



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

<b>NIM</b>	<b>71230970</b>
<b>Nama Lengkap</b>	<b>Gregorius Daniel Jodan Perminas</b>
<b>Minggu ke / Materi</b>	<b>02 / Variable, Expression dan Statements</b>

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.

PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
YOGYAKARTA  
2024

## BAGIAN 1: MATERI MINGGU INI (40%)

### Values dan Tipe Data

Value merupakan salah satu komponen utama program, dapat berupa huruf ataupun angka. Contoh value antara lain: 2,3.5,'a','b',"Hello!". Tiap value juga memiliki tipe data berikut contoh tipe data: integer, float, string, complex, dll. Interpreter dapat melakukan identifikasi terhadap string karena pada penulisannya ditutup menggunakan tanda petik (quotation mark).

Berikut ini merupakan cara untuk melihat tipe data pada Python dengan menggunakan fungsi `type()`

```
1 x = 5
2 y = "Hello!"
3 z = 3 + 4j
4 print (x, "tipe datanya adalah ", type(x))
5 print (y, "tipe datanya adalah ", type(y))
6 print (z, "tipe datanya adalah ", type(z))

5 tipe datanya adalah <class 'int'>
Hello! tipe datanya adalah <class 'str'>
(3+4j) tipe datanya adalah <class 'complex'>
```

Gambar 1 - Melihat Tipe Data

### Variabel dan Aturan Penulisan

Variabel merupakan sebuah nama yang digunakan untuk menyimpan suatu nilai dalam memori komputer. Variabel di dalam Python bersifat dinamis dan dapat menyimpan berbagai jenis data. Variabel Python tidak perlu dideklarasikan tipe data tertentu dan Variabel Python dapat diubah saat program dijalankan.

Untuk memberikan nama pada variabel ada beberapa aturan yang perlu dipatuhi:

1. Nama variable boleh diawali menggunakan huruf atau garis bawah (`_`), contoh: `nama`, `_nama`, `namaKu`, `nama_variable`.
2. Karakter selanjutnya dapat berupa huruf, garis bawah (`_`) atau angka, contoh: `_nama`, `n2`, `nilai1`.
3. Karakter pada nama variable bersifat sensitif (case-sensitif). Artinya huruf besar dan kecil dibedakan. Misalnya, `variabel_Ku` dan `variabel_ku`, keduanya dianggap variabel yang berbeda.
4. Nama variabel tidak boleh menggunakan kata kunci yang sudah ada dalam python. Berikut beberapa contohnya:

<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>lamda</b>	<b>return</b>	<b>wait</b>

Gambar 2 - Keyword dalam Python

```

1  umur = 20
2  print(umur)
3  type(umur)
4  umur = "dua puluh"
5  print(umur)
6  type(umur)
7  namaDepan = "Joko"
8  namaBelakang = "Widodo"
9  nama = namaDepan + " " + namaBelakang
10 umur = 22
11 hobi = "Berenang"
12 print("Biodata:\n", nama, "\n", umur, "\n", hobi)

```

```

20
dua puluh
Biodata:
Joko Widodo
22
Berenang

```

*Gambar 3 - Contoh Penggunaan Variabel di Python*

## Statements dan Expressions

Statements merupakan semua baris kode Python yang dapat dieksekusi. Statement tidak akan mengembalikan nilai apapun. Namun, statement akan melakukan suatu proses yang dapat menghasilkan beberapa keadaan.

- if
- if ... else
- while
- do ... while
- for
- for each
- switch

*Gambar 4 - Beberapa Contoh Statement dalam Python*

Sedangkan Expressions merupakan representasi dari nilai dan dapat terdiri dari gabungan antara values, variable dan operator. Dalam konsep pemrograman, expression mengacu pada sesuatu yang dapat dihitung dan menghasilkan nilai.

```

1 + 7
"Invasi" + "Kode"
x * y
True != False
10.0 / 4

```

*Gambar 5 - Beberapa Contoh Expressions dalam Python*

## Operator dan Operand

Operator adalah simbol tertentu yang digunakan untuk melakukan operasi aritmatika maupun logika dalam Python. Nilai yang padanya dilakukan operasi disebut operand. Misalnya pada  $2 + 3$ . Di sini tanda  $+$  adalah operator penjumlahan 2 dan 3 adalah operand.

Berikut ini adalah beberapa operator yang terdapat dalam Python:

Operator	Operation	Contoh	Deskripsi
+	Penjumlahan/Addition	$10 + 2$	Menambahkan 10 dengan 2
-	Pengurangan/Subtraction	$10 - 2$	Mengurangkan 10 dengan 2
*	Pengalian/Multiplication	$10 * 2$	Mengalikan 10 dengan 2
/	Pembagian/Division	$10 / 2$	Membagi 10 dengan 2
%	Sisa Bagi/Modulus	$10 \% 2$	Memberikan sisa pembagian 10 dengan 2
**	Pemangkatan/Exponential	$10 ** 2$	Memangkatkan 10 dengan 2
//	Pembagian Bulat/Floor Division	$10 // 2$	Membagikan 10 dengan 2 dan menghilangkan angka desimal dari hasil pembagian tersebut jika ada

Gambar 6 - Operator dalam Python

Terdapat aturan yang akan diterapkan apabila ada lebih dari satu operator dalam expressions. Urutan operasi sering disingkat dengan BODMAS / PEMDAS.

1. Brackets / Parentheses:  $()$
2. Orders / Exponents:  $**$
3. Division and Multiplication:  $/, *, //, \%$
4. Addition and Subtraction:  $+, -$

Jika kita masih meragukan perhitungan yang dilakukan python maka kita juga dapat menggunakan tanda kurung untuk memastikan bahwa urutan perhitungan dilakukan dengan benar.

## Operator Dalam String

Operator  $+$  ketika bekerja dengan string tidak berarti penjumlahan secara matematika, melainkan penggabungan antar string atau istilahnya adalah "concatenate". Contoh:

```
1 a = 5
2 b = 10
3 c = a + b
4 print(c)
5
6 d = "5"
7 e = "10"
8 f = d + e
9 print(f)
```

```
15
510
```

Gambar 7 - Contoh Operasi Penjumlahan Strings dan Hasilnya

Operator `*` juga dapat digunakan untuk mengalikan integer dengan content dari strings.

```
1 a = "Jokowi Keren!\n"
2 print (a * 3)
```

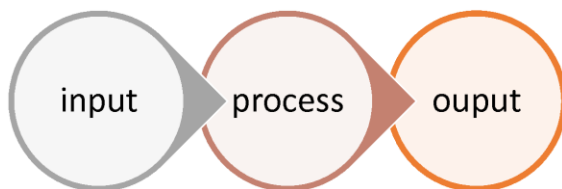
```
Jokowi Keren!
Jokowi Keren!
Jokowi Keren!
```

Gambar 8 - Contoh Operasi Perkalian Strings dan Hasilnya

## Menerima Input dari Pengguna

Pada umumnya alur kerja program adalah input – process – output. Input adalah data/masukan yang diberikan *user* supaya program bisa berjalan. Proses adalah langkah-langkah yang dilakukan oleh program untuk menyelesaikan suatu masalah. Sedangkan Output adalah hasil yang didapatkan setelah menjalankan langkah-langkah tersebut.

### Sistem secara umum



Dalam pemrograman, **Proses** dilakukan oleh Compiler/interpreter  
**Input** → masukan dari user/pengguna  
**Output** → luaran program yang diharapkan pengguna (ditampilkan)



Gambar 9 - Input, Process, Output

Python dapat menerima input dari pengguna berupa text. Dalam hal ini Python sudah menyiapkan sebuah function yang disebut **input**. Ketika function ini dipanggil maka program akan berhenti sejenak dan menunggu masukan dari pengguna, lalu program akan berjalan kembali setelah pengguna menekan enter pada keyboard. Perlu diingat bahwa function **input** akan mengembalikan nilai dalam bentuk strings.

```
1 a = input("Masukkan sebuah angka untuk dijumlahkan: ")
2 print (a+5)
```

Masukkan sebuah angka untuk dijumlahkan: 5

Traceback (most recent call last):

File "d:\UKDW\Semester 2 - 2024\PraktikumAlpro\_A\_Dida\ProgramHere\Laporanmodul2.py", line 2, in <module>

print (a+5)

TypeError: can only concatenate str (not "int") to str

*Gambar 10 - Demonstrasi Input 1*

```
1 a = int(input("Masukkan sebuah angka untuk dijumlahkan: "))
2 print (a+5)
```

Masukkan sebuah angka untuk dijumlahkan: 5

10

*Gambar 11 - Demonstrasi Input 2*

Dari gambar 10 dan 11 kita dapat membuktikan bahwa Input memang memang mengembalikan nilai dalam bentuk strings. Maka pada gambar 10 terjadi error karena Python tidak dapat menjumlahkan integer dan strings. Sedangkan pada gambar 11 program dapat berjalan dengan baik karena sudah di konversi menjadi integer terlebih dahulu baru dijumlahkan dengan angka 5.

## 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.

```
1 # Ini Adalah Program untuk Menerima Input Dari Pengguna
2 a = int(input("Masukkan sebuah angka untuk dijumlahkan: "))
3 print (a+5)
```

*Gambar 12 - Contoh Komentar pada Python*

## BAGIAN 2: LATIHAN MANDIRI (60%)

### SOAL 1

```
# Latihan Mandiri 2.1

# Menurut WHO, kategori standar berat badan ideal pria dan wanita dewasa berdasarkan BMI adalah sebagai berikut:
# Kurang dari 18,5 berarti berat badan kurang (underweight).
# Antara 18,5 - 24,9 berarti berat badan normal
# Antara 25-29,9 berarti berat badan berlebih (overweight).
# Di atas 30 berarti obesitas

print("Selamat datang di Program Pengukuran Berat Badan Ideal!")
print("=====")
nilaiBmiDiharapkan = float(input("Silahkan masukkan nilai BMI yang diharapkan: "))
tinggiBadanCm = int(input("Silahkan masukkan tinggi badan Anda di sini dalam satuan cm: "))

tinggiBadanM = tinggiBadanCm / 100
beratBadan = nilaiBmiDiharapkan * (tinggiBadanM ** 2)

print(f"Dengan nilai BMI: {nilaiBmiDiharapkan} dan tinggi Badan: {tinggiBadanM} m, maka berat badan yang diperlukan yaitu {beratBadan} kg")

Selamat datang di Program Pengukuran Berat Badan Ideal!
=====
Silahkan masukkan nilai BMI yang diharapkan: 21
Silahkan masukkan tinggi badan Anda di sini dalam satuan cm: 165
Dengan nilai BMI: 21.0 dan tinggi Badan: 1.65 m, maka berat badan yang diperlukan yaitu 57.17249999999999 kg
```

Gambar 13 - Program dan Output untuk Latihan 2.1

Di kasus ini yang akan menjadi input pertama adalah nilai BMI Index yang diharapkan dengan menyesuaikan data nilai yang telah disusun oleh WHO pada gambar 13 bagian atas. Kemudian data input kedua adalah tinggi badan seseorang dalam satuan meter, maka pada baris ke-12 data input tinggi akan dibagi 100. Hal ini dilakukan karena pada umumnya orang menghitung tinggi dalam satuan centimeter (hanya mempermudah pengguna). Sebagai catatan tambahan, karena hasil input biasanya berbentuk strings maka pada baris ke 9-10 dilakukan konversi terlebih dahulu agar tipe data menjadi float atau integer supaya operasi aritmatika dapat dilakukan.

Rumus yang diketahui adalah  $BMI = \frac{Berat}{Tinggi^2}$ , namun permintaan soal adalah mencari berat yang diperlukan, maka rumus bisa diubah menjadi  $Berat = BMI \times Tinggi^2$ .

Kemudian setelah perhitungan selesai dilakukan, terdapat statement print agar program menampilkan hasil.

### SOAL 2

```
# Latihan Mandiri 2.2

print("Program Menghitung Hasil Fungsi")
print("=====")

nilaiX = float(input("Silahkan masukkan nilai bilangan real X yang diinginkan: "))
hasilFungsi = 2 * (nilaiX ** 3) + 2 * nilaiX + 15 / nilaiX

print(f"Untuk nilai X: {nilaiX}, maka hasil fungsi: {hasilFungsi}")

Program Menghitung Hasil Fungsi
=====
Silahkan masukkan nilai bilangan real X yang diinginkan: 5
Untuk nilai X: 5.0, maka hasil fungsi: 263.0
```

Dalam kasus ini input yang diperlukan hanya satu yaitu nilai X. Setelah input dimasukkan kemudian dilakukan konversi agar perhitungan aritmatika dapat dilakukan. Fungsi yang diminta oleh soal untuk dihitung adalah  $f(x) = 2 \cdot x^3 + 2 \cdot x + \frac{15}{x}$ . Maka untuk mengubahnya dalam bahasa Python kita bisa mengubah perkalian dengan tanda bintang (\*), pembagian dengan tanda garis miring (/), dan perpangkatan dengan tanda bintang double (\*\*).

## SOAL 3

```
# Latihan Mandiri 2.3

print("Selamat Datang di Program Perhitungan Keuangan")
print("=====")

gajiPerJam = int(input("Silahkan masukkan gaji per jam yang diharapkan: Rp "))
jamKerjaPerMinggu = int(input("Silahkan masukkan lama jam kerja per minggu: "))

pendapatanSebelumPajak = (jamKerjaPerMinggu * 5) * gajiPerJam
pendapatanSesudahPajak = pendapatanSebelumPajak - pendapatanSebelumPajak * 0.14
uangBajuAksesoris = pendapatanSesudahPajak * 0.10
uangAlatTulis = pendapatanSesudahPajak * 0.01
uangSedekah = (pendapatanSesudahPajak - uangBajuAksesoris - uangAlatTulis) * 0.25

uangSedekahLoop = uangSedekah
uangYatim = 0
while uangSedekahLoop >= 1000:
    uangSedekahLoop -= 1000
    uangYatim += 1000 * 0.30

uangKaumDhuafa = uangSedekah - uangYatim

print(f"Pendapatan Budi selama libur musim panas sebelum melakukan pembayaran pajak adalah Rp {pendapatanSebelumPajak}")
print(f"Pendapatan Budi selama libur musim panas setelah melakukan pembayaran pajak adalah Rp {pendapatanSesudahPajak}")
print(f"Jumlah uang yang akan Budi habiskan untuk membeli pakaian dan aksesoris adalah Rp {uangBajuAksesoris}")
print(f"Jumlah uang yang akan Budi habiskan untuk membeli alat tulis adalah Rp {uangAlatTulis}")
print(f"Jumlah uang yang akan Budi sedekahkan adalah Rp {uangSedekah}")
print(f"Jumlah uang yang akan diterima anak yatim adalah Rp {uangYatim}")
print(f"Jumlah uang yang akan diterima kaum dhuafa adalah Rp {uangKaumDhuafa}")

Selamat Datang di Program Perhitungan Keuangan
=====
Silahkan masukkan gaji per jam yang diharapkan: Rp 50000
Silahkan masukkan lama jam kerja per minggu: 36
Pendapatan Budi selama libur musim panas sebelum melakukan pembayaran pajak adalah Rp 9000000
Pendapatan Budi selama libur musim panas setelah melakukan pembayaran pajak adalah Rp 7740000.0
Jumlah uang yang akan Budi habiskan untuk membeli pakaian dan aksesoris adalah Rp 774000.0
Jumlah uang yang akan Budi habiskan untuk membeli alat tulis adalah Rp 77400.0
Jumlah uang yang akan Budi sedekahkan adalah Rp 1722150.0
Jumlah uang yang akan diterima anak yatim adalah Rp 516600.0
Jumlah uang yang akan diterima kaum dhuafa adalah Rp 1205550.0
```

Dalam kasus ini, yang menjadi input adalah gaji per jam yang diinginkan dan berapa lama jam kerja seseorang tiap minggunya. Sama seperti dua kasus sebelumnya, input harus di konversi terlebih dahulu menjadi integer agar operasi aritmatika dapat dilakukan. Kemudian perhitungan mulai dilakukan:

1. **Pendapatan Sebelum Pajak = Jam kerja per minggu x 5** ← *dikali karena bekerja selama 5 minggu x gaji per jam*
2. **Pendapatan Sesudah Pajak = Pendapatan Sebelum Pajak - Pendapatan Sebelum Pajak x 14%** (*pajak penghasilan*) atau bisa juga ditulis **0.14**
3. **Uang Pakaian dan Aksesoris = Pendapatan Sesudah Pajak x 10%** atau bisa ditulis **0.10**
4. **Uang Alat Tulis = Pendapatan Sesudah Pajak x 1%** atau bisa ditulis **0.01**
5. **Uang Sedekah = (Pendapatan Sesudah Pajak – Uang Pakaian dan Aksesoris – Uang Alat Tulis)** ← *Dikurangi terlebih dahulu karena soal meminta bahwa uang sedekah merupakan sisa uang yang dimiliki, kemudian x 25%* atau dapat ditulis **0.25**



6. **Uang yang akan diterima Anak Yatim di sini menggunakan fungsi loop Python dengan perintah `while`.** Di sini dibuat dua variabel baru yaitu **variabel dengan nama `UangSedekahLoop`** supaya tidak mempengaruhi hasil print variabel `UangSedekah` **dan variabel dengan nama `UangYatim`** yang akan dimulai dari 0. Permintaan soal adalah setiap Rp. 1000 uang yang disedekahkan maka 30% akan diserahkan ke anak yatim. Maka kemudian perintah yang diberikan adalah ketika uang masih lebih atau sama dengan Rp. 1000 (*uang dibawah Rp 1000 tidak dikali dengan 30 persen lagi, karena yang dikali 30 persen hanya setiap pemberian Rp. 1000 atau ke atas*) lakukan perulangan di bawah ini:
- Mengurangi `UangSedekahLoop` Rp. 1000
  - Menambah `UangYatim` Rp. 1000 x 30%
7. Kemudian setelah itu soal meminta agar **sisanya uang diserahkan ke kaum dhuafa**. Maka hasil `UangSedekah` tadi dikurangi oleh values yang disimpan variabel `UangYatim` untuk mendapatkan hasil uang yang akan diserahkan ke kaum dhuafa.
- Setelah semua permintaan soal terpenuhi maka statement print diberikan untuk menampilkan hasil perhitungan.