

Var1	Opr.	Var2	Output
False	and	False	False
False		True	False
True		False	False
True		True	True

Operator Logika **OR**

Var1	Opr.	Var2	Output
False	or	False	False
False		True	True
True		False	True
True		True	True

Operator Logika **NOT**

Opr.	Var1	Output
not	False	True
	True	False

```
# Operator Logika
A = True
B = False

print('A and B = ', A and B)
print('A or B = ', A or B)
print('not A = ', not A)
```

```
# output
A and B = False
A or B = True
not A = False
```


Operator Bitwise



Operator Bitwise

Semua operasi dalam komputer diwakili oleh **bit**, yaitu serangkaian bilangan 0 (nol) dan 1 (satu). Operator **bitwise** memungkinkan kita untuk langsung **mengoperasikan** atau **memanipulasi** bit.

Sebagai contoh:

$x = 10$ (dalam biner 0000 1010)

$y = 4$ (dalam biner 0000 0100).

Operator Bitwise (2)

Operator Binary	Simbol	Deskripsi	Contoh
AND	&	Operasi logika AND	$x \& y$
OR		Operasi logika OR	$x y$
XOR*	^	Operasi logika XOR	$x \wedge y$
NOT	~	Operasi logika NOT	$\sim x$
Geser Kanan	>>	Operasi bit geser kanan	$x \gg 3$
Geser Kiri	<<	Operasi bit geser kiri	$y \ll 2$

* Operator Logika XOR

Var1	Opr.	Var2	Output
False	xor	False	False
False		True	True
True		False	True
True		True	False

```
# Operator bitwise
x = 7          # 7 dalam bilangan biner = 0000 0111
y = 3          # 3 dalam bilangan biner = 0000 0011

print("x AND y: ", x & y)          # Bitwise AND
print("x OR y: ", x | y)           # Bitwise OR
print("x XOR y: ", x ^ y)          # Bitwise XOR
print("NOT x: ", ~x)               # Bitwise NOT
print("x GESER KANAN y: ", x >> y) # Right Shift
print("x GESER KIRI y: ", x << y)  # Left Shift
```

```
# output
x AND y:  3
x OR y:   7
x XOR y:  4
NOT x:    -8
x GESER KANAN y:  0
x GESER KIRI y:  56
```

Operator Keanggotaan



Operator Keanggotaan

- Operator **keanggotaan** (*membership*) berguna untuk menguji apakah suatu nilai ditemukan dalam suatu urutan, yaitu *string*, *list*, *tuple*, *set*, atau *dictionary*.
- Ada dua operator keanggotaan di Python, '**in**' dan '**not in**'.
- Perhatikan bahwa kita hanya dapat menguji **keberadaan kunci** (dan bukan nilainya) dalam kasus *dictionary*.
- Output dari operasi keanggotaan ini adalah hanya kondisi benar (**True**) atau salah (**False**).

Operator	Deskripsi	Contoh
in	Hasil akan True , jika suatu nilai berada dalam urutan yang ditentukan dan False jika sebaliknya.	Var1 in Var2
not in	Hasil menjadi True , jika suatu nilai tidak berada dalam urutan yang ditentukan dan False jika sebaliknya.	Var1 not in Var2

```
# Operator Keanggotaan (Membership)

var1 = 'Sahabat Robonesia'      # string
var2 = {1:'a',2:'b',3:'c'}      # dictionary

print('R' in var1)
print('Sahabat' not in var2)
print(1 in var2)
print('a' in var2)
print(3 in var2)
```

```
# output
True
True
True
False
True
```

Operator Identitas



Fungsi Identitas (**id**)

- Python menyediakan fungsi **id()** untuk menampilkan nilai **id** (identitas) atau **alamat memori** dari suatu objek (Variabel).
- Nilai **id** yang diperoleh dari penggunaan fungsi **id()** adalah bilangan integer (atau bilangan long integer), yang merupakan nilai yang unik dan konstan untuk suatu objek **selama masa penggunaannya**.

```
# Penggunaan fungsi identitas id()
t = 157
print(id(t))      # Nilai id/identitas/alamat memori dari variabel 't'
```

```
# Output
2392671327344
```

Operator Identitas

- Operator **identitas** berguna untuk menguji apakah dua variabel berada pada **alamat memori** yang sama.
- Ada dua operator identitas di Python, '**is**' dan '**is not**'.
- Perhatikan bahwa **dua variabel yang memiliki nilai yang sama** tidak berarti keduanya identik jika memiliki nilai **id** (**alamat memori**) yang berbeda.
- Output dari operasi identitas ini adalah kondisi benar (**True**) atau salah (**False**).

Operator	Deskripsi	Contoh
is	Hasil akan True , jika variabel di kedua sisi operator menunjuk ke objek yang sama dan False jika sebaliknya.	Var1 is Var2
is not	Hasil akan False , jika variabel di kedua sisi operator menunjuk ke objek yang sama dan True jika sebaliknya.	Var1 is not Var2

```
a = 10
b = 10
print("Alamat memori variabel a: ", id(a))      # Nilai id variabel 'a'
print("Alamat memori variabel b: ", id(b))      # Nilai id variabel 'b'
print("Apakah a adalah b:", a is b)            # Operator identitas
print("Apakah a bukanlah b:", a is not b)       # Operator identitas

c = [2,4,6,8]
d = [2,4,6,8]
print("Alamat memori variabel c: ", id(c))      # Nilai id variabel 'c'
print("Alamat memori variabel d: ", id(d))      # Nilai id variabel 'd'
print("Apakah c adalah d:", c is d)            # Operator identitas
print("Apakah c bukanlah d:", c is not d)       # Operator identitas
```

Output

```
Alamat memori variabel a: 1843175162384
Alamat memori variabel b: 1843175162384
Apakah a adalah b: True
Apakah a bukanlah b: False
```

```
Alamat memori variabel c: 1843176493760
Alamat memori variabel d: 1843176524096
Apakah c adalah d: False
Apakah c bukanlah d: True
```



Terima Kasih