



---

# 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	71251230
Nama Lengkap	Okky Alexander
Minggu ke / Materi	02 / Variable, Expression dan Statements

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

## MATERI VALUE DAN TYPE

Value adalah komponen dasar program seperti angka atau huruf yang memiliki tipe tertentu, contohnya integer pada `print(4)`, float pada `print(10.876)`, serta string yang menggunakan tanda petik seperti `print('Z')` atau `print('True')`. Setiap nilai ini dapat diidentifikasi tipenya menggunakan fungsi `type()`, di mana `type(5)` akan menghasilkan `<class 'int'>` dan `type(2.0)` menghasilkan `<class 'float'>`.

Berikut contoh kodenya untuk memahami bagaimana Python memproses nilai-nilai tersebut:

```
>>> print(4)
```

Output: 4 (Integer)

```
>>> print(10.876)
```

Output: 10.876 (Float)

```
>>> print('Z')
```

Output: Z (String)

```
>>> x = 1+2j
```

```
>>> print(type(x))
```

Output: `<class 'complex'>`

```
>>> print(1,000,000)
```

Output: 1 0 0 (Dianggap 3 parameter berbeda)

## MATERI VARIABLE

Variable adalah lokasi di memori komputer yang dicadangkan untuk menyimpan nilai selama program dieksekusi, di mana isinya dapat dimanipulasi atau diubah melalui operasi tertentu. Python memiliki sifat dinamis, yang berarti variabel tidak perlu dideklarasikan tipe datanya secara spesifik dan tipenya bisa berubah secara otomatis mengikuti nilai yang dimasukkan. Untuk menampilkan isi dari sebuah variabel, kita cukup menggunakan perintah `print()` diikuti dengan nama variabelnya.

Berikut contoh codenya:

```
>>> n = 17
```

```
>>> print(n)
```

Output : 17

## MATERI NAMA VARIABLE DAN KEYWORDS

Penamaan variabel dalam Python harus mengikuti aturan baku agar tidak terjadi *error*. Nama variabel wajib diawali oleh huruf atau garis bawah (`_`), dan karakter selanjutnya boleh dikombinasikan dengan angka. Penting untuk diingat bahwa variabel bersifat case-sensitive, sehingga perbedaan huruf besar dan kecil (seperti `variabel_Ku` dan `variabel_ku`) akan dianggap sebagai dua variabel yang berbeda. Selain itu, nama variabel dilarang menggunakan keyword atau kata kunci bawaan sistem Python seperti `if`, `while`, atau `for`.

Python sendiri memiliki 35 keyword yang tidak boleh digunakan untuk memberi nama variabel.

<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>nonlocal</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>lambda</b>	<b>return</b>	<b>wait</b>

Gambar 1.3: 35 keyword.

Berikut ini contoh penggunaan variable dalam bahasa pemrograman Python.

```
>>> nama = "Budi"
>>> variabel_Ku = "Data A"
>>> print(variabel_Ku) # Output: Data A
```

## MATERI STATEMENTS

Statements adalah unit instruksi dalam kode Python yang dapat dieksekusi oleh interpreter, mencakup berbagai jenis seperti expression statements (perintah langsung) dan assignment (pemberian nilai). Dalam mode interaktif, interpreter akan langsung mengeksekusi dan menampilkan hasil dari setiap statement, sedangkan dalam script mode, statements biasanya disusun secara berurutan (sekuensial) di dalam sebuah file untuk menjalankan logika program yang lebih kompleks.

```
Minggu 2 > coba.py > ...
1 print(1)
2 x=2
3 print(x)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS E:\Koolyeah\github\PrakAlPro-71251230> python -u "e:\Koolyeah\github\PrakAlPro-71251230\Minggu 2\coba.py"
1
2
PS E:\Koolyeah\github\PrakAlPro-71251230>
```

Gambar 1.2: Contoh statement dan outputnya.

## MATERI OPERATOR DAN OPERAND

Operator adalah simbol khusus yang digunakan untuk melakukan operasi matematika atau logika terhadap nilai yang disebut sebagai operand. Sebagai contoh, pada ekspresi  $2 + 3$ , simbol  $+$  berperan sebagai operator, sementara angka 2 dan 3 adalah operand-nya. Di dalam Python, operator aritmatika memungkinkan kita untuk menjalankan berbagai perhitungan matematis dasar seperti penjumlahan ( $+$ ), pengurangan ( $-$ ), perkalian ( $*$ ), pembagian ( $/$ ), hingga operasi yang lebih spesifik seperti sisa bagi (modulus), perpangkatan (exponentiation), dan pembagian bulat (floor division).

Beberapa contoh penggunaan operator aritmatika pada Python dapat dilihat pada potongan kode berikut ini

```
>>> hour = 5
>>> print (hour-1)
4
```

## MATERI EXPRESSIONS

Expression adalah kombinasi atau gabungan dari *values*, variabel, dan operator yang merepresentasikan suatu nilai tertentu. Bahkan sebuah nilai tunggal seperti 17 atau variabel  $x$  sudah dianggap sebagai expression karena keduanya memiliki nilai. Dalam mode interaktif, Python akan langsung melakukan evaluasi terhadap setiap expression yang dimasukkan dan menampilkan hasilnya secara instan, seperti pada operasi  $1 + 1$  yang menghasilkan 2.

Berikut contoh codenya:

```
>>> 17
17
```

```
>>> 3 + 2
```

```
5
```

## MATERI URUTAN OPERASI

Python mengikuti aturan prioritas matematika global yang dikenal dengan singkatan PEMDAS untuk menentukan urutan evaluasi dalam sebuah expression. Prioritas tertinggi dipegang oleh Parentheses (tanda kurung), yang berfungsi untuk mengelompokkan operasi atau memperjelas alur perhitungan. Tingkat berikutnya adalah Exponentiation (pemangkatan), disusul oleh Multiplication (perkalian) dan Division (pembagian) yang memiliki derajat lebih tinggi daripada penjumlahan dan pengurangan. Apabila terdapat beberapa operator dengan tingkat prioritas yang sama dalam satu baris kode, Python akan mengeksekusinya secara berurutan dari arah kiri ke kanan. Penggunaan tanda kurung sangat disarankan sebagai praktik terbaik untuk menghindari ambiguitas dan memastikan hasil perhitungan tetap akurat sesuai dengan logika yang diinginkan.

## MATERI OPERATOR MODULUS DAN STRING

Modulus adalah operator bersimbol % yang berfungsi untuk mengambil sisa hasil bagi dari dua bilangan bulat. Misalnya,  $7 \% 3$  menghasilkan 1, karena 7 dibagi 3 adalah 2 dengan sisa 1. Operator ini sering digunakan untuk memvalidasi apakah suatu bilangan habis dibagi bilangan lain (hasil 0) atau untuk memisahkan digit angka, seperti  $125 \% 10$  yang menghasilkan angka terakhir yaitu 5.

Dalam Python, operator aritmatika memiliki fungsi berbeda saat digunakan pada String. Operator + berfungsi sebagai konkatenasi atau penggabungan teks, bukan penjumlahan matematis; misalnya '100' + '150' akan menghasilkan '100150'. Sementara itu, operator \* berfungsi sebagai repetisi, di mana sebuah string akan diulang sebanyak nilai integer yang ditentukan, seperti 'Test ' \* 3 yang menghasilkan 'Test Test Test '.

## MATERI MENANGANI INPUT DARI PENGGUNA

Setiap program komputer bekerja berdasarkan siklus IPO (INPUT – PROSES – OUTPUT). Input adalah data mentah yang dimasukkan ke dalam sistem (seperti kartu ATM dan nomor PIN), Proses adalah serangkaian instruksi atau logika yang dilakukan sistem untuk mengolah data tersebut (seperti verifikasi saldo dan penghitungan uang), dan Output adalah hasil akhir yang diberikan kepada pengguna (seperti uang tunai dan struk transaksi). Konsep ini merupakan dasar bagaimana sebuah masalah dipecahkan secara sistematis melalui pemrograman.

Python menggunakan fungsi bawaan `input()` untuk menerima data dari pengguna melalui keyboard, yang secara otomatis akan tersimpan dalam tipe data string. Program akan berhenti sejenak (pause) dan menunggu hingga pengguna menekan tombol Enter sebelum melanjutkan eksekusi. Untuk memandu pengguna, kita bisa menyertakan teks prompt di dalam kurung, serta menggunakan simbol `\n` untuk membuat baris baru agar input diketik di bawah pertanyaan. Jika data yang masuk berupa angka dan ingin dihitung secara matematis, nilai tersebut harus dikonversi terlebih dahulu menggunakan fungsi `int()` atau `float()`. Namun, perlu diperhatikan bahwa proses konversi ini akan menyebabkan `ValueError` jika pengguna memasukkan teks non-angka.

Berikut contoh angkanya:

- Input Dasar: `name = input('Nama: ')` → Jika diisi "Sancaka", maka `name` berisi "Sancaka".
- Input Angka: `suhu = input('Suhu: ')` → Jika diisi "24", maka `int(suhu)` menghasilkan 24 (integer).
- Operasi Matematika: `int("24") + 5` menghasilkan 29.
- Error: `int("Dua puluh")` → Menghasilkan `ValueError` karena teks tidak bisa diubah menjadi angka bulat.

## MATERI KOMENTAR

Komentar adalah catatan yang ditulis di dalam kode untuk membantu programmer memahami maksud instruksi, namun diabaikan sepenuhnya oleh interpreter Python saat dijalankan. Komentar ditandai dengan simbol pagar (`#`); semua teks setelah simbol tersebut pada baris yang sama tidak akan dieksekusi sebagai perintah. Meskipun Python tidak memiliki fitur khusus untuk komentar multibaris (seperti `/* ... */` di bahasa lain), pengembang biasanya menggunakan tanda pagar di setiap baris secara berurutan untuk memberikan penjelasan yang panjang.

Berikut contoh penggunaannya:

- Komentar Baris Baru: `# Menghitung luas lingkaran digunakan di atas baris kode.`
- Komentar Sebaris (Inline): `print("Halo") # Menampilkan salam diletakkan di samping kode.`
- Komentar Multibaris:  
# Langkah 1: Ambil input  
# Langkah 2: Proses data  
# Langkah 3: Tampilkan hasil

Source code github : [https://github.com/tuangkeman/71251230\\_Okky.git](https://github.com/tuangkeman/71251230_Okky.git)

## LATIHAN 2.1

```
tinggi_cm = int(input("Masukkan tinggi badan (cm): "))
bmi_harapan = float(input("Masukkan nilai BMI yang diharapkan: "))
tinggi_m = tinggi_cm / 100
berat_perlu = bmi_harapan * (tinggi_m**2)
print(f"Berat badan yang diperlukan adalah: {berat_perlu:.2f} kg")
```

Gambar 1.3 : Code Program Berat badan yang diperlukan

tinggi\_cm : untuk memasukan input tinggi badan berdasarkan cm

int(input()) agar angka yang diinput pengguna terbaca interger contoh 160 nanti tersimpan 160

bmi\_harapan = untuk memasukan input nilai bmi yang diinginkan

float(input()) agar angka yang diinput terbaca float contoh 22 nanti tersimpan 22.0

tinggi\_m = untuk mengubah cm ke meter karena diperlukan untuk rumus menghitung berat bmi

berat\_perlu = Menghitung berat badan dengan membalik rumus BMI: Berat = BMI \* Tinggi\*\*2.

Print= Menampilkan hasil perhitungan ke layar dengan format 2 angka di belakang koma (:.2f).

## LATIHAN 2.2

```
x = int(input("Masukan nilai x : "))
hasil = (2* (x**3)) + (2 * x) + (15 / x)
print(f"Hasil dari f({x}) adalah: {hasil}")
```

Gambar 1.4 : Code Program menghitung fungsi

x = int(input(...)): Mengambil input teks dari pengguna dan mengubahnya menjadi bilangan bulat (int).

2 \* (x\*\*3): Bagian ini menghitung  $2x^3$ . Dalam Python, pangkat ditulis dengan simbol \*\*.

15 / x: Bagian ini melakukan pembagian.

hasil: Variabel ini menyimpan total penjumlahan dari ketiga komponen fungsi tersebut.

## LATIHAN 2.3

```
gaji_per_jam = float(input("Masukkan gaji per jam yang diinginkan: "))
jam_per_minggu = int(input("Masukkan jumlah jam kerja per minggu: "))
pendapatan_kotor = gaji_per_jam * jam_per_minggu * 5
pajak = 0.14 * pendapatan_kotor
pendapatan_bersih = pendapatan_kotor - pajak
beli_baju = 0.1 * pendapatan_bersih
beli_alat_tulis = 0.01 * pendapatan_bersih
sisu_uang_setelah_belanja = pendapatan_bersih - beli_baju - beli_alat_tulis
total_sedekah = 0.25 * sisu_uang_setelah_belanja
sedekah_anak_yatim = 0.30 * total_sedekah
sedekah_dhuafa = total_sedekah - sedekah_anak_yatim
print("Ringkasan Keuangan Budi")
print(f"1. Pendapatan sebelum pajak: Rp {pendapatan_kotor:,.2f}")
print(f"2. Pendapatan setelah pajak: Rp {pendapatan_bersih:,.2f}")
print(f"3. Pengeluaran pakaian & aksesoris: Rp {beli_baju:,.2f}")
print(f"4. Pengeluaran alat tulis: Rp {beli_alat_tulis:,.2f}")
print(f"5. Total uang yang disedekahkan: Rp {total_sedekah:,.2f}")
print(f"6. Sedekah untuk anak yatim: Rp {sedekah_anak_yatim:,.2f}")
print(f"7. Sedekah untuk kaum dhuafa: Rp {sedekah_dhuafa:,.2f}")
```

Gambar 1.4 : Code Program jumlah uang

Gaji\_per\_jam , jam\_per\_minggu = Mengambil input gaji per jam dan jam kerja seminggu dari pengguna.

pendapatan\_kotor = Menghitung total pendapatan selama 5 minggu (Gaji \* Jam \* 5).

pajak , pendapatan\_bersih = Menghitung pajak 14% dan mengurangnya dari total pendapatan untuk dapat Pendapatan Bersih.

beli\_baju , beli\_alat\_tulis = Menghitung biaya baju (10%) dan alat tulis (1%) dari Pendapatan Bersih.

sisu\_uang\_setelah\_belanja, total\_sedekah = Menghitung sisa uang setelah belanja, lalu mengambil 25% darinya untuk Total Sedekah.

Sedekah\_anak\_yatim, sedekah\_duafa = Membagi Total Sedekah menjadi 30% untuk anak yatim dan sisanya (70%) untuk kaum dhuafa.

Print = Menampilkan semua hasil perhitungan ke layar dengan format mata uang yang rapi.