

# DASAR PYTHON

## BAB 4 : OPERATORS

---

### 4.1 Tujuan

Mahasiswa mengenal Operators pada bahasa pemrograman python

### 4.2 Ulasan Materi

#### A. Operators

##### 1. Operators

Operator digunakan untuk melakukan operasi pada variabel dan nilai.

Pada contoh di bawah ini, kami menggunakan operator + untuk menjumlahkan dua nilai :

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

Output :

```
15
```

Python membagi operator dalam grup berikut :

- a. Operator Aritmatika (Arithmetic operators)
- b. Operator Penugasan (Assignment operators)
- c. Operator Perbandingan (Comparison operators)
- d. Operator Logika (Logical operators)
- e. Operator Identitas (Identity operators)
- f. Operator Keanggotaan (Membership operators)
- g. Operator Bitwise (Bitwise operators)

##### 2. Operator Aritmatika (Arithmetic operators)

Operator aritmatika digunakan dengan nilai numerik untuk melakukan operasi matematika umum :

- a. + : Addition (Tambahan)  
Contoh :  $x + y$
- b. - : Subtraction (Pengurangan)  
Contoh :  $x - y$
- c. \* : Multiplication (Perkalian)  
Contoh :  $x * y$
- d. / : Division (Pembagian)  
Contoh :  $x / y$
- e. % : Modulus  
Contoh :  $x \% y$
- f. \*\* : Exponentiation (Eksponen)  
Contoh :  $x ** y$
- g. // : Floor division  
Contoh :  $x // y$

### 3. Operator Penugasan (Assignment operators)

Operator penugasan digunakan untuk menetapkan nilai ke variable :

- a. =  
Contoh :  $x = 5$  sama dengan  $x = 5$
- b. +=  
Contoh :  $x += 3$  sama dengan  $x = x + 3$
- c. -=  
Contoh :  $x -= 3$  sama dengan  $x = x - 3$
- d. \*=  
Contoh :  $x *= 3$  sama dengan  $x = x * 3$
- e. /=  
Contoh :  $x /= 3$  sama dengan  $x = x / 3$
- f. %=  
Contoh :  $x \% = 3$  sama dengan  $x = x \% 3$
- g. //=

- Contoh :  $x \text{ /= } 3$  sama dengan  $x = x \text{ // } 3$
- h.  $**=$   
 Contoh :  $x **= 3$  sama dengan  $x = x ** 3$
- i.  $\&=$   
 Contoh :  $x \&= 3$  sama dengan  $x = x \& 3$
- j.  $|=$   
 Contoh :  $x |= 3$  sama dengan  $x = x | 3$
- k.  $\wedge=$   
 Contoh :  $x \wedge= 3$  sama dengan  $x = x \wedge 3$
- l.  $>>=$   
 Contoh :  $x >>= 3$  sama dengan  $x = x >> 3$
- m.  $<<=$   
 Contoh :  $x <<= 3$  sama dengan  $x = x << 3$

#### 4. Operator Perbandingan (Comparison operators)

Operator perbandingan digunakan untuk membandingkan dua nilai :

- a.  $==$  : Equal  
 Contoh :  $x == y$
- b.  $!=$  : Not equal  
 Contoh :  $x != y$
- c.  $>$  : Greater than  
 Contoh :  $x > y$
- d.  $<$  : Less than  
 Contoh :  $x < y$
- e.  $>=$  : Greater than or equal to  
 Contoh :  $x >= y$
- f.  $<=$  : Less than or equal to  
 Contoh :  $x <= y$

#### 5. Operator Logika (Logical operators)

Operator logika digunakan untuk menggabungkan pernyataan kondisional :

- a. **and** : Mengembalikan True jika kedua pernyataan benar  
 Contoh:  $x < 5$  and  $x < 10$

- b. `or` : Mengembalikan True jika salah satu pernyataan benar  
Contoh: `x < 5 or x < 4`
- c. `not` : Membalikkan hasilnya, mengembalikan False jika hasilnya benar  
Contoh: `not(x < 5 and x < 10)`

## 6. Operator Identitas (Identity operators)

Operator identitas digunakan untuk membandingkan objek, bukan jika mereka sama, tetapi jika mereka sebenarnya adalah objek yang sama, dengan lokasi memori yang sama :

- a. `is` : Mengembalikan True jika kedua variabel adalah objek yang sama  
Contoh: `x is y`
- b. `is not` : Mengembalikan True jika kedua variabel bukan objek yang sama  
Contoh: `x is not y`

## 7. Operator Keanggotaan (Membership operators)

Operator keanggotaan digunakan untuk menguji apakah urutan disajikan dalam suatu objek :

- a. `in` : Mengembalikan True jika urutan dengan nilai yang ditentukan ada di objek  
Contoh: `x in y`
- b. `not in` : Mengembalikan True jika urutan dengan nilai yang ditentukan tidak ada dalam objek  
Contoh: `x not in y`

## 8. Operator Bitwise (Bitwise operators)

Operator bitwise digunakan untuk membandingkan angka (biner) :

- a. `&` : AND  
Set setiap bit ke 1 jika kedua bit adalah 1
- b. `|` : OR  
Set setiap bit ke 1 jika salah satu dari dua bit adalah 1
- c. `^` : XOR  
Setel setiap bit menjadi 1 jika hanya satu dari dua bit yang bernilai 1
- d. `~` : NOT  
Membalikkan atau *inverts* semua bit

- e. << : Zero fill left shift

Geser ke kiri dengan menekan angka nol dari kanan dan biarkan bit paling kiri jatuh

- f. >> : Signed right shift

Geser ke kanan dengan mendorong salinan bit paling kiri ke dalam dari kiri, dan biarkan bit paling kanan jatuh

### 4.3 Praktikum

#### Contoh : Operator Aritmatika Python

Pada contoh ini kita akan belajar mengenai operator aritmatika dalam bahasa pemrograman python. Operator Aritmatika digunakan sebagai pengoperasian dalam matematika seperti penjumlahan, pengurangan, perkalian, sampai dengan pembagian. Untuk merealisasikan pada operator aritmatika, siapkan lima variabel dengan bertipe integer yaitu a, b, c, d, dan e. Tambahkan masing-masing nilai seperti pada langkah-langkah dibawah ini.

1. Buatlah variabel dengan nama c yang bernilai integer 5

```
c = 5
```

2. Berikutnya buat variabel dengan nama d yang memiliki nilai integer 3

```
d = 3
```

3. Tuliskan kode yang mendefinisikan variabel dengan nama e yang memiliki nilai integer 12

```
e = 12
```

4. Selanjutnya buat fungsi dengan nama addition yang memiliki parameter variabel c dan variabel d

```
def addition(c, d):
```

5. Didalam fungsi addition tambahkan perintah return dan jumlahkan variabel c dan variabel d menggunakan operator aritmatika +

```
return c+d
```

6. Tampilkan hasil penjumlahan dari variabel c dan d menggunakan fungsi print()

```
print(addition(c, d))
```

7. Selanjutnya implementasi operator aritmatika pengurangan, buatlah fungsi subtraction yang memiliki parameter variabel c dan d

```
def subtraction(c, d):
```

8. Didalam fungsi subtraction tambahkan perintah return dan buat sintaks untuk pengurangan variabel c dan variabel d menggunakan operator aritmatika -

```
    return c-d
```

9. Tampilkan hasil pengurangan dari variabel c dan d menggunakan fungsi print()

```
print(subtraction(c, d))
```

10. Selanjutnya implementasi operator aritmatika perkalian, buatlah fungsi multiplication yang memiliki parameter variabel c dan d

```
def multiplication(c, d):
```

11. Didalam fungsi multiplication tambahkan perintah return dan buat sintaks untuk perkalian variabel c dan variabel d menggunakan operator aritmatika \*

```
    return c*d
```

12. Tampilkan hasil perkalian dari variabel c dan d menggunakan fungsi print()

```
print(multiplication(c, d))
```

13. Selanjutnya implementasi operator aritmatika pembagian, buatlah fungsi division yang memiliki parameter variabel e dan d

```
def division(e, d):
```

14. Didalam fungsi division tambahkan perintah return dan buat sintaks untuk pembagian variabel e dan variabel d menggunakan operator aritmatika /

```
    return e/d
```

15. Tampilkan hasil pembagian dari variabel e dan d

```
print(division(e, d))
```

16. Berikut full source code dari contoh operator aritmatika python

```
c = 5
d = 3
e = 12

def addition(c, d):
    return c+d

print(addition(c, d))
```

```
def subtraction(c, d):  
    return c-d  
  
print(subtraction(c, d))  
  
def multiplication(c, d):  
    return c*d  
  
print(multiplication(c, d))  
  
def division(e, d):  
    return e/d  
  
print(division(e, d))
```

### **Percobaan 1 : Operator Aritmatika Python**

Sekarang kerjakan percobaan 1 dengan mengikuti langkah-langkah dibawah ini !

1. Buatlah variabel ***angka1*** yang memiliki nilai 20
2. Buatlah variabel ***angka2*** yang memiliki nilai 3
3. Selanjutnya buat fungsi dengan nama ***perkalian*** yang memiliki parameter variabel angka1 dan angka2
4. Berikan sintaks untuk perkalian variabel angka1 dan angka2 dengan memanfaatkan operator aritmatika \*
5. Tampilkan hasil perkalian menggunakan fungsi print()
6. Jalankan kode program percobaan 1 diatas dengan menekan tombol Check Code Validity