

## BAB IV

### OPERATOR

#### 4.1 Tujuan Praktikum

- Mahasiswa dapat memahami jenis operator.
- Mahasiswa mampu memahami cara menggunakan operator.
- Mahasiswa dapat membuat program sederhana menggunakan operator.

#### 4.2 Alokasi Waktu

1 x Pertemuan = 50 Menit

#### 4.3 Dasar Teori

##### 4.3.1 Pengantar Operator

Operator adalah sebuah simbol yang digunakan untuk melakukan sebuah operasi pada satu atau lebih operand. Sedangkan operand adalah sebuah variabel atau sebuah nilai yang digunakan dalam sebuah operasi. Contoh  $5 + 3$  (simbol  $+$  disebut sebagai operator sedangkan 5 dan 3 disebut sebagai operand). Beberapa operator yang terdapat pada bahasa pemrograman python adalah:

- a) Operator Aritmatika (*Arithmetic Operators*).
- b) Operator Perbandingan (*Comparison / Relational Operators*).
- c) Operator Penugasan (*Assignment Operators*).
- d) Operator Logika (*Logical Operators*).
- e) Operator Keanggotaan (*Membership Operators*).
- f) Operator Identitas (*Identity Operators*).

##### 4.3.2 Jenis-Jenis Operator

- a) Operator Aritmatika (*Arithmetic Operators*).

Operator aritmatika adalah operator umum yang biasa digunakan untuk melakukan perhitungan aritmatika. Operator ini terdiri dari:

*Tabel 4. Jenis-jenis operator aritmatika*

Operator	Simbol	Contoh	Keterangan
Penjumlahan	+	$2 + 5 = 7$	Menjumlahkan nilai dari

			masing-masing operand
Pengurangan	-	$5 - 2 = 3$	Mengurangi nilai operand sebelah kiri dengan nilai operand sebelah kanan
Perkalian	*	$2 * 5 = 10$	Mengalikan nilai operand
Pembagian	/	$10 / 5 = 2$	Membagi nilai <i>operand</i> di sebelah kiri dengan nilai <i>operand</i> di sebelah kanan
Sisa Bagi	%	$11 \% 2 = 1$	Sisa bagi dari nilai <i>operand</i> di sebelah kiri ketika dibagi dengan nilai <i>operand</i> di sebelah kanan
Pangkat	**	$8 ** 2 = 64$	Memangkatkan nilai <i>operand</i> sebelah kiri dengan nilai <i>operand</i> di sebelah kanan
Pembagian Bulat	//	$10 // 3 = 3$	Seperti halnya operator pembagian tetapi dibulatkan ke bawah atau angka di belakang koma dihilangkan

b) Operator Perbandingan (*Comparison / Relational Operators*).

Perbandingan adalah operator yang digunakan untuk membandingkan nilai dari masing-masing *operand*. Hasil dari operasi menggunakan operator perbandingan hanya ada 2 yaitu *True* dan *False*. Operator ini terdiri dari:

Tabel 5. Jenis-jenis operator perbandingan

Operator	Simbol	Contoh	Keterangan
Kurang dari	<	$5 < 2$ (False)	Bernilai <i>False</i> karena nilai <i>operand</i> di sebelah kiri lebih besar dari nilai <i>operand</i> di sebelah kanan sedangkan operator yang digunakan adalah <
Lebih dari	>	$5 > 2$ (True)	Bernilai <i>True</i> karena nilai <i>operand</i> di sebelah kiri lebih

			besar dari nilai <i>operand</i> di sebelah kanan dan operator yang digunakan adalah >
Kurang dari atau sama dengan	<=	2 <= 5 (True)	Bernilai <i>True</i> karena nilai <i>operand</i> di sebelah kiri kurang dari <b>atau</b> sama dengan nilai <i>operand</i> di sebelah kanan dan operator yang digunakan <=
Lebih dari atau sama dengan	>=	5 >= 10 (False)	Bernilai <i>False</i> karena nilai <i>operand</i> di sebelah kiri lebih kecil dari <b>atau</b> sama dengan nilai <i>operand</i> di sebelah kanan dan operator yang digunakan >=
Sama dengan	==	11 == 2 (False)	Bernilai <i>False</i> karena nilai <i>operand</i> di sebelah kiri tidak sama dengan nilai <i>operand</i> di sebelah kanan dan operator yang digunakan adalah ==
Tidak sama dengan	!=	8 != 2 (True)	Bernilai <i>True</i> karena nilai <i>operand</i> di sebelah kiri tidak sama dengan nilai <i>operand</i> di sebelah kanan dan operator yang digunakan adalah !=

c) Operator Penugasan (*Assignment Operators*).

Operator penugasan adalah operator yang digunakan untuk memberikan atau memodifikasi sebuah nilai dalam sebuah variabel. Operator ini terdiri dari:

Tabel 6. Jenis-jenis operator penugasan

Operator	Simbol	Contoh	Keterangan
Sama dengan	=	$a = 5$	Memberikan nilai <b>5</b> pada variabel <b>a</b>
Tambah sama dengan	+=	$a += 5$	Memberikan nilai kepada variabel <b>a</b> dengan nilai variabel itu sendiri ditambah dengan nilai <b>5</b> atau ( $a = a + 5$ )
Kurang sama dengan	-=	$a -= 5$	Memberikan nilai kepada variabel <b>a</b> dengan nilai variabel itu sendiri dikurang dengan nilai <b>5</b> atau ( $a = a - 5$ )
Kali sama dengan	*=	$a *= 10$	Memberikan nilai kepada variabel <b>a</b> dengan nilai variabel itu sendiri dikali dengan nilai <b>10</b> atau ( $a = a * 10$ )
Bagi sama dengan	/=	$a /= 2$	Memberikan nilai kepada variabel <b>a</b> dengan nilai variabel itu sendiri dibagi dengan nilai <b>2</b> atau ( $a = a / 2$ )
Sisa bagi sama dengan	%=	$a \% = 2$	Memberikan nilai kepada variabel <b>a</b> dengan nilai variabel itu sendiri dibagi dengan nilai <b>2</b> atau ( $a = a \% 2$ ) tetapi hasil yang ditampilkan adalah sisa bagi
Pangkat sama dengan	**=	$a ** = 3$	Memberikan nilai kepada variabel <b>a</b> dengan nilai variabel itu sendiri dipangkatkan dengan nilai <b>3</b> atau ( $a = a ** 3$ )
Pembagian bulat sama dengan	//=	$a //= 3$	Memberikan nilai kepada variabel <b>a</b> dengan nilai variabel itu sendiri dibagi dengan nilai <b>3</b> atau ( $a = a // 3$ ) tetapi hasil yang ditampilkan adalah nilai pembagian yang dibulatkan

d) Operator Logika (*Logical Operators*).

Operator Logika digunakan untuk membandingkan dua Operand atau dua nilai yang bertipe Boolean dan akan menghasilkan nilai Boolean yaitu TRUE atau FALSE. Berikut beberapa jenis operatornya:

Tabel 7. Jenis-jenis operator logika

Operator	Contoh	Keterangan
and	$7 > 2$ and $2 > 3$ (False)	Bernilai <i>True</i> karena kedua <i>operand</i> bernilai <i>True</i> . $7 > 2 = \text{True}$ , $2 > 1 = \text{True}$ . Jika salah satu atau kedua <i>operand</i> tersebut bernilai <i>False</i> maka hasilnya <i>False</i> .
or	$7 < 2$ or $2 > 1$ (False)	Bernilai <i>True</i> karena salah satu <i>operand</i> bernilai <i>True</i> . $7 < 2 = \text{False}$ , $2 > 1 = \text{True}$ . Jika salah satu atau kedua <i>operand</i> tersebut bernilai <i>True</i> maka hasilnya <i>True</i> . Hasil akan bernilai <i>False</i> ketika kedua <i>operand</i> bernilai <i>False</i> .
not	$a = \text{not True}$ (False)	Bernilai <i>False</i> karena nilai dari <i>operand</i> bernilai <i>True</i> . Jika nilai <i>operand</i> awalnya bernilai <i>False</i> maka hasilnya akan bernilai <i>True</i> , begitupun sebaliknya.

e) Operator Keanggotaan (*Membership Operators*).

Operator keanggotaan adalah operator yang bisa digunakan untuk memeriksa apakah ada sebuah nilai yang menjadi bagian keanggotaan dari sebuah *sequence*. *Sequence* bisa terdiri dari sebuah *list*, *string* atau *tuple*. Hasil dari operasi menggunakan operator keanggotaan hanya ada 2 yaitu *True* dan *False*. Operator ini terdiri dari:

Tabel 8. Jenis-jenis operator keanggotaan

Operator	Contoh	Keterangan
in	list_dat a = [1,2,3,4, 5] 5 in list_dat a (True)	Dilakukan pemeriksaan terhadap sebuah <i>list</i> apakah ada nilai 5 di dalam <i>list</i> dengan nama variabel list_data. Karena nilai 5 terdapat dalam <i>list</i> tersebut, hasil yang ditampilkan adalah <i>True</i>
not in	list_dat a = [1,2,3,4, 5,8] 8 not in list_dat a (False)	Dilakukan pemeriksaan terhadap sebuah <i>list</i> apakah nilai 5 tidak ada di dalam <i>list</i> dengan nama variabel list_data. Karena nilai 5 ada di dalam <i>list</i> tersebut, hasil yang ditampilkan adalah <i>False</i>

f) Operator Identitas (*Identity Operators*).

Operator identitas adalah operator yang digunakan untuk memeriksa identitas dari nilai apakah memiliki identitas memori yang sama atau tidak. Hasil dari operasi menggunakan operator keanggotaan hanya ada 2 yaitu *True* dan *False*. Operator ini terdiri dari:

Tabel 9. Jenis-jenis operator identitas

Operator	Contoh	Keterangan
is	a = 10 b = 910 a is b (False)	Dilakukan pemeriksaan apakah nilai <i>operand</i> di sebelah kiri memiliki identitas memori yang sama dengan nilai <i>operand</i> yang berada di sebelah kanan. Jika sama maka akan bernilai <i>True</i> , jika tidak akan bernilai <i>False</i> .

is not	<pre>a = 10 b = 10 a is not b (True)</pre>	Dilakukan pemeriksaan apakah nilai <i>operand</i> di sebelah kiri memiliki identitas memori yang berbeda dengan nilai <i>operand</i> yang berada di sebelah kanan. Jika sama maka akan bernilai <i>True</i> , jika tidak akan bernilai <i>False</i> .
--------	--	---

## 4.4 Latihan Praktikum

### 4.4.1 Operator Aritmatika.

```
✓ [1] x = 9
    y = 2

    print("x + y :", x+y)    #Penjumlahan
    print("x - y :", x-y)    #Pengurangan
    print("x * y :", x*y)    #Perkalian
    print("x / y :", x/y)    #Pembagian (Menghasilkan Bilangan Pecahan)
    print("x // y :", x//y)  #Pembagian (Menghasilkan Bilangan Bulat)
    print("x % y :", x%y)    #Modulus (Sisa Pembagian)
    print("x ** y :", x**y)  #Perpangkatan.

x + y : 11
x - y : 7
x * y : 18
x / y : 4.5
x // y : 4
x % y : 1
x ** y : 81
```

Gambar 39. Coding operator aritmatika

### 4.4.2 Operator Perbandingan (Comparison / Relational Operators).

```
▼ Operator Perbandingan.

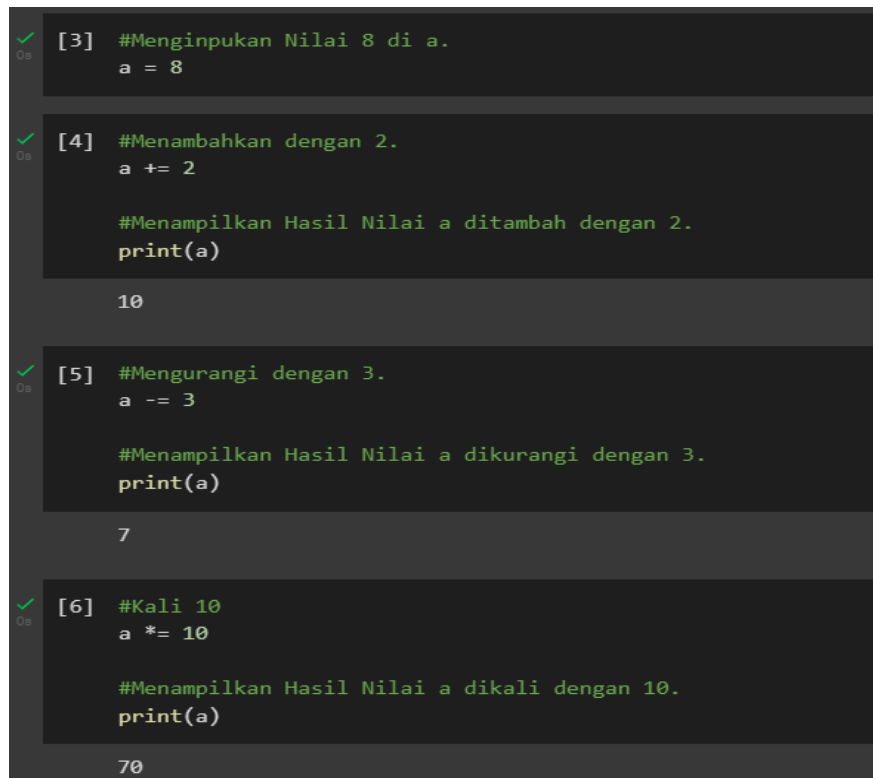
✓ [2] x = 5
    y = 4

    print("x > y :", x>y)    #Lebih Besar
    print("x < y :", x<y)    #Lebih Kecil
    print("x == y :", x==y)  #Sama Dengan
    print("x != y :", x!=y)  #Tidak Sama Dengan
    print("x >= y :", x>=y)  #Lebih Besar Sama Dengan
    print("x <= y :", x<=y)  #Lebih Kecil Sama Dengan

x > y : True
x < y : False
x == y : False
x != y : True
x >= y : True
x <= y : False
```

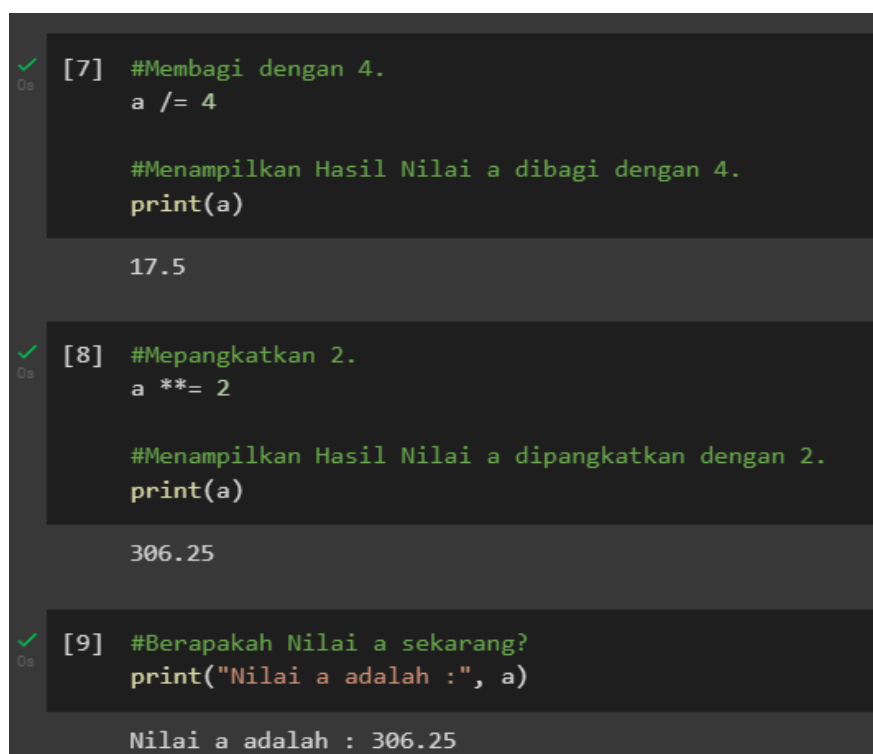
Gambar 40. Coding operator perbandingan

#### 4.4.3 Operator Penugasan (Assignment Operators).



```
[3] #Menginputkan Nilai 8 di a.  
a = 8  
  
[4] #Menambahkan dengan 2.  
a += 2  
  
#Menampilkan Hasil Nilai a ditambah dengan 2.  
print(a)  
  
10  
  
[5] #Mengurangi dengan 3.  
a -= 3  
  
#Menampilkan Hasil Nilai a dikurangi dengan 3.  
print(a)  
  
7  
  
[6] #Kali 10  
a *= 10  
  
#Menampilkan Hasil Nilai a dikali dengan 10.  
print(a)  
  
70
```

Gambar 41. Coding operator penugasan 1



```
[7] #Membagi dengan 4.  
a /= 4  
  
#Menampilkan Hasil Nilai a dibagi dengan 4.  
print(a)  
  
17.5  
  
[8] #Mepangkatkan 2.  
a **= 2  
  
#Menampilkan Hasil Nilai a dipangkatkan dengan 2.  
print(a)  
  
306.25  
  
[9] #Berapakah Nilai a sekarang?  
print("Nilai a adalah :", a)  
  
Nilai a adalah : 306.25
```

Gambar 42. Coding operator penugasan 2



#### 4.4.4 Operator Logika (Logical Operators).

```
✓ [10] x = True
    y = False

    #Logika And.
    z = x and y
    print("True and False :", z)

    #Logika Or.
    z = x or y
    print("True or False :", z)

    #Logika Not.
    z = not x
    print("Not True      :", z)

True and False : False
True or False  : True
Not True       : False
```

Gambar 43. Coding operator logika

#### 4.4.5 Operator Keanggotaan (Membership Operators).

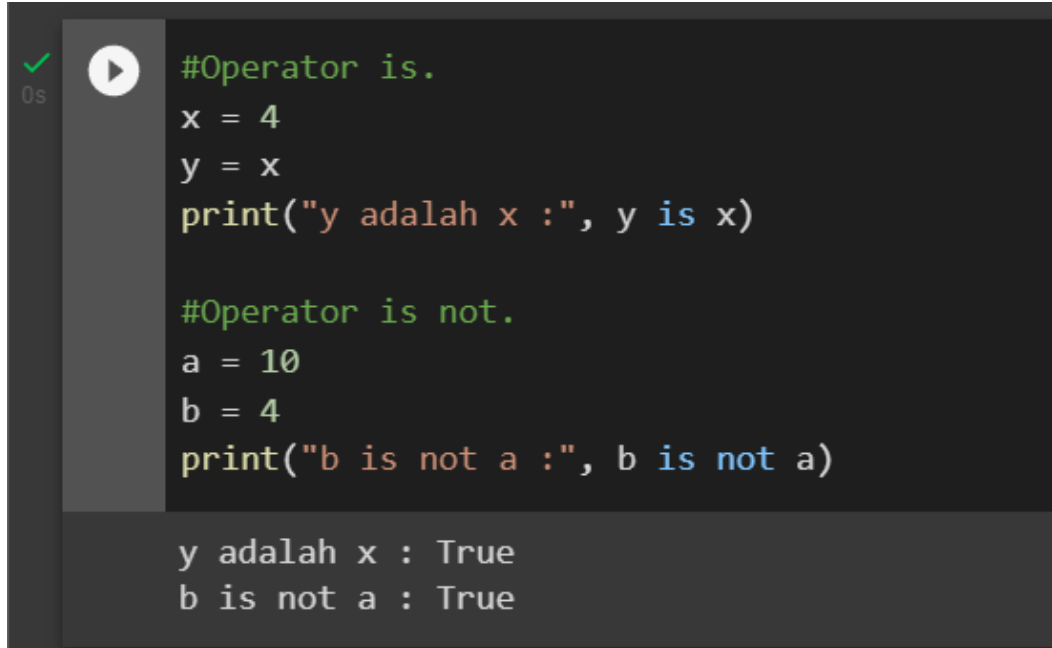
```
✓ [11] #Operator in.
    x = [1,5,9]
    y = 5
    print("y in x   :", y in x)

    #Operator not in.
    x = [1,5,9]
    y = 5
    print("y not in :", y not in x)

y in x   : True
y not in : False
```

Gambar 44. Coding operator keanggotaan

#### 4.4.6 Identitas (Identity Operators).



```
#Operator is.
x = 4
y = x
print("y adalah x :", y is x)

#Operator is not.
a = 10
b = 4
print("b is not a :", b is not a)
```

y adalah x : True  
b is not a : True

Gambar 45. Coding operator identitas

#### 4.5 Tugas.

1. Jika nilai a adalah 9 dan b adalah 5, buatlah kodingan yang menjawab pertanyaan berikut:
  - g) Apakah a sama dengan b?
  - h) Apakah  $a < b$ ?
  - i) Apakah  $a > b$ ?
  - j) Apakah  $a \leq b$ ?
  - k) Apakah  $a \geq b$ ?
  - l) Apakah  $a \neq b$ ?
2. Jika nilai a adalah 5 dan b adalah 6, buat kodingan untuk menjawab pertanyaan berikut:
  - $a + b = ?$
  - $b - a = ?$
  - $a * b = ?$
  - $a / b = ?$