



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71230992
Nama Lengkap	Andriano Kurniawan Ladjeba
Minggu ke / Materi	02 / Variable, Expression dan Statements

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%)

MATERI 1

Values dan type

Value merupakan komponen utama dari program, seperti huruf atau angka. Value yang sering kita kenal 1,2,'a','z' dan "Hello word". Value dibagi menjadi beberapa tipe yang berbeda, misalnya 2 untuk sebuah nilai integer dan "Hello World" untuk sebuah nilai string. Interpreter dapat melakukan indentifikasi terhadap string karena pada penulisannya ditutup menggunakan tanda petik (quation mark).

untuk mencoba memahami values dan type, silahkan coba beberapa baris kode menggunakan python interactive mode seperti yang ditunjukkan pada potongan kode dibawah ini.

```
>>> print(7)
```

```
7
```

```
>>> print(17.54)
```

```
17.54
```

```
>>> print("lalilulelo")
```

```
lalilulelo
```

```
>>> print('True')
```

```
True
```

```
>>> print('False')
```

```
False
```

Perintah print juga bekerja untuk mengeksekusi perintah yang sudah dimasukkan, selain untuk value string perintah print juga bekerja untuk value seperti interger (bilangan bulat), float (bilangan pecahan), Character (huruf), *bool* (benar/salah). Untuk mencobanya kita akan mencoba menggunakan tipe-tipe di atas dengan perintah Python yang menjalankan interpreter. Python juga menyediakan fungsi built-in untuk mengubah atau mendefinisikan tipe data pada value dengan menggunakan fungsi `type()`

```
>>> x = 10
```

```
>>> print(x, "tipenya adalah",type(x))
```

```
10 tipenya adalah <class 'int'>
```

```
>>> print(x, "tipenya adalah", type(x))
```

```
10 tipenya adalah <class 'int'>
```

```
>>> x = 5+5j
```

```
>>> print(x, "tipenya adalah", type(x))
```

```
(5+5j) tipenya adalah <class 'complex'>
```

Ketika menggunakan bilangan bulat besar, beberapa model penulisan menggunakan tanda koma(,) diantara kelompok tiga digit. Misalnya pada penulisan 1.000.000. Dalam python, akan dianggap sebagai bulat.

```
>>> print(1,000,000)
1 0 0
```

Hal ini terjadi dikarenakan Python menganggap bahwa 1,000,000,000 merupakan kiriman parameter sebanyak 3 parameter pada fungsi print, yaitu 1,0,dan 0.

Variabel

Salah satu fitur powerfull dalam

```
>>> pesan = 'Selamatt pagii, mari belajar python'
```

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

```
>>> pi = 3.1415926535897931
```

Variabel dapat menyimpan berbagai jenis data. Dalam pemrograman Python, variabel memiliki sifat dinamis, artinya variablePython tidak perlu dideklarasikan tipe data tertentu dan variable Python dapat diubah saat program dijalankan.

Potongan kode di atas adalah contoh penggunaan variabel. Contoh pertama adalah variabel pesan yang berisi string, contoh kedua adalah variabel n yang berisi nilai integer 17, dan contoh ketiga adalah nilai pi (π).

Untuk menampilkan nilai suatu variabel, Anda dapat menggunakan perintah print .

```
>>> print(n)
17
>>> print(ipk)
3.29
```

Nama Variabel dan Keywords

Pemberian nama pada variabel mengacu pada panduan berikut ini.

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 adalah variabel yang berbeda.
4. Nama variabel tidak boleh menggunakan kata kunci yang sudah ada dalam python seperti if, while, for, dsb.

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

and	del	from	none	True
as	elif	global	nonlocaly	Try
assert	else	if	not	While
break	except	import	or	Width
class	false	in	pass	yield
continue	finally	is	raise	Async
def	for	lamda	return	wait

Berikut ini contoh penggunaan variable dalam bahasa pemrograman Python.

#proses memasukan data kedalam variabel

```
Nama = "Agung Sejagat"
print>Nama)
umur = 20
print(umur)
type(umur)
umur ="dua puluh satu"
print(umur)
type(umur)
namaDepan = "Joko"
namaBelakang = "Widodo"
nama = namaDepan + " " + namaBelakang
umur = 22
hobi "Berenang"
print("Biodata\n",nama,"\n", umur, "\n",hobi)
```

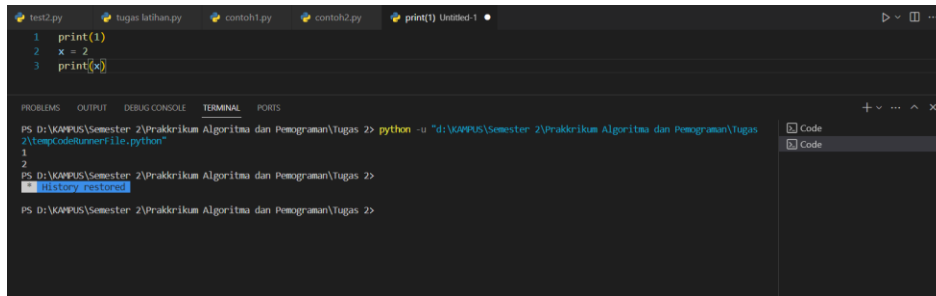
#contoh variabel lainnya

```
Inivariabel = "Halo"
Ini_juga-variabel = "Hai"
_inivariabeljuga = "Hi"
inivariabel1222 = "Bye"
panjang = 10
lebar = 5
luas = panjang * lebar
print(luas)
```

Statements

Ini adalah suatu instruksi dari komputer untuk memberi tahu apa yang dilakukan. Statements juga bagian dari code interpreter Python yang dapat dieksekusi. Jika pada mode interaktif akan langsung memunculkan hasilnya. Berbeda halnya dengan script mode.

Contoh Statements dan hasilnya :



```
1 print(1)
2 x = 2
3 print(x)
```

```
PS D:\KAMPUS\Semester 2\Praktikum Algoritma dan Pemrograman\tugas 2> python -u "d:\KAMPUS\Semester 2\Praktikum Algoritma dan Pemrograman\tugas 2\tempCodeRunnerFile.py"
1
2
PS D:\KAMPUS\Semester 2\Praktikum Algoritma dan Pemrograman\tugas 2>
```

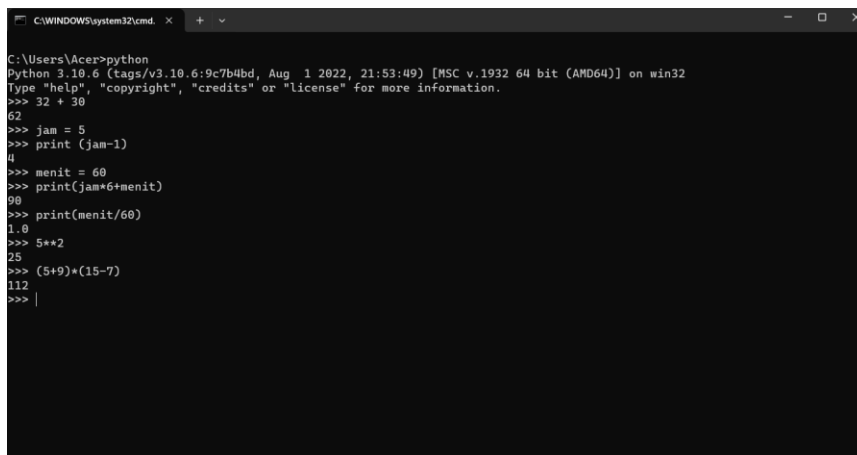
Operator dan Operand

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

Pada bagian ini secara khusus akan membahas operator aritmatika pada Python. Operator aritmatika adalah operator yang digunakan untuk melakukan operasi matematika, seperti penjumlahan, pengurangan, perkalian, pembagian, dan sebagainya. Tabel berikut menunjukkan jenis operator aritmatika.

Operator	Nama dan Fungsi	Contoh
+	Penjumlahan untuk 2 buah operand	$x + y$
-	Pengurangan untuk 2 buah operand	$x - y$
*	Perkalian untuk 2 buah operand	$x * y$
/	Pembagian untuk 2 buah operand	x / y
**	Pangkat untuk pemangkatan bilangan	$x ** y$

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



```
C:\Users\Acer>python
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 32 + 98
62
>>> jam = 5
>>> print(jam-1)
4
>>> menit = 60
>>> print(jam*6+menit)
90
>>> print(menit/60)
1.0
>>> 5**2
25
>>> (5+9)*(15-7)
112
>>> |
```

Expressions

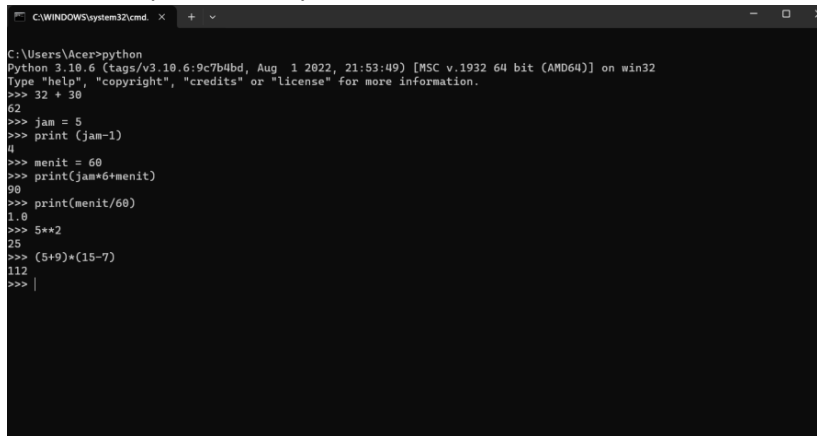
Expression merupakan representasi dari nilai dan dapat terdiri dari gabungan antara values, variable dan operator. Values dengan sendirinya dapat dianggap sebagai expression dan juga variabel. Secara umum, semuanya dapat disebut dengan expression.

17

x

x + 17

Ketika menggunakan expression dalam model interactive, interpreter akan melakukan evaluasi dan menampilkan hasilnya.



```
C:\WINDOWS\system32\cmd. x + v
C:\Users\Acer>python
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 32 + 30
62
>>> jam = 5
>>> print (jam-1)
4
>>> menit = 60
>>> print(jam*6+menit)
90
>>> print(menit/60)
1.0
>>> 5**2
25
>>> (5+9)*(15-7)
112
>>> |
```

Urutan Operasi

Urutan operasi berlaku ketika ekspresi berisi beberapa operator. Urutan operasi bergantung pada aturan prioritas. Untuk operasi matematika, Python mengikuti transformasi matematika.

Urutan operasi sering disingkat dengan **PEMDAS - Parentheses, Exponentiation, Multiplication and Division, Operator**.

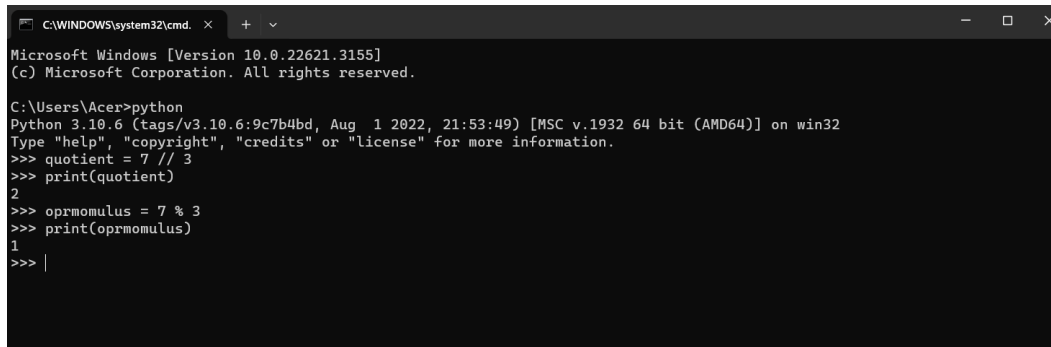
- **Parentheses** (Tanda kurung) - merupakan prioritas tertinggi dan digunakan untuk memaksa expression dalam urutan yang sesuai. Contohnya $2*(3-1)$ hasilnya 4, dan $(1+1) ** (5-2)$ hasilnya 8. Penggunaan tanda kurung dapat digunakan untuk membuat expression menjadi lebih mudah untuk dibaca, misalnya $(minute * 100) / 60$.
- **Exponentiation** (Eksponensial/Pemangkatan) - merupakan prioritas tertinggi berikutnya, contoh $2**1+1$ hasilnya 3, bukan 4, dan $3*1**3$ hasilnya 3 bukan 27
- **Multiplication and Division** (Perkalian dan Pembagian) - memiliki 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.
- **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.

Jika terjadi keraguan, silakan letakkan tanda kurung di dalam ekspresi untuk memastikan bahwa komposisinya sesuai dengan yang diinginkan.

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 (%).



```
C:\WINDOWS\system32\cmd. x + v
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Acer>python
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> quotient = 7 // 3
>>> print(quotient)
2
>>> oprmomulus = 7 % 3
>>> print(oprmomulus)
1
>>> |
```

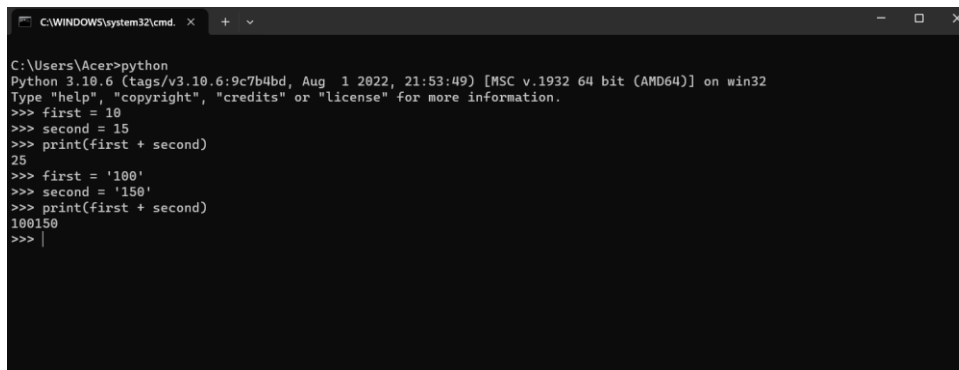
7 dibagi dengan 3 menghasilkan 2 dnegan sisa hasil bagi 1.

Contoh penggunaan operator modulus.

- Memeriksa satu angka dapat dibagi dengan yang lain, misal jika $x \% y$ adalah 0, maka x dapat dibagi oleh y.
- Dapat mengekstrak digit paling kanan atau digit dari suatu angka. Misalnya, $x \% 10$ menghasilkan digit x paling kanan (dalam basis 10). Demikian pula, $x \% 100$ menghasilkan dua digit terakhir.

String

Operator + ketika bekerja dengan string tidak berarti penjumlahan secara matematika, melainkan penggabungan antar string. Contoh:



```
C:\WINDOWS\system32\cmd. x + v
C:\Users\Acer>python
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> first = 10
>>> second = 15
>>> print(first + second)
25
>>> first = '100'
>>> second = '150'
>>> print(first + second)
100150
>>> |
```

Operator * juga bekerja dengan string dengan melakukan perkalian antara content string dan interger.

```
C:\WINDOWS\system32\cmd. x + v
C:\Users\Acer>python
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> first = 'test'
>>> second = 3
>>> print(first * second)
testtesttest
>>> |
```

Menangani Input dari Pengguna

Sebuah program biasanya memiliki alur kerja Input - Proses - Output yang alurnya ditunjukkan pada Gambar dibawah.

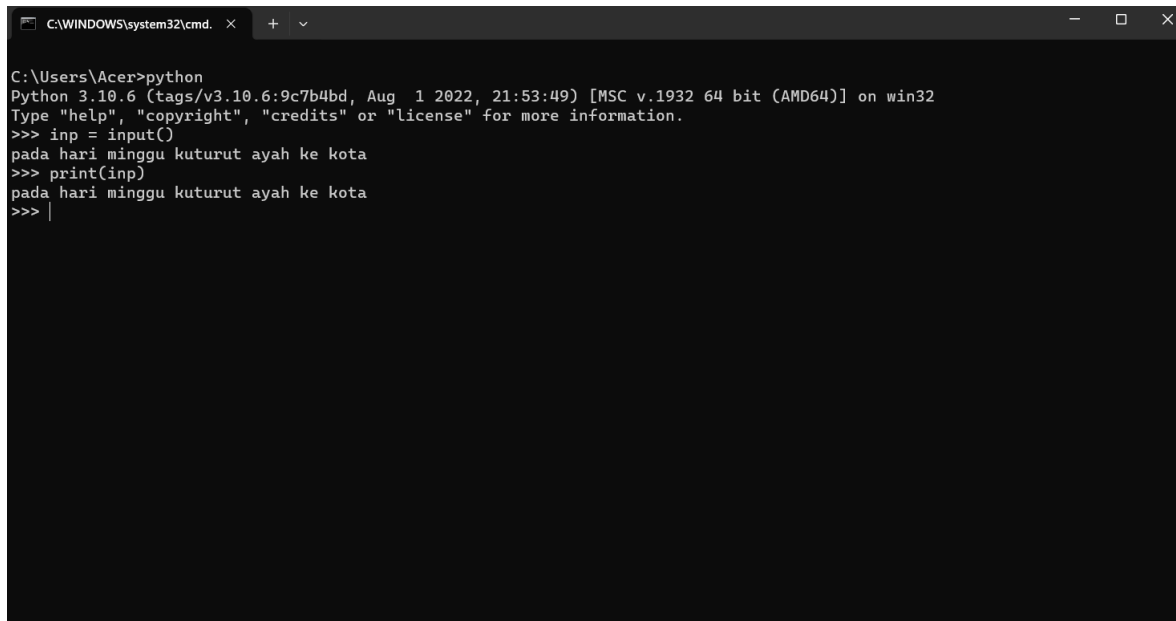
Input adalah data/masukan yang dibutuhkan supaya program bisa berjalan. Proses adalah langkah-langkah yang dilakukan oleh program untuk memecahkan masalah. Sedangkan Output adalah hasil yang didapatkan setelah menjalankan langkah-langkah tersebut. Sebagai contoh misalnya mengambil uang lewat ATM. Pengambilan uang melalui ATM dapat dibagi menjadi 3 bagian (Input-Proses-Output) tersebut, yaitu:



1. Masukkan kartu ATM. Masukkan PIN anda. Kartu ATM dan PIN adalah Input yang diperlukan supaya anda bisa mengambil uang.
2. Anda memilih menu Pengambilan Uang. Kemudian anda memasukkan nominal yang diinginkan. Bagian ini juga merupakan Input.
3. Mesin ATM akan memproses transaksi anda dengan menghubungi server Bank yang bersangkutan. Dilakukan berbagai macam pengecekan (misal: apakah saldonya cukup? apakah kartunya masih berlaku? apakah ada blokir?). Bagian ini disebut Proses.
4. Mesin ATM mengeluarkan uang, bukti pengambilan uang dan kartu anda. Uang yang keluar merupakan Output dari kegiatan ini. Selain itu, saldo anda juga berkurang. Pengurangan saldo juga merupakan hasil dari kegiatan ini.

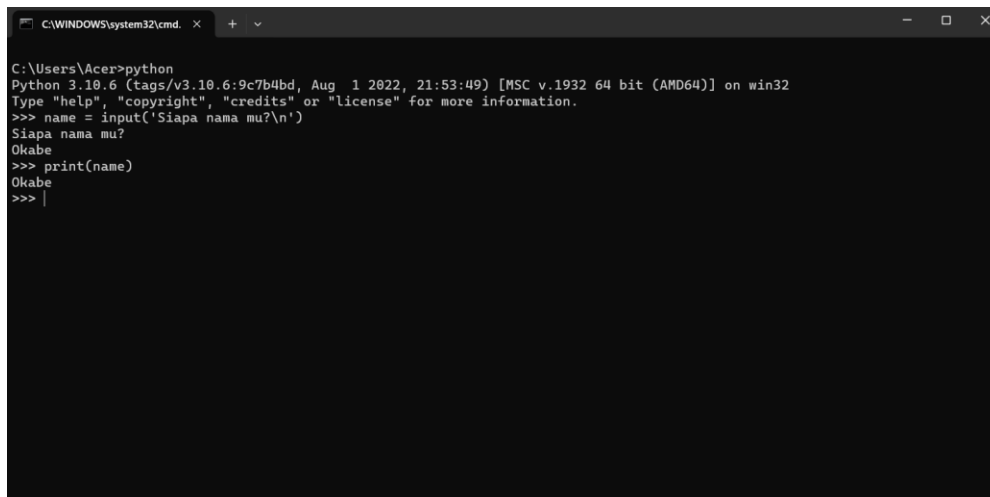
Python juga dapat memproses masukan pengguna. Input dalam hal ini adalah input teks yang dimasukkan oleh pengguna. Oleh karena itu, Python menyediakan fungsi bawaan yang disebut "input" untuk mendapatkan masukan dari keyboard. Ketika fungsi ini dipanggil, program akan berhenti dan

menunggu pengguna mengetikkan sesuatu. Ketika pengguna menekan tombol Enter, program melanjutkan dan mengembalikan apa yang diketik pengguna sebagai string.



```
C:\WINDOWS\system32\cmd. x + v
C:\Users\Acer>python
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> inp = input()
pada hari minggu kuturut ayah ke kota
>>> print(inp)
pada hari minggu kuturut ayah ke kota
>>> |
```

Sebelum mendapatkan input dari pengguna, lebih baik untuk mencetak prompt yang memberitahu pengguna apa yang harus diinput. String tersebut dapat diteruskan ke input untuk ditampilkan kepada pengguna sebelum berhenti untuk input.



```
C:\WINDOWS\system32\cmd. x + v
C:\Users\Acer>python
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> name = input('Siapa nama mu?\n')
Siapa nama mu?
Okabe
>>> print(name)
Okabe
>>> |
```

Tanda `\n` pada akhir perintah mewakili baris baru atau ganti baris sehingga hasil input pengguna muncul dibawah perintah.

Ketika mengharapkan pengguna untuk mengetik bilangan bulat, dapat dilakukan dengan mengonversi nilai kembali ke int menggunakan fungsi `int()`:

```
C:\WINDOWS\system32\cmd. X + v
C:\Users\Acer>python
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> prompt = 'Berapa suhu ruangan sekarang\n'
>>> suhu = input(prompt)
Berapa suhu ruangan sekarang
24
>>> int(suhu)
24
>>> int(suhu) + 5
29
>>> |
```

Akan menjadi error jika pengguna memasukkan data selain angka.

```
>>> prompt = 'Berapa suhu ruangan sekarang\n'
>>> suhu = input(prompt)
Berapa suhu ruangan sekarang
ruangan depan atau belakang?
>>> int(suhu)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: invalid literal for int() with base 10: 'ruangan depan atau belakang?'
>>> |
```

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.

```
# komentar pertama
print("hai dunia!") # komentar kedua
```

Outputnya

hai dunia!

Python tidak memiliki fitur komentar multibaris. Kita harus mengomentari satu persatu baris seperti berikut:

```
# Ini komentar
# Ini juga adalah komentar
# Ini juga masih komentar
```

MATERI 2

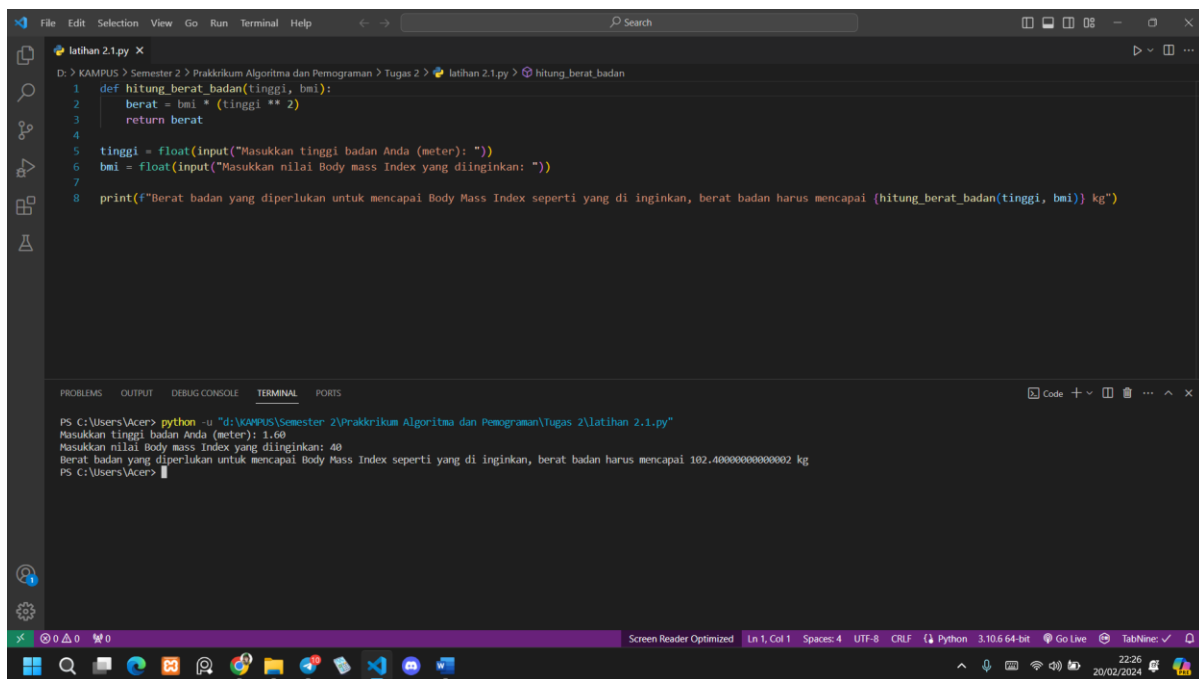
Penjelasan materi 2, dst... sesuai format ini.

BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

SOAL 1

Latihan 2.1



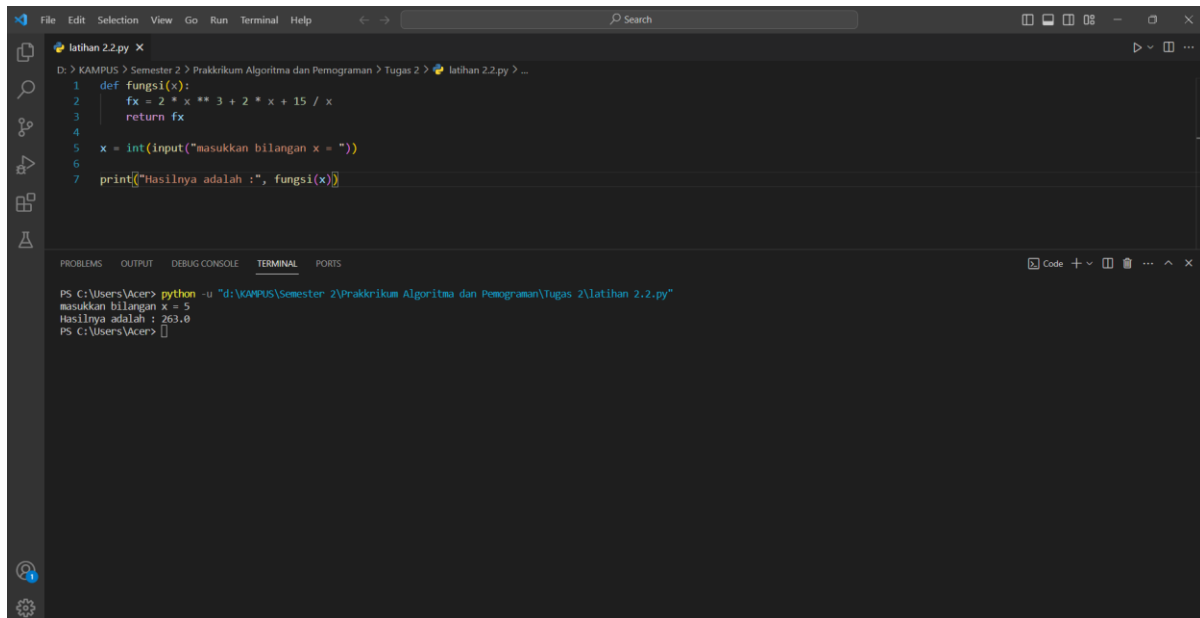
```
File Edit Selection View Go Run Terminal Help
latihan 2.1.py x
D: > KAMPUS > Semester 2 > Prakkrikum Algoritma dan Pemograman > Tugas 2 > latihan 2.1.py > hitung_berat_badan
1 def hitung_berat_badan(tinggi, bmi):
2     berat = bmi * (tinggi ** 2)
3     return berat
4
5 tinggi = float(input("Masukkan tinggi badan Anda (meter): "))
6 bmi = float(input("Masukkan nilai Body mass Index yang diinginkan: "))
7
8 print(f"Berat badan yang diperlukan untuk mencapai Body Mass Index seperti yang di inginkan, berat badan harus mencapai {hitung_berat_badan(tinggi, bmi)} kg")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\Acer> python -u "d:\KAMPUS\Semester 2\Prakkrikum Algoritma dan Pemograman\Tugas 2\latihan 2.1.py"
Masukkan tinggi badan Anda (meter): 1.60
Masukkan nilai Body mass Index yang diinginkan: 40
Berat badan yang diperlukan untuk mencapai Body Mass Index seperti yang di inginkan, berat badan harus mencapai 102.40000000000000 kg
PS C:\Users\Acer>
```

Screen Reader Optimized | Ln 1, Col 1 | Spaces: 4 | UTF-8 | CRLF | Python 3.10.6 64-bit | Go Live | TabNine: ✓ | 22:26 20/02/2024

SOAL 2

Latihan 2.2



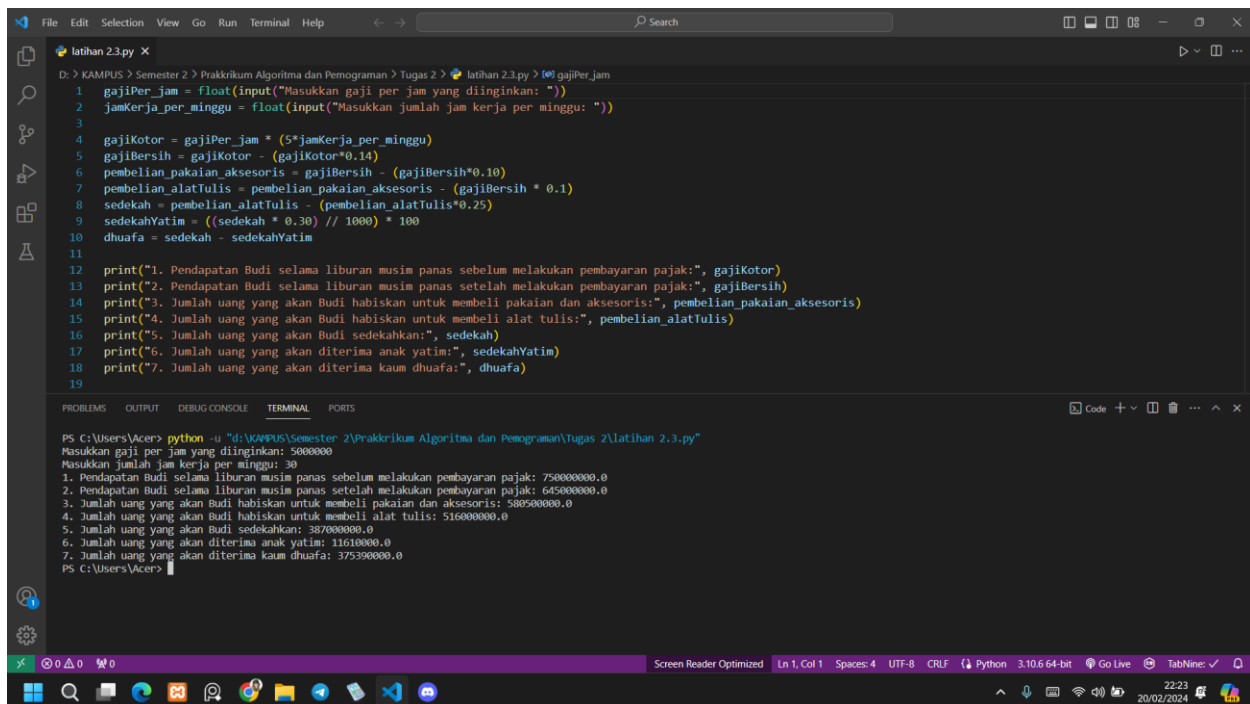
The screenshot shows a Python IDE with a file named 'latihan 2.2.py'. The code defines a function 'fungsi(x)' that calculates $2x^3 + 2x + 15/x$ and returns the result. It then prompts the user to input a value for 'x' and prints the result. The terminal shows the execution of the script, where the user inputs '5' and the output is '263.0'.

```
1 def fungsi(x):
2     fx = 2 * x ** 3 + 2 * x + 15 / x
3     return fx
4
5 x = int(input("masukkan bilangan x = "))
6
7 print("Hasilnya adalah :", fungsi(x))
```

```
PS C:\Users\Acer> python -u "d:\KAMPUS\Semester 2\Praktikum Algoritma dan Pemrograman\Tugas 2\latihan 2.2.py"
masukkan bilangan x = 5
Hasilnya adalah : 263.0
PS C:\Users\Acer>
```

SOAL 3

Latihan 2.3



The screenshot shows a Python IDE with a file named 'latihan 2.3.py'. The code calculates various financial and social values based on user input for 'gajiPer_jam' and 'jamKerja_per_minggu'. It calculates 'gajiKotor', 'gajiBersih', 'pembelian_pakaian_aksesoris', 'pembelian_alatTulis', 'sedekah', 'sedekahYatim', and 'dhuafa'. The terminal shows the execution of the script, where the user inputs '5000000' for 'gajiPer_jam' and '30' for 'jamKerja_per_minggu'. The output displays the calculated values for each category.

```
1 gajiPer_jam = float(input("Masukkan gaji per jam yang diinginkan: "))
2 jamKerja_per_minggu = float(input("Masukkan jumlah jam kerja per minggu: "))
3
4 gajiKotor = gajiPer_jam * (5*jamKerja_per_minggu)
5 gajiBersih = gajiKotor - (gajiKotor*0.14)
6 pembelian_pakaian_aksesoris = gajiBersih - (gajiBersih*0.10)
7 pembelian_alatTulis = pembelian_pakaian_aksesoris - (gajiBersih * 0.1)
8 sedekah = pembelian_alatTulis - (pembelian_alatTulis*0.25)
9 sedekahYatim = ((sedekah * 0.30) // 1000) * 100
10 dhuafa = sedekah - sedekahYatim
11
12 print("1. Pendapatan Budi selama liburan musim panas sebelum melakukan pembayaran pajak:", gajiKotor)
13 print("2. Pendapatan Budi selama liburan musim panas setelah melakukan pembayaran pajak:", gajiBersih)
14 print("3. Jumlah uang yang akan Budi habiskan untuk membeli pakaian dan aksesoris:", pembelian_pakaian_aksesoris)
15 print("4. Jumlah uang yang akan Budi habiskan untuk membeli alat tulis:", pembelian_alatTulis)
16 print("5. Jumlah uang yang akan Budi sedekahkan:", sedekah)
17 print("6. Jumlah uang yang akan diterima anak yatim:", sedekahYatim)
18 print("7. Jumlah uang yang akan diterima kaum dhuafa:", dhuafa)
19
```

```
PS C:\Users\Acer> python -u "d:\KAMPUS\Semester 2\Praktikum Algoritma dan Pemrograman\Tugas 2\latihan 2.3.py"
Masukkan gaji per jam yang diinginkan: 5000000
Masukkan jumlah jam kerja per minggu: 30
1. Pendapatan Budi selama liburan musim panas sebelum melakukan pembayaran pajak: 75000000.0
2. Pendapatan Budi selama liburan musim panas setelah melakukan pembayaran pajak: 64500000.0
3. Jumlah uang yang akan Budi habiskan untuk membeli pakaian dan aksesoris: 58050000.0
4. Jumlah uang yang akan Budi habiskan untuk membeli alat tulis: 51600000.0
5. Jumlah uang yang akan Budi sedekahkan: 38700000.0
6. Jumlah uang yang akan diterima anak yatim: 11610000.0
7. Jumlah uang yang akan diterima kaum dhuafa: 37539000.0
PS C:\Users\Acer>
```