

NAMA: Azril Anggra Atmoko

NIM: 241351096

Pengenalan Pemrograman (Tugas 3)

OPERATOR PADA PYTHON

Apa itu Operator pada Python?

Operator pada Python adalah simbol atau konstruksi yang digunakan untuk melakukan operasi pada nilai atau variabel. Operator memungkinkan kita untuk melakukan berbagai jenis operasi aritmatika, logika, perbandingan, dan lainnya pada data.

Bahasa pemrograman Python mendukung berbagai macam operator, diantaranya :

1. **Operator Aritmatika (Arithmetic Operators)**
2. **Operator Penugasan (Assignment Operators)**
3. **Operator Perbandingan (Comparison (Relational) Operators)**
4. **Operator Logika (Logical Operators)**
5. **Operator Bitwise (Bitwise Operators)**
6. **Operator Keanggotaan (Membership Operators)**
7. **Operator Identitas (Identity Operators)**

1. Operator Aritmatika

Operator aritmatika Adalah operator yang digunakan dalam melakukan suatu proses operasi matematika, seperti penjumlahan, perkalian dan sebagainya.

Operator Aritmatika

Operator	Contoh	Penjelasan
Penjumlahan +	$1 + 3 = 4$	Menjumlahkan nilai dari masing-masing operan atau bilangan
Pengurangan -	$4 - 1 = 3$	Mengurangi nilai operan di sebelah kiri menggunakan operan di sebelah kanan
Perkalian *	$2 * 4 = 8$	Mengalikan operan/bilangan
Pembagian /	$10 / 5 = 2$	Untuk membagi operan di sebelah kiri menggunakan operan di sebelah kanan
Sisa Bagi %	$11 \% 2 = 1$	Mendapatkan sisa pembagian dari operan di sebelah kiri operator ketika dibagi oleh operan di sebelah kanan
Pangkat **	$8 ** 2 = 64$	Memangkatkan operan disebelah kiri operator dengan operan di sebelah kanan operator
Pembagian Bulat //	$10 // 3 = 3$	Sama seperti pembagian. Hanya saja angka dibelakang koma dihilangkan

Dibawah ini adalah contoh penggunaan operator aritmatika pada Python:

```
#OPERATOR ARITMATIKA

#Penjumlahan
print(13 + 2)
# Output: 15

#Pengurangan
hutang = 10000
bayar = 5000
sisahutang = hutang - bayar
print("Sisa hutang Anda adalah:", sisahutang)
# Output: Sisa hutang Anda adalah: 5000

#Perkalian
panjang = 15
lebar = 8
luas = panjang * lebar
print(luas)
# Output: 120

#Pembagian
kue = 16
anak = 4
kuePerAnak = kue / anak
print("Setiap anak akan mendapatkan bagian kue sebanyak:", kuePerAnak)
# Output: Setiap anak akan mendapatkan bagian kue sebanyak: 4.0

#Sisa Bagi / Modulus
bilangan1 = 14
bilangan2 = 5
hasil = bilangan1 % bilangan2
print("Sisa bagi dari bilangan ", bilangan1, " dan ", bilangan2, " adalah ", hasil)
# Output: Sisa bagi dari bilangan 14 dan 5 adalah 4

#Pangkat
bilangan3 = 8
bilangan4 = 2
hasilPangkat = bilangan3 ** bilangan4
print(hasilPangkat)
# Output: 64

#Pembagian Bulat
print(10//3)
# Output: 3
```

2. Operator Perbandingan

Operator perbandingan (comparison operators) digunakan untuk membandingkan suatu nilai dari masing-masing operan.

Operator	Contoh	Penjelasan
Sama dengan <code>==</code>	<code>1 == 1</code>	bernilai True Jika masing-masing operan memiliki nilai yang sama, maka kondisi bernilai benar atau True.
Tidak sama dengan <code>!=</code>	<code>2 != 2</code>	bernilai False Akan menghasilkan nilai kebalikan dari kondisi sebenarnya.
Tidak sama dengan <code><></code>	<code>2 <> 2</code>	bernilai False Akan menghasilkan nilai kebalikan dari kondisi sebenarnya.
Lebih besar dari <code>></code>	<code>5 > 3</code>	bernilai True Jika nilai operan kiri lebih besar dari nilai operan kanan, maka kondisi menjadi benar.
Lebih kecil dari <code><</code>	<code>5 < 3</code>	bernilai True Jika nilai operan kiri lebih kecil dari nilai operan kanan, maka kondisi menjadi benar.
Lebih besar atau sama dengan <code>>=</code>	<code>5 >= 3</code>	bernilai True Jika nilai operan kiri lebih besar dari nilai operan kanan, atau sama, maka kondisi menjadi benar.
Lebih kecil atau sama dengan <code><=</code>	<code>5 <= 3</code>	bernilai True Jika nilai operan kiri lebih kecil dari nilai operan kanan, atau sama, maka kondisi menjadi benar.

Dibawah ini adalah contoh penggunaan operator perbandingan pada Python:

```
# SAMA DENGAN
print(1 == 1) # Hasilnya akan bernilai True karena satu sama dengan satu
print(1 == 2) # Hasilnya akan bernilai False karena satu tidak sama dengan dua

# TIDAK SAMA DENGAN
print(2 != 2) # Hasilnya akan bernilai False karena dua seharusnya sama dengan dua
print(2 != 3) # Hasilnya akan bernilai True karena dua tidak sama dengan tiga

# LEBIH BESAR DARI
print(5 > 3) # Hasilnya akan bernilai True karena lima lebih besar dari tiga

# LEBIH KECIL DARI
print(5 < 3) # Hasilnya akan bernilai False karena lima tidak lebih besar dari tiga

# LEBIH BESAR DARI SAMA DENGAN
print(5 >= 3) # Hasilnya akan bernilai True karena lima lebih besar dari sama dengan tiga

# LEBIH KECIL DARI SAMA DENGAN
print(5 <= 3) # Hasilnya akan bernilai False karena lima tidak lebih besar dari sama dengan tiga
```

3. Operator Penugasan

Operator penugasan digunakan untuk memberikan atau memodifikasi nilai ke dalam sebuah variabel.

Operator	Contoh	Penjelasan
Sama dengan <code>=</code>	<code>a = 1</code>	Memberikan nilai di kanan ke dalam variabel yang berada di sebelah kiri.
Tambah sama dengan <code>+=</code>	<code>a += 2</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri ditambah dengan nilai di sebelah kanan.
Kurang sama dengan <code>-=</code>	<code>a -= 2</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dikurangi dengan nilai di sebelah kanan.
Kali sama dengan <code>*=</code>	<code>a *= 2</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dikali dengan nilai di sebelah kanan.
Bagi sama dengan <code>/=</code>	<code>a /= 4</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dibagi dengan nilai di sebelah kanan.
Sisa bagi sama dengan <code>%=</code>	<code>a %= 3</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dibagi dengan nilai di sebelah kanan. Yang diambil nantinya adalah sisa baginya.
Pangkat sama dengan <code>**=</code>	<code>a **= 3</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dipangkatkan dengan nilai di sebelah kanan.
Pembagian bulat sama dengan <code>//=</code>	<code>a //= 3</code>	Membagi bulat operan sebelah kiri operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri.

Dibawah ini adalah contoh penggunaan operator penugasan pada Python:

```
a = 2
a += a
print(a)
# Output: 4

a = 3
a *= a
print(a)
# Output: 9
```

4. Operator Logika

Operator logika digunakan untuk menggabungkan beberapa kondisi. Operator ini termasuk AND, OR, dan NOT. Mereka mengembalikan nilai boolean berdasarkan kombinasi kondisi yang diberikan.

Operator	Operation	Contoh	Deskripsi
AND	Benar dan benar maka benar, benar dan salah maka salah	$2 < 5$ and $2 < 10$	Kedua pernyataan harus benar untuk mendapatkan hasil benar
OR	Benar atau salah maka benar, salah atau salah maka salah	$2 < 5$ or $3 < 1$	Hanya perlu salah satu pernyataan benar untuk mendapatkan hasil benar
NOT	Tidak (benar dan benar) maka salah, tidak (benar atau salah) maka salah	$\text{not}(3 < 5 \text{ or } 4 < 3)$	Membalikan sebuah hasil dari pernyataan

Dibawah ini adalah contoh penggunaan operator logika pada Python:

```
x = 2
print(x<1 or x<10) # Jika salah satu kondisi terpenuhi maka akan bernilai True
# Output: True

x = 2
print(x<1 and x<10) # Jika kedua kondisi terpenuhi maka akan bernilai True
# Output: False

x = 2
print(not(x<1 and x<10)) # Di Negasikan
# Output: True
```

5. Operator Bitwise

Operator bitwise bekerja pada level biner, memanipulasi bit individu dalam bilangan. Contohnya termasuk operator AND, OR, XOR, NOT, serta operator pergeseran.

Dibawah ini adalah contoh penggunaan operator bitwise pada Python:

```
# Menetapkan nilai pada variabel
a = 4 # biner: 0100
b = 5 # biner: 0101

# Menampilkan nilai awal variabel
print("Angka pertama:", a)
print("Angka kedua:", b)

# Operator AND
hasil_and = a & b
print(hasil_and, bin(hasil_and))

# Operator OR
hasil_or = a | b
print(hasil_or, bin(hasil_or))

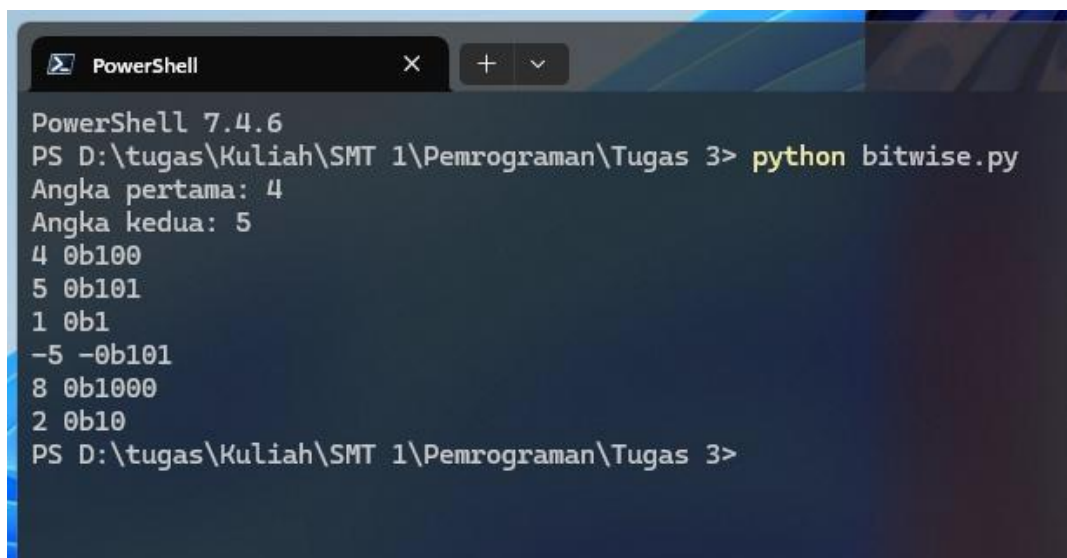
# Operator XOR
hasil_xor = a ^ b
print(hasil_xor, bin(hasil_xor))

# Operator NOT untuk a
hasil_not_a = ~a
print(hasil_not_a, bin(hasil_not_a))

# Operator Left Shift pada a
shift_left = a << 1
print(shift_left, bin(shift_left))

# Operator Right Shift pada a
shift_right = a >> 1
print(shift_right, bin(shift_right))
```

Output:



```
PowerShell 7.4.6
PS D:\tugas\Kuliah\SMT 1\Pemrograman\Tugas 3> python bitwise.py
Angka pertama: 4
Angka kedua: 5
4 0b100
5 0b101
1 0b1
-5 -0b101
8 0b1000
2 0b10
PS D:\tugas\Kuliah\SMT 1\Pemrograman\Tugas 3>
```

6. Operator Keanggotaan

Operator keanggotaan digunakan untuk memeriksa apakah suatu nilai atau variabel ada di dalam suatu koleksi (seperti daftar, tuple, atau string). Contohnya adalah `in` dan `not in`.

Operator ini dapat mencari apakah sebuah nilai termasuk di dalam satu deret data tertentu. Hasilnya adalah benar jika nilai tersebut merupakan bagian dari deret data tersebut, dan sebaliknya jika bukan merupakan bagian dari deret itu maka hasilnya salah.

Operator	Operation	Contoh	Deskripsi
IN	Apakah variabel a merupakan bagian dari variabel b	<code>a in b</code>	Hasilnya benar jika a merupakan bagian dari b
NOT IN	Apakah variabel a bukan merupakan bagian dari variabel b	<code>a not in b</code>	Hasilnya benar jika a bukan merupakan bagian dari b

Dibawah ini adalah contoh penggunaan operator keanggotaan pada Python:

```
a = 5
b = [1,4,5,6,7]
print( a in b )
# Output: True

a = 10
b = [1,4,5,6,7]
print( a in b )
# Output: False
```

7. Operator Identitas

Operator identitas digunakan untuk membandingkan objek. Operator ini termasuk `is` dan `is not`, yang memeriksa apakah dua referensi objek mengarah ke objek yang sama di memori.

Operator identitas secara garis besarnya adalah membandingkan apakah operand sebelah kiri dan kanan adalah sama dengan, objek yang sama, atau menggunakan lokasi memori yang sama.

Operator	Operation	Contoh	Deskripsi
IS	Apakah variabel a dan variabel b merupakan objek yang sama	a is b	Hasilnya benar jika a dan b merupakan objek yang sama
IS NOT	Apakah variabel a dan variabel bukan merupakan objek yang sama	a is not b	Hasilnya benar jika a dan b bukan merupakan objek yang sama

Dibawah ini adalah contoh penggunaan operator identitas pada Python:

```
a = 2
b = 3
print(a is b)
# Output: False

a = 2
b = 2
print(a is b)
# Output: True

a = 2
b = 3
print(a is not b)
# Output: True
```