



---

# Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2025/2026

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

NIM	71251221
Nama Lengkap	RIZKY FEBRIANTO KRISTIANSYAH
Minggu ke / Materi	02 / Variable Expression & Statement

PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
YOGYAKARTA  
2026

## BAGIAN 1: Variable Expression & Statement

### Values dan Type

Value adalah nilai dari sebuah variabel. Value dibagi menjadi beberapa tipe data yang berbeda, yaitu: integer (bilangan bulat), float (bilangan pecahan), character (huruf), string (kumpulan char), dan bool (True/False). Untuk mengetahui tipe data apakah sebuah value dari suatu variabel, kita bisa menggunakan fungsi `type()`

```
[4]: bulat = 6
      pecahan = 2.5
      kalimat = "Ini string"
      kompleks = 1+2j

      print("Tipenya adalah: ", type(bulat))
      print("Tipenya adalah: ", type(pecahan))
      print("Tipenya adalah: ", type(kalimat))
      print("Tipenya adalah: ", type(kompleks))

      Tipenya adalah: <class 'int'>
      Tipenya adalah: <class 'float'>
      Tipenya adalah: <class 'str'>
      Tipenya adalah: <class 'complex'>
```

Gambar 1.1: Contoh penggunaan `type()`

### Variabel

Variabel adalah lokasi memori yang dicadangkan untuk menyimpan value. Variabel menyimpan data yang dilakukan selama program dieksekusi, yang nantinya value dari variabel tersebut dapat diubah oleh operasi-operasi tertentu pada program yang menggunakan variabel.

Variabel dapat menyimpan berbagai macam tipe data. Di dalam pemrograman Python, variabel mempunyai sifat yang dinamis, artinya variabel di Python tidak perlu dideklarasikan tipe data tertentu dan tipe data dari variabel Python dapat diubah saat program dijalankan.

```
pesan = 'Selamat Datang!'
n = 17
ipk = 3.5
```

Gambar 1.2: Contoh variabel beserta nilainya

Potongan kode tersebut merupakan contoh dari variabel. Variabel pertama bernama `pesan` yang berisi string, variabel kedua bernama `n` yang berisi nilai integer 17 dan variabel

ketiga bernama ipk yang berisi nilai float. Untuk menampilkan nilai dari variabel, dapat menggunakan perintah print()

```
print(pesan)
print(n)
print(ipk)
```

```
Selamat Datang!
17
3.5
```

Gambar 1.3: Output perintah print()

## Variabel

Pemberian nama pada variabel mengacu pada aturan berikut:

1. Nama variabel boleh diawali menggunakan huruf atau garis bawah (\_), contoh: nama, \_pajak, keluargaZhang, pewaris\_heavenlydemon
2. Karakter selanjutnya dapat berupa huruf, garis bawah, atau angka, contoh: n2, nilai\_1
3. Karakter pada nama variabel bersifat case sensitive. Sistem mengasumsikan perbedaan huruf besar dan kecil sebagai variabel yang berbeda. Contoh: kriptoScam dan kriptoscam adalah dua variabel yang berbeda
4. Nama variabel tidak boleh menggunakan keyword yang sudah ada dalam python.

Keyword yang tidak boleh digunakan sebagai nama variabel antara lain:

<b>and</b>	<b>del</b>	<b>from</b>	<b>None</b>	<b>True</b>
<b>as</b>	<b>elif</b>	<b>global</b>	<b>nonlocaly</b>	<b>try</b>
<b>assert</b>	<b>else</b>	<b>if</b>	<b>not</b>	<b>while</b>
<b>break</b>	<b>except</b>	<b>import</b>	<b>or</b>	<b>width</b>
<b>class</b>	<b>False</b>	<b>in</b>	<b>pass</b>	<b>yield</b>
<b>continue</b>	<b>finally</b>	<b>is</b>	<b>raise</b>	<b>async</b>
<b>def</b>	<b>for</b>	<b>lamda</b>	<b>return</b>	<b>wait</b>

Gambar 1.4: Keyword yang tidak boleh digunakan sebagai nama variabel

## Statements

Statements merupakan bagian dari code interpreter Python yang dapat dieksekusi. Misalnya pada statement print, dapat berupa expression statements dan assignment.

Ketika menggunakan Python dalam mode interaktif, interpreter secara langsung akan melakukan eksekusi dan menampilkan hasilnya. Hal ini tentu saja berbeda ketika menggunakan script mode. Script biasanya berisi statements yang saling berhubungan secara sekuensial.

## Operator dan Operand

Operator adalah simbol tertentu yang digunakan untuk melakukan operasi aritmatika maupun logika. Nilai yang padanya dilakukan operasi disebut sebagai operand. Misalnya adalah  $2 + 3$ . Dalam kasus ini, tanda  $+$  adalah operator penjumlahan, dan angka 2 dan 3 adalah operand. Tabel berikut menunjukkan jenis operator aritmatika pada Python.

Operator	Nama dan Fungsi	Contoh
$+$	Penjumlahan, menjumlahkan 2 buah operand	$x + y$
$-$	Pengurangan, mengurangi 2 buah operand	$x - y$
$*$	Perkalian, mengalikan 2 buah operand	$x * y$
$/$	Pembagian, membagi 2 buah operand	$x / y$
$**$	Pemangkatan, memangkatkan bilangan	$x ** y$

Tabel 2.1: Operator pada Python

## Expressions

Expression merupakan representasi dari nilai dan dapat terdiri dari gabungan antara value, variabel, dan operator. Value dengan sendirinya dapat dianggap sebagai expression dan juga variabel. Secara umum, semuanya dapat disebut dengan expression.

## Urutan Operasi

Urutan operasi berlaku bila ada lebih dari satu operator dalam satu expression. Urutan operasi bergantung pada aturan prioritas. Untuk operasi aritmatika, Python mengikuti konversi matematika. Urutan operasi sering disingkat dengan **PEMDAS – Parentheses, Exponentiation, Multiplication, and Division Operator**.

1. Paranthese (Tanda kurung) – Merupakan prioritas tertinggi dan digunakan untuk memaksa expression dalam urutan yang sesuai. Contohnya  $2*(3-1)$  hasilnya 4 karena  $(3-1)$  akan dikerjakan terlebih dahulu, dan  $(1+1)**(5-2)$  hasilnya  $2**3$  yaitu 8. Penggunaan tanda kurung juga dapat digunakan untuk membuat expression menjadi lebih mudah untuk dibaca, misalnya gaji – (pajak / 100)

2. Exponentiation (Eksponensial/Perpangkatan) – Merupakan prioritas tertinggi kedua. Contohnya  $2^{**}1+1$  hasilnya 3 bukan 4 karena diproses  $2^{**}1$  dulu kemudian ditambah 1, dan  $3*1^{**}3$  hasilnya 3 bukan 27 dengan konsep yang sama.
3. Multiplication and Division (Perkalian dan Pembagian) – Memiliki prioritas di bawah eksponen dengan prioritas yang sama tetapi lebih tinggi dari penjumlahan dan pengurangan. Penjumlahan dan pengurangan juga memiliki prioritas yang sama pula. Contoh  $2*3-1$  hasilnya 5 bukan 4, dan  $6+4/2$  hasilnya 8 bukan 5.
4. Operators – operator memiliki prioritas yang sama, dibaca dari kiri ke kanan. Contoh  $5-3-1$  hasilnya 1 bukan 3 karena operasi pengurangan  $5-3$  terlebih dahulu baru kemudian hasilnya dikurangi dengan 1.

## Operator Modulus dan String

### Modulus

Operator Modulus merupakan sisa hasil bagi dari bilangan pertama dengan bilangan kedua. Operator ini hanya berlaku pada tipe data integer. Dalam Python, operator modulus dilambangkan dengan tanda persen (%)

```
modulus = 96%50
print(modulus)
```

46

Gambar 1.5: Contoh penggunaan modulus

Contoh kegunaan operator modulus adalah untuk memeriksa apakah satu angka dapat dibagi dengan yang, misalnya jika  $x \% y$  adalah 0 maka  $x$  dapat dibagi oleh  $y$ , dan untuk mengekstrak digit paling kanan dari suatu angka, misalnya  $x \% 10$  menghasilkan digit  $x$  paling kanan (dalam basis 10).

### String

Operator  $+$  ketika bekerja dengan string tidak berarti penjumlahan secara matematika, melainkan penggabungan antar string. Operator  $*$  juga bekerja dengan string dengan melakukan perkalian antara content string dengan integer. Contoh:

```
abad = '19'
tahun = '45'
seruan = 'MERDEKA! '
print(abad+tahun, seruan * 3)
```

```
1945 MERDEKA! MERDEKA! MERDEKA!
```

Gambar 1.6: Contoh penggunaan operator string

## Menangani Input dari Pengguna

Python dapat menangani input dari pengguna menggunakan built-in function yang disebut `input()` untuk mendapatkan input dari keyboard. Ketika fungsi ini dipanggil, program akan berhenti dan menunggu pengguna untuk mengetik sesuatu. Ketika pengguna menekan tombol Enter, program akan dilanjutkan dan input akan mengembalikan apa yang diketik oleh pengguna sebagai tipe data tertentu.

```
celcius = int(input("Suhu berapa derajat Celcius sekarang?\n"))
fahrenheit = celcius * 9 / 5 + 32
print("Konversi suhu Celcius ke Fahrenheit, sekarang suhu", fahrenheit, "derajat Fahrenheit")

Suhu berapa derajat Celcius sekarang?
0
Konversi suhu Celcius ke Fahrenheit, sekarang suhu 32.0 derajat Fahrenheit
```

Gambar 1.7: Penggunaan Input integer untuk konversi suhu

Misalkan program meminta input berbentuk integer, maka bila user memberikan input berupa string, akan terjadi error seperti ini:

```
celcius = int(input("Suhu berapa derajat Celcius sekarang?\n"))
fahrenheit = celcius * 9 / 5 + 32
print("Konversi suhu Celcius ke Fahrenheit, sekarang suhu", fahrenheit, "derajat Fahrenheit")

Suhu berapa derajat Celcius sekarang?
lima belas
```

Fix Code

```
-----
ValueError                                Traceback (most recent call last)
Cell In[17], line 1
----> 1 celcius = int(input("Suhu berapa derajat Celcius sekarang?\n"))
      2 fahrenheit = celcius * 9 / 5 + 32
      3 print("Konversi suhu Celcius ke Fahrenheit, sekarang suhu", fahrenheit, "derajat Fahrenheit")

ValueError: invalid literal for int() with base 10: 'lima belas'
```

Gambar 1.8: Error message kesalahan input tipe data

## Komentar

Tanda pagar (#) digunakan untuk menandai komentar di Python. Komentar tidak akan diproses oleh interpreter Python. Komentar hanya berguna untuk programmer untuk

memudahkan memahami maksud dari kode. Python tidak memiliki fitur komentar multibaris. Kita harus mengomentari satu persatu baris seperti berikut:

```
pesan = "Hello World!" #komentar duls  
#komen komen  
#gomen  
print(pesan) #ga jumatatan bro?  
#trims sdh mengingatkan
```

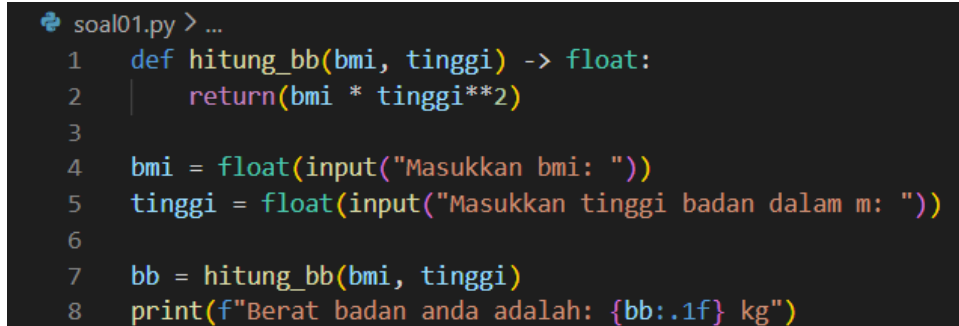
Hello World!

Gambar 1.8: Contoh penggunaan komentar

## BAGIAN 2: LATIHAN MANDIRI

Link Github repository: [https://github.com/ifgodwillsit/71251221\\_febrian](https://github.com/ifgodwillsit/71251221_febrian)

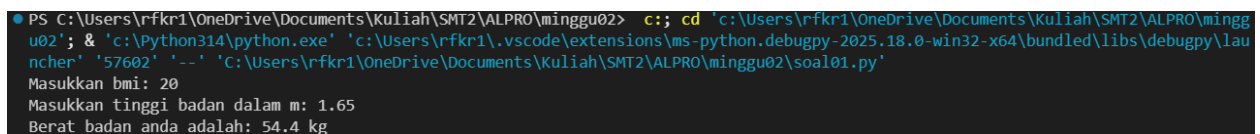
### SOAL 1



```
soal01.py > ...
1 def hitung_bb(bmi, tinggi) -> float:
2     | return(bmi * tinggi**2)
3
4 bmi = float(input("Masukkan bmi: "))
5 tinggi = float(input("Masukkan tinggi badan dalam m: "))
6
7 bb = hitung_bb(bmi, tinggi)
8 print(f"Berat badan anda adalah: {bb:.1f} kg")
```

Gambar 2.1: Source code soal 1

Diketahui user input untuk BMI dan tinggi badan. Variabel BMI disimpan dalam tipe data float. Tinggi badan karena diminta dalam satuan meter, maka disimpan dalam tipe data float juga. Maka untuk mengakomodasi user menginput BMI dan tinggi diketik line 4 dan 5. Diketahui rumus BMI = berat/tinggi<sup>2</sup>, maka untuk mencari bb menggunakan rumus bmi\*tinggi\*\*2. Kemudian untuk menghitung BB, dibuat fungsi hitung\_bb dengan parameter bmi dan tinggi di line 1 yang akan mengembalikan nilai bmi\*tinggi\*\*2. Return nilai ini kemudian diassign sebagai value dari variabel bb atau berat badan di line 7. Untuk mengeluarkan outputnya maka diketik perintah print “Berat badan anda adalah {bb:.1f} kg”, di mana {bb:.1f} berarti memanggil value bb dalam float dengan jumlah desibel sebanyak 1 angka di belakang koma.



```
PS C:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02> c:; cd 'c:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02'; & 'c:\Python314\python.exe' 'c:\Users\rfrkr1\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '57602' '--' 'C:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02\soal01.py'
Masukkan bmi: 20
Masukkan tinggi badan dalam m: 1.65
Berat badan anda adalah: 54.4 kg
```

Gambar 2.2: Pembuktian soal 1



## SOAL 2

```
soal02.py > ...
1 def fx(x):
2     return 2 * x**3 + 2 * x + 15/x
3
4 x = float(input("Input nilai x: "))
5 hasil = fx(x)
6 print(f"Hasil formula f(x)= 2x^3 + 2x + 15/x adalah: {hasil:.2f}")
```

Gambar 2.3: Source code untuk soal 2

Diketahui fungsi eksponen yaitu  $f(x) = 2x^3 + 2x + 15/x$ . Ditanyakan adalah nilai  $f(x)$ . User dapat menginput nilai  $x$  lewat line 4, dengan  $x$  dapat bernilai bilangan pecahan atau bulat (tipe data float). Dibuat fungsi  $fx$  pada line 1 untuk menghitung hasil dari fungsi eksponen, dengan mengembalikan nilai  $2x^3 + 2x + 15/x$ . Variabel  $hasil$  pada line 5 diassign nilai return dari fungsi  $fx$ . Line 6 memberikan output hasil dari fungsi  $fx$ , dengan memanggil variabel  $hasil$  yang berupa float dan dibulatkan menjadi 2 bilangan di belakang koma.

```
PS C:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02> c::; cd 'c:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02'; & 'c:\Python314\python.exe' 'c:\Users\rfrkr1\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '61746' '--' 'C:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02\soal02.py'
Input nilai x: 2
Hasil formula f(x)= 2x^3 + 2x + 15/x adalah: 27.50
```

Gambar 2.4: Pembuktian soal 2

## SOAL 3

```
soal03.py > ...
1 lamaKerja = 5
2 gajiPerjam = int(input("Gaji perjam yang Budi inginkan: Rp. "))
3 jamKerja = int(input("Jam kerja rata-rata Budi perminggu: "))
4
5 gajiKotor = gajiPerjam * jamKerja * lamaKerja
6 print("Gaji Budi selama libur panas sebelum dipotong pajak: Rp.", gajiKotor)
7
8 gajiMinPajak = gajiKotor - (gajiKotor * 14 / 100)
9 print("Gaji Budi selama libur panas setelah dipotong pajak: Rp.", gajiMinPajak)
10
11 clothing = gajiMinPajak * 10 / 100
12 print("Pengeluaran Budi untuk pakaian dan aksesoris: Rp.", clothing)
13
14 stationer = gajiMinPajak * 1 / 100
15 print("Pengeluaran Budi untuk alat tulis: Rp.", stationer)
16
17 gajiMinPengeluaran = gajiMinPajak - clothing - stationer
18 sedekah = gajiMinPengeluaran * 25 / 100
19 print("Uang yang Budi sedekahkan: Rp.", sedekah)
20
21 yatim = sedekah * 30 / 100
22 print("Sedekah yang diterima anak yatim: Rp.", yatim)
23
24 dhuafa = sedekah * 70 / 100
25 print("Sedekah yang diterima dhuafa: Rp.", dhuafa)
```

Gambar 2.5: Source code soal 3

Diketahui Budi bekerja selama 5 minggu, dan user input gaji perjam serta jumlah jam kerja Budi perminggu. Dibuat variabel gajiKotor untuk menghitung total gaji Budi sebelum dipotong apapun dengan rumus  $\text{gajiPerjam} * \text{jamKerja} * \text{lamaKerja}$ .

Pertama-tama, gaji Budi dipotong oleh pajak sebesar 14%. Dibuat variabel gajiMinPajak untuk menghitung hasilnya dengan rumus  $\text{gajiKotor} - (\text{gajiKotor} * 14/100)$ .

Setelah dipotong pajak, gaji Budi dipakai 10% untuk pakaian dan aksesoris serta 1% untuk alat tulis. Maka dibuat variabel clothing untuk menghitung gaji yang dipakai membeli pakaian dan variabel stationer untuk menghitung gaji yang dipakai membeli alat tulis. Kemudian dibuat variabel gajiMinPengeluaran untuk menghitung gaji Budi setelah dipakai membeli pakaian dan aksesoris.

Kemudian diketahui bahwa Budi menyisihkan 25% sisa gajinya untuk disedekahkan ke anak yatim dan kaum dhuafa. Dibuat variabel sedekah untuk menghitung jumlah uang yang Budi sedekahkan. Kemudian diketahui 30% dari uang sedekah itu diberikan kepada anak yatim dan 70% sisanya kepada kaum dhuafa. Oleh sebab itu dibuatlah variabel yatim dan variabel dhuafa untuk menghitung jumlah uang yang disedekahkan kepada setiap kaum tersebut.

```
PS C:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02> c:: cd 'c:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02'; & 'c:\Python314\python.exe' 'c:\Users\rfrkr1\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '64864' '--' 'C:\Users\rfrkr1\OneDrive\Documents\Kuliah\SMT2\ALPRO\minggu02\soal03.py'
Gaji perjam yang Budi inginkan: Rp.15000
Jam kerja rata-rata Budi perminggu: 30
Gaji Budi selama libur panas sebelum dipotong pajak: Rp. 2250000
Gaji Budi selama libur panas setelah dipotong pajak: Rp. 1935000.0
Pengeluaran Budi untuk pakaian dan aksesoris: Rp. 193500.0
Pengeluaran Budi untuk alat tulis: Rp. 19350.0
Uang yang Budi sedekahkan: Rp. 430537.5
Sedekah yang diterima anak yatim: Rp. 129161.25
Sedekah yang diterima dhuafa: Rp. 301376.25
```

Gambar 2.6: Pembuktian soal 3