

3. Mengenal Tipe Data dan Operator

Tipe Data di Python

Variabel menyimpan data yang dilakukan selama program dieksekusi dan isinya dapat diubah oleh operasi – operasi tertentu pada program yang menggunakan variabel tersebut.

Di dalam Python, terdapat beberapa tipe data yang cukup unik bila dibandingkan dengan bahasa pemrograman seperti C, Java, dan yang lainnya. Tipe data pada Python adalah sebagai berikut :

- Boolean, contoh True and False
- Complex, pasangan angka real dan imajiner, misalnya $1 + 5j$
- Date, bilangan yang dapat dikonversi menjadi format tanggal, misalnya 26-09-2013
- Float, bilangan yang mempunyai koma, misalnya 3.14, 6.387
- Hexadecimal, bilangan dalam format heksa, misalnya 7b, 4d2
- Integer, bilangan bulat, misalnya 10, 20, 30, 15, 37
- Long, bilangan bulat yang panjang, misal 123456789123456789L
- None, data yang tidak terdefinisi tipe data apapun
- String, data yang berisi kalimat. Bisa dibentuk dengan diapit tanda ' dan ', atau diapit “ dan “, atau diapit “""" dan """ untuk membentuk paragraf.
- List, sebuah data berupa untaian yang menyimpan berbagai tipe data dan isinya bisa diubah. Lebih lengkapnya akan dibahas di bab 6.
- Tuple, sebuah data berupa untaian yang menyimpan berbagai tipe data tapi isinya tidak bisa diubah. Lebih lengkapnya akan dibahas di bab 6.
- Dictionary, sebuah data berupa untaian yang menyimpan berbagai tipe data berupa pasangan penunjuk dan nilai. Lebih lengkapnya akan dibahas di bab 6.
- Objek, sebuah data yang berisi atribut dan method. Lebih lengkapnya akan dibahas di bab 10

Operator – Operator di Python

Selain variabel diatas, terdapat juga beberapa operator untuk pemrosesan data di Python. Agar lebih memahami seperti apa cara kerja operator di python, bukalah console **python** dan coba contoh disamping penjelasan tiap operator. Berikut operator yang ada di Python :

1. Aritmatika (Arithmetic Operator)

Op er ato r	Penjelasan	Contoh
+	Penjumlahan, menambahkan dua buah operan	a, b = 10, 5 hasil = a + b

Op er ato r	Penjelasan	Contoh
		# hasil akan 15 print "hasil : ", hasil
-	Pengurangan, mengurangi operan disebelahkiri operator dengan operan di sebelah kanan operator	a, b = 10, 8 hasil = a - b # hasil akan 2 print "hasil : ", hasil
*	Perkalian, mengalikan operan di sebelah kiri dengan operan di sebelah kanan operator	a, b = 3, 5 hasil = a * b # hasil akan 15 print "hasil : ", hasil
/	Pembagian, membagi operan di sebelah kiri dengan operan disebelah kanan operator	a, b = 4, 2 hasil = a / b # hasil akan 2 print "hasil : ", hasil
%	Modulus, mendapatkan sisa pembagian dari operan di sebelah kiri operator ketika dibagi oleh operan di sebelah kanan	a, b = 11, 2 hasil = a % b # hasil akan 1 print "hasil : ", hasil
**	Pemangkatan, memangkatkan operan disebelah kiri operator dengan operan di sebelah kanan operator	a, b = 11, 2 hasil = a ** b # hasil akan 121 print "hasil : ", hasil
//	Pembagian bulat, prosesnya sama seperti pembagian. Hanya saja angka dibelakang koma dihilangkan	a, b = 11, 2 hasil = a // b # hasil akan 5 print "hasil : ", hasil

2. Perbandingan (Comparison Operator)

Op er ato r	Penjelasan	Contoh
=	Memeriksa apakah kedua nilai (operan) yang dibandingkan sama atau tidak. Jika sama akan dikembalikan nilai True jika tidak sama akan dikembalikan nilai False .	a, b = 10, 10 # hasil akan True print "hasil : ", a == b

Op er ato r	Penjelasan	Contoh
!=	Memeriksa apakah kedua nilai yang dibandingkan sama atau tidak. Jika tidak sama akan dikembalikan nilai True jika sama akan dikembalikan nilai False .	<pre> a, b = 10, 8 # hasil akan True print "hasil : ", a != b c = 10 # hasil akan False print "hasil : ", a != c </pre>
< >	Fungsinya sama dengan operator !=	<pre> a, b = 10, 8 # hasil akan True print "hasil : ", a <> b c = 10 # hasil akan False print "hasil : ", a <> c </pre>
>	Memeriksa apakah nilai di sebelah kiri operator lebih besar dari nilai di sebelah kanan operator	<pre> a, b = 4, 2 # hasil akan True print "hasil : ", a > b </pre>
<	Memeriksa apakah nilai di sebelah kiri operator lebih kecil dari nilai di sebelah kanan operator	<pre> a, b = 2, 4 # hasil akan True print "hasil : ", a < b </pre>
>= =	Memeriksa apakah nilai di sebelah kiri operator lebih besar dari nilai di sebelah kanan operator atau memiliki nilai yang sama	<pre> a, b = 4, 2 c = 4 # hasil akan True print "hasil : ", a >= b # hasil akan True print "hasil : ", a >= c # hasil akan False print "hasil : ", b >= a </pre>
<= =	Memeriksa apakah nilai di sebelah kiri operator lebih kecil dari nilai di sebelah kanan operator atau memiliki nilai yang sama	<pre> a, b = 4, 2 c = 4 # hasil akan False print "hasil : ", a <= b # hasil akan True print "hasil : ", a <= c # hasil akan True </pre>

Op er ato r	Penjelasan	Contoh
		print "hasil : ", b <= a

3. Penugasan (Assignment Operator)

Op er ato r	Penjelasan	Contoh
=	Mengisikan nilai di sebelah kanan operator ke nilai di sebelah kiri operator	a = 10 # hasil akan 10 print a b = 15 # hasil akan 15 print b
+=	Menambahkan operan sebelah kiri operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri	a, b,= 10, 8 # hasil akan 18 sma dgn a = a + b a += b print "hasil : ", a
-=	Mengurangi operan sebelah kiri operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri	a, b,= 10, 8 # hasil akan 2 sma dgn a = a - b a -= b print "hasil : ", a
*=	Mengalikan operan sebelah kiri operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri	a, b,= 10, 8 # hasil akan 80 sma dgn a = a * b a *= b print "hasil : ", a
/=	Membagi operan sebelah kiri operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri	a, b,= 10, 5 # hasil akan 2 sma dgn a = a / b a /= b print "hasil : ", a
%=	Mengambil sisa bagi dari operan sebelah kiri operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri	a, b,= 10, 4 # hasil akan 2 sma dgn a = a % b a %= b print "hasil : ", a
** =	Memangkatkan operan sebelah kiri	a, b,= 10, 2

Op er ato r	Penjelasan	Contoh
	operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri	# hasil akan 100 sma dgn a = a ** b a **= b print "hasil : ", a
// =	Membagi bulat operan sebelah kiri operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri	a, b,= 10, 4 # hasil akan 2 sma dgn a = a // b a //= b print "hasil : ", a

4. Biner (Bitwiser Operator)

Op er ato r	Penjelasan	Contoh
&	Operator biner AND, memeriksa apakah operan di sebelah kiri dan operan sebelah kanan mempunyai angka biner 1 di setiap bit. Jika keduanya bernilai 1 maka bit hasil operasi akan bernilai 1	a, b = 13, 37 # a akan bernilai '0000 1101' # b akan bernilai '0010 0101' c = a & b # c akan bernilai 5 = '0000 0101' print c
	Operator biner OR, memeriksa apakah operan di sebelah kiri dan operan sebelah kanan mempunyai angka biner 1 di setiap bit. Jika salah satunya bernilai 1 maka bit hasil operasi akan bernilai 1	a, b = 13, 37 # a akan bernilai '0000 1101' # b akan bernilai '0010 0101' c = a b # c akan bernilai 45 = '0010 1101' print c
^	Operator biner XOR, memeriksa apakah operan di sebelah kiri dan operan sebelah kanan mempunyai angka biner 1 di setiap bit. Jika keduanya bernilai 1 maka bit hasil operasi akan bernilai 0	a, b = 13, 37 # a akan bernilai '0000 1101' # b akan bernilai '0010 0101' c = a ^ b # c akan bernilai 40 = '0010 1000' print c
~	Operator biner Negative, membalik nilai bit. Misal dari 1 menjadi 0, dari 0 menjadi 1	a, b = 13, 37 # a akan bernilai '0000 1101'

Op er ato r	Penjelasan	Contoh
		# b akan bernilai '0010 0101'
	Operator penggeser biner ke kiri, deret bit akan digeser ke kiri sebanyak n kali	a, b = 13, 37
< <		# a akan bernilai '0000 1101' # b akan bernilai '0010 0101' # hasil bernilai 52 = "0011 0100" print a << 2 # hasil bernilai 148 = '1001 0100' print b << 2
	Operator penggeser biner ke kanan, deret bit akan digeser ke kanan sebanyak satu kali	a, b = 13, 37 # a akan bernilai '0000 1101' # b akan bernilai '0010 0101'
> >		# hasil bernilai 3 = '0000 0011' print a >> 2 # hasil bernilai 9 = '0000 1001' print b >> 2

5. Logika (Logical Operator)

Op er ato r	Penjelasan	Contoh
and	Jika kedua operan bernilai True, maka kondisi akan bernilai True. Selain kondisi tadi maka akan bernilai False	a, b = True, True # hasil akan True print a and b
or	Jika salah satu atau kedua operan bernilai True maka kondisi akan bernilai True. Jika keduanya False maka kondisi akan bernilai False	a, b = True, False # hasil akan True print a or b print b or a print a or a # hasil akan False print b or b
not	Membalikkan nilai kebenaran pada operan,	a, b = True, False

Op er ato r	Penjelasan	Contoh
	misal jika asalnya True akan menjadi False dan begitupun sebaliknya	# hasil akan True print not a print not b

6. Keanggotaan (Membership Operator)

Op er ato r	Penjelasan	Contoh
in	Memeriksa apakah nilai yang dicari berada pada list atau struktur data python lainnya. Jika nilai tersebut ada maka kondisi akan bernilai True	sebuah_list = [1, 2, 3,4 ,5] print 5 in sebuah_list

7. Identitas (Identity Operator)

n o t i n	Memeriksa apakah nilai yang dicari tidak ada pada list atau struktur data python lainnya. Jika nilai tersebut tidak ada maka kondisi akan bernilai True	sebuah_list = [1, 2, 3,4 ,5] print 10 not in sebuah_list
-----------------------	---	---

Op er ato r	Penjelasan	Contoh
is	Memeriksa apakah nilai di sebelah kiri operan memiliki identitas memori yang sama dengan nilai di sebelah kanan operan. Jika sama maka kondisi bernilai True	a, b = 10, 10 # hasil akan True print a is b
i s n o t	Memeriksa apakah nilai di sebelah kiri operan memiliki identitas memori yang berbeda dengan nilai di sebelah kanan operan. Jika berbeda maka kondisi bernilai True	a, b = 10, 5 # hasil akan True print a is not b

Prioritas Eksekusi Operator di Python

Dari sekian banyaknya operator yang telah disebutkan, masing – masing mempunyai prioritas pemrosesan yang dapat dilihat pada tabel berikut. Prioritas tersebut makin ke bawah makin akhir untuk dieksekusi. Paling atas adalah yang akan didahulukan daripada operator lain, sedangkan paling bawah adalah operator yang paling terakhir dieksekusi :

Operator	Keterangan
**	Aritmatika
~, +, -	Bitwise
*, /, %, //	Aritmatika
+, -	Aritmatika
>>, <<	Bitwise
&	Bitwise
^,	Bitwise
<=, <, >, >=	Perbandingan
<>, ==, !=	Perbandingan
=, %=, /=, //, Penugasan -=, +=, *=, **=	
is, is not	identitas
in, not in	membership
not, or, and	logika