



Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2025/2026

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUMINI 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	71251182
Nama Lengkap	Michael Dylan
Minggu ke / Materi	01 / Bahasa Pemrograman Python

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

BAGIAN 1: MATERI MINGGU 1 (40%)

1.3.1 Kenapa Python?

Python adalah bahasa pemrograman tingkat tinggi yang mudah dipelajari dan digunakan. Python bersifat interpreted dan mendukung Object Oriented Programming (OOP). Berdasarkan survei Stack Overflow (<https://survey.stackoverflow.co/2019#technology>), Python termasuk salah satu bahasa pemrograman yang paling populer karena sintaksnya sederhana dan dapat digunakan untuk berbagai keperluan. Logo Python bisa dilihat pada gambar 1.1.



Gambar 1.1: Logo Python (diambil dari <https://www.python.org/>)

Kesederhanaan bentuk dan sintaks Python dilihat dari perbandingan source code program “Hello World” jika dibandingkan dengan Java dan C.

Bahasa Pemrograman Java:

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

Bahasa Pemrograman C:

```
#include <stdio.h>  
  
int main()  
{  
    printf("Hello World");  
    return 0;  
}
```

Bahasa Pemrograman Python:

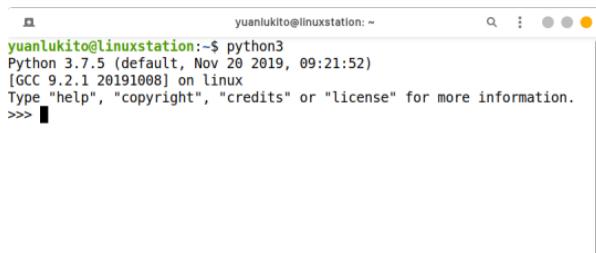
```
print("Hello World");
```

Python memiliki banyak kelebihan seperti pustaka yang lengkap, bersifat open source, dan mudah dipelajari karena sintaksnya sederhana. Python juga didukung oleh banyak dokumentasi dan

terus dikembangkan. Namun, Python memiliki beberapa kekurangan seperti kurang optimal untuk aplikasi mobile, penggunaan memori yang cukup besar, dan kecepatan lebih lambat dibandingkan bahasa seperti C. Meskipun begitu, Python tetap cocok untuk pemula karena mudah untuk belajar pemrograman.

1.3.2 Menginstall Python 3

Saat ini terdapat dua versi Python yang banyak digunakan, yaitu Python 2 dan Python 3 yang versi yang lebih baru. Pada sistem operasi Linux dan macOS, Python dapat dijalankan pada terminal di Ubuntu menggunakan perintah `python3` seperti yang ditunjukkan pada Gambar 1.2.



```
yuanlukito@linuxstation:~$ python3
Python 3.7.5 (default, Nov 20 2019, 09:21:52)
[GCC 9.2.1 20191008] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

Gambar 1.2: Python v3 di Ubuntu Linux, terinstall Python versi 3.7.5. Sumber (

https://lms.ukdw.ac.id/pluginfile.php/86892/mod_resource/content/2/01%20-%20Baha%20Pemrograman%20Python.pdf)

Untuk Windows, bisa menginstall distribusi Anaconda yang relatif mudah diinstall untuk Python. Pilihlah Anaconda Individual Edition 64-Bit Graphical Installer, seperti pada Gambar 1.3. Sesuaikan dengan sistem operasi yang sedang anda gunakan.

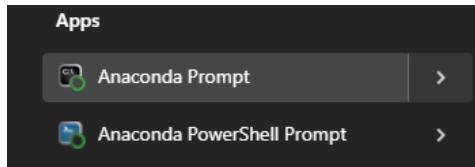


Gambar 1.3: Distribusi Anaconda Individual Edition. Sumber (<https://www.anaconda.com/download/success>)

1.3.3 Menjalankan Python Mode Interaktif

Untuk masuk mode interaktif, jalankan perintah `python3` pada terminal di Ubuntu. Jika anda

menggunakan Windows, jalankan Anaconda Prompt, kemudian ketikkan perintah python seperti yang ditunjukkan pada Gambar 1.4 dan Gambar 1.5.



Gambar 1.4: Menu Anaconda Prompt setelah terinstall di Windows.

```
(base) C:\Users\merro>python
Python 3.13.9 | packaged by Anaconda, Inc. | (main, Oct 21 2025, 19:09:58) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

A screenshot of the Anaconda Prompt window titled "Anaconda Prompt - python". It shows the Python version information: Python 3.13.9, packaged by Anaconda, Inc., main branch, Oct 21 2025, 19:09:58, MSC v.1929 64 bit (AMD64) on win32. The prompt ">>>" is visible at the bottom.

Gambar 1.5: Informasi versi Python akan ditampilkan dan siap menerima perintah pada Anaconda Prompt - Python

Mode interaktif memungkinkan pengguna memasukkan perintah satu per satu dan langsung diproses oleh Python. contohnya, pengguna dapat menghitung luas segitiga dengan alas tinggi 10 cm dan tinggi 8 cm secara berurutan seperti Gambar 1.6.

- alas = 10
- tinggi = 8
- luas = 0.5 * alas * tinggi
- luas

```
(base) C:\Users\merro>python
Python 3.13.9 | packaged by Anaconda, Inc. | (main, Oct 21 2025, 19:09:58) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> alas = 10
>>> tinggi = 8
>>> luas = 0.5 * alas * tinggi
>>> luas
40.0
>>>
```

A screenshot of the Anaconda Prompt window titled "Anaconda Prompt - python". It shows the calculation of the area of a triangle with base 10 and height 8, resulting in 40.0. The prompt ">>>" is visible at the bottom.

Gambar 1.6: Menghitung Luas Segitiga dengan bantuan Anaconda Prompt – Python3

Perintah pertama dan kedua untuk mengisi nilai variabel panjang dengan 10 dan tinggi dengan 8. Perintah ketiga untuk menghitung luas segitiga dengan rumus $0.5 \times \text{panjang} \times \text{tinggi}$ dan menyimpannya ke dalam variabel luas. Perintah terakhir digunakan untuk menampilkan nilai dari variabel luas. Variabel adalah tempat untuk menyimpan nilai yang akan digunakan dalam proses selanjutnya. Untuk mengakhiri mode interaktif, pengguna dapat mengetikkan perintah exit().

1.3.4 Editor untuk Python

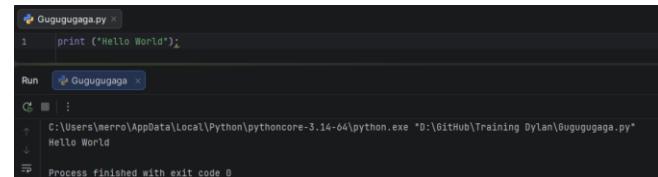
Untuk menulis program Python, diperlukan editor seperti Visual Studio Code, PyCharm, Spyder, ActivePython, atau IDLE. PyCharm dan Spyder merupakan IDE dengan fitur lebih lengkap, namun untuk praktikum, Visual Studio Code sudah cukup karena sederhana dan mudah digunakan. Contoh tampilan Visual Studio Code yang sudah terpasang Python Extension bisa dilihat pada Gambar 1.7 dan Gambar 1.8. Editor lain seperti PyCharm punya fitur lebih lengkap dan bisa digunakan untuk berbagai macam bentuk project Python. Tampilan PyCharm bisa dilihat pada Gambar 1.9.



Gambar 1.7: Python Extension untuk Visual Studio Code.



Gambar 1.8: Tampilan Visual Studio Code saat menjalankan script Python.



Gambar 1.9: Tampilan dari PyCharm saat mengedit file Python.

1.3.5 Menjalankan Script Python di Terminal/Console

sebelumnya sudah mencoba menjalankan Python dalam mode interaktif (seperti pada Gambar 1.6). Pada mode interaktif, pengguna mengetik perintah satu per satu dan hasilnya langsung dijalankan oleh Python. Mode ini cocok untuk mencoba perintah baru, tetapi perintah harus diketik ulang jika keluar dari mode tersebut. Python juga memiliki mode script, yaitu menjalankan perintah yang disimpan dalam file **.py**. Script dapat dijalankan melalui Terminal atau Command Prompt dengan perintah **python3 namafile.py** pada Ubuntu atau **python namafile.py** pada Windows. Seperti dibawah ini.

```
# nilai kurs 1 US$ ke IDR
```

```
kursusd = 13950
```

```

# informasi program

print('Program konversi US$ ke IDR')

print('Kurs saat ini 1 US$ = ',kursusd, 'Rupiah')

# input jumlah US$ yang mau ditukar

jumlahusd = float(input('Masukkan jumlah uang yang mau ditukar ke Rupiah: '))

# hitung nilainya dalam Rupiah

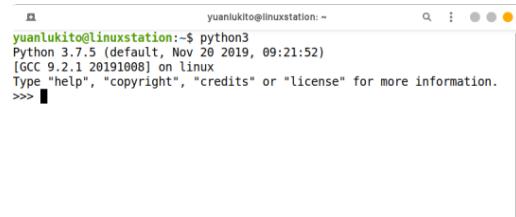
dalamrupiah = jumlahusd * kursusd

# tampilkan hasilnya

print('Hasil konversi = Rp. ', dalamrupiah)

```

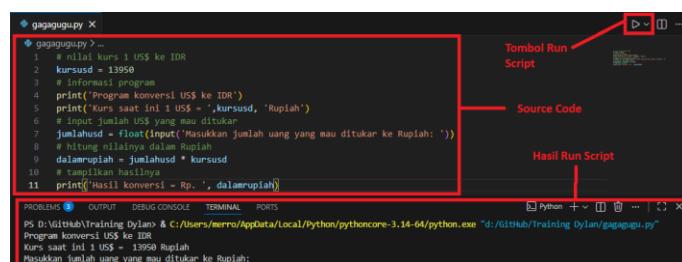
Untuk jalankan script tersebut, ketik perintah **python3 nilaitukardollar.py** (Ubuntu) atau **python nilaitukardollar.py** (Windows). Pada Ubuntu, tampilan hasilnya bisa dilihat pada Gambar 1.10.



Gambar 1.10: Menjalankan script Python di Terminal Ubuntu. Sumber (

https://lms.ukdw.ac.id/pluginfile.php/86892/mod_resource/content/2/01%20-%20Bahan%20Pemrograman%20Python.pdf)

Baris diawali dengan tanda **#** merupakan komentar. Komentar digunakan memberikan keterangan pada program dan tidak dijalankan oleh Python. Jika menggunakan editor seperti Visual Studio Code, pengguna tidak perlu membuka terminal secara manual karena sudah tersedia tombol **Run** untuk menjalankan script. Seperti pada Gambar 1.11.



Gambar 1.11: Menjalankan script Python pada Visual Studio Code.

1.3.6 Mencari Bug dan Memperbaikinya (debugging)

Bug adalah kesalahan dalam program yang terdiri dari dua jenis, yaitu syntax error atau compile

error akibat kesalahan penulisan, serta runtime error yang terjadi saat program dijalankan. Python menjalankan kode perbaris, jika terjadi kesalahan program akan berhenti. Pada contoh program diskon, terjadi NameError karena perbedaan penulisan variabel, yaitu nominal_diskon ditulis menjadi nominaldiskon, sehingga variabel tidak dikenali oleh program. Seperti di Gambar 1.12. Saat ada kesalahan sintaks akan diberi tahu seperti di Gambar 1.12.

```

guguga.py > ...
1 belanja = 100000
2 #besarnya diskon (dalam persen)
3 diskon = 30
4 #besarnya diskon
5 nominal_diskon = (diskon/100) * belanja
6 #hitung jumlah yang harus dibayar
7 bayar = nominaldiskon - belanja
8
guguga.py | 1 problem
"nominaldiskon" is not defined PyLance(reportUndefinedVariable)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\GitHub\Training Dylan & C:/Users/merro/AppData/Local/Python/pythoncore-3.14-64/python.exe "d:/GitHub/Training Dylan/guguga.py"
Traceback (most recent call last):
File "d:/GitHub/Training Dylan/guguga.py", line 8, in <module>
    bayar = nominaldiskon - belanja
           ^^^^^^^^^^
NameError: name 'nominaldiskon' is not defined. Did you mean: 'nominal_diskon'?

```

Gambar 1.12: Kesalahan muncul saat dijalankan dan diberitahu sintaks salah pada Vscode.

Setelah diperbaiki, maka sekarang script bisa dijalankan sampai selesai seperti di Gambar 1.13. Namun hasilnya tidak yang diharapkan. Seharusnya, jika harga barang Rp100.000 dan diskon 30%, hasil totalnya Rp70.000, tapi yang dihasilkan nilai -70000. Kesalahan ini termasuk runtime error, yaitu kesalahan yang terjadi saat program dijalankan akibat kesalahan logika atau algoritma. Kesalahan terdapat pada perhitungan pembayaran karena rumus yang digunakan terbalik, seharusnya harga dikurangi diskon.

```

guguga.py > ...
3 #besarnya diskon (dalam persen)
4 diskon = 30
5 #besarnya diskon
6 nominal_diskon = (diskon/100) * belanja
7 #hitung jumlah yang harus dibayar
8 bayar = nominal_diskon - belanja
9 #tampilkan hasilnya
10 print('Anda harus membayar: Rp. ', bayar)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\GitHub\Training Dylan & C:/Users/merro/AppData/Local/Python/pythoncore-3.14-64/python.exe "d:/GitHub/Training Dylan/guguga.py"
Anda harus membayar: Rp. -70000.0

```

Gambar 1.13: Script berhasil dijalankan pada Vscode namun tidak sesuai diharapkan

```

guguga.py > ...
1 #jumlah belanja (dalam rupiah)
2 belanja = 100000
3 #besarnya diskon (dalam persen)
4 diskon = 30
5 #besarnya diskon
6 nominal_diskon = (diskon/100) * belanja
7 #hitung jumlah yang harus dibayar
8 bayar = belanja - nominal_diskon
9 #tampilkan hasilnya
10 print('Anda harus membayar: Rp. ', bayar)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\GitHub\Training Dylan & C:/Users/merro/AppData/Local/Python/pythoncore-3.14-64/python.exe "d:/GitHub/Training Dylan/guguga.py"
Anda harus membayar: Rp. 70000.0

```

Gambar 1.14: Hasil sudah sesuai diharapkan yang dijalankan di Vscode

Setelah diperbaiki, script tersebut akan menghasilkan hasil yang diharapkan seperti Gambar 1.14

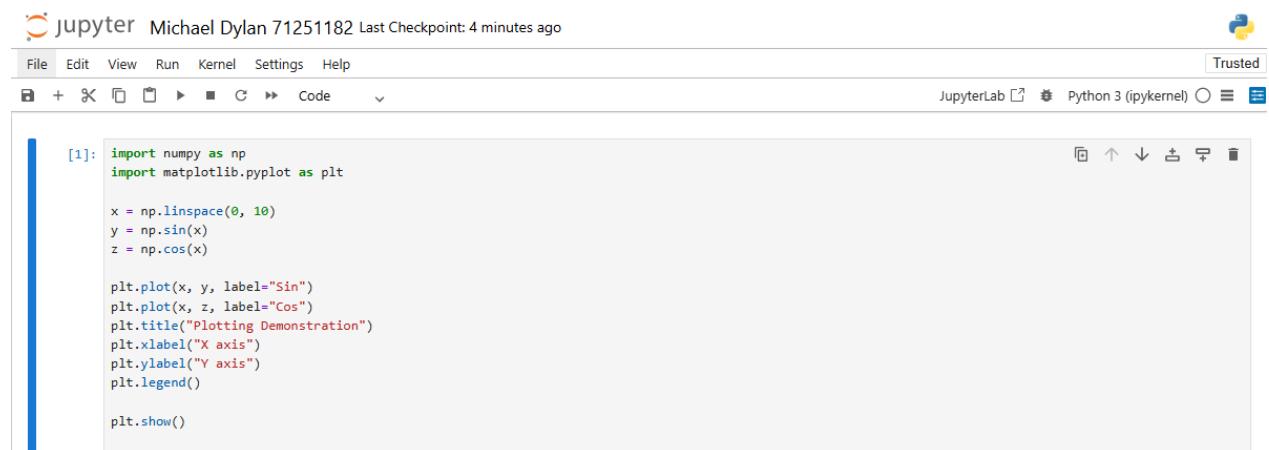
BAGIAN 2: LATIHAN MANDIRI (60%)

Link Github = https://github.com/MichaelDylan-ti/71251182_michaeldylan/tree/main

LATIHAN 1.1

Buatlah satu Notebook baru, berilah nama notebook tersebut dengan format Nama lengkap (NIM) anda. Kemudian ketikkan ulang kode program seperti yang ada di Gambar 1.24. Tuliskan langkah-langkah yang anda lakukan secara berurutan, sampai mendapatkan hasil seperti pada gambar tersebut. Perhatikan, anda harus menjelaskan langkah-langkah disertai gambar/screen-shot. Jangan lupa memasukkan langkah-langkah instalasi package-package yang diperlukan (matplotlib dan numpy). Instalasi package tersebut biasanya tidak diperlukan jika anda menginstall Python dari Anaconda.

Source Code:



The screenshot shows a Jupyter Notebook interface. At the top, there's a header with the logo, the user's name "Michael Dylan 71251182", and a "Last Checkpoint: 4 minutes ago" message. To the right of the header is a "Trusted" badge. Below the header is a menu bar with options: File, Edit, View, Run, Kernel, Settings, Help. Underneath the menu bar is a toolbar with various icons for file operations like new, open, save, and run. On the far right of the toolbar is a "Python 3 (ipykernel)" badge. The main area is a code cell labeled [1]: containing Python code for plotting sine and cosine functions. The code imports numpy and matplotlib.pyplot, generates data points for x, y, and z, and then plots two lines with labels "Sin" and "Cos". It also sets the title, x-axis label, y-axis label, and adds a legend before showing the plot.

```
[1]: import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(0, 10)
y = np.sin(x)
z = np.cos(x)

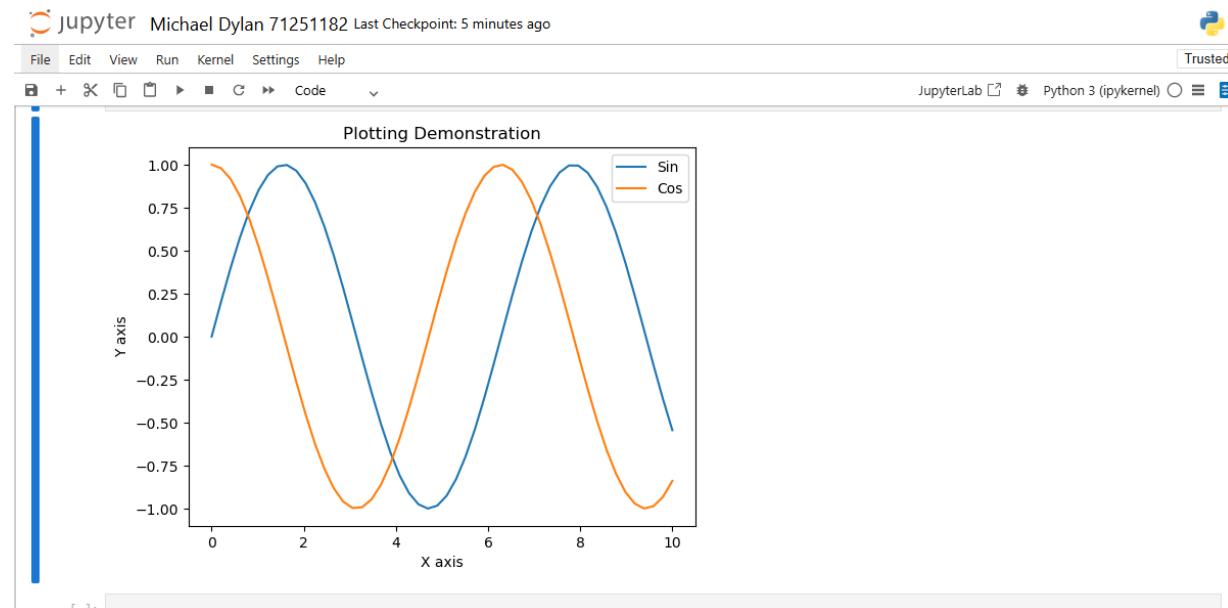
plt.plot(x, y, label="Sin")
plt.plot(x, z, label="Cos")
plt.title("Plotting Demonstration")
plt.xlabel("X axis")
plt.ylabel("Y axis")
plt.legend()

plt.show()
```

Gambar 1.15: Source Code untuk Latihan 1.1. Menggunakan Jupiter Notebook

Pada Gambar 1.15. Source code tersebut menggunakan Python untuk membuat grafik fungsi sinus (\sin) dan kosinus (\cos). Pertama, kode mengimpor pustaka numpy untuk membuat data angka dan matplotlib.pyplot untuk menggambar grafik. Variabel x berisi angka dari 0 sampai 10, sedangkan y berisi nilai $\sin(x)$ dan z berisi nilai $\cos(x)$. Fungsi `plt.plot()` digunakan untuk menggambar kedua grafik tersebut dengan label masing-masing. Selanjutnya, `plt.title()`, `plt.xlabel()`, `plt.ylabel()`, dan `plt.legend()` menambahkan judul, nama sumbu, dan keterangan garis pada grafik. Fungsi `plt.show()` menampilkan grafik.

Output:



Gambar 1.16: Output dari Source Code yang dijalankan di Jupiter Notebook

Gambar 1.16 ialah Outputnya yang terdapat grafik dua garis, satu menunjukkan fungsi sinus dan satu lagi fungsi kosinus, masing-masing diberi label dan terlihat jelas pada sumbu X dan Y.

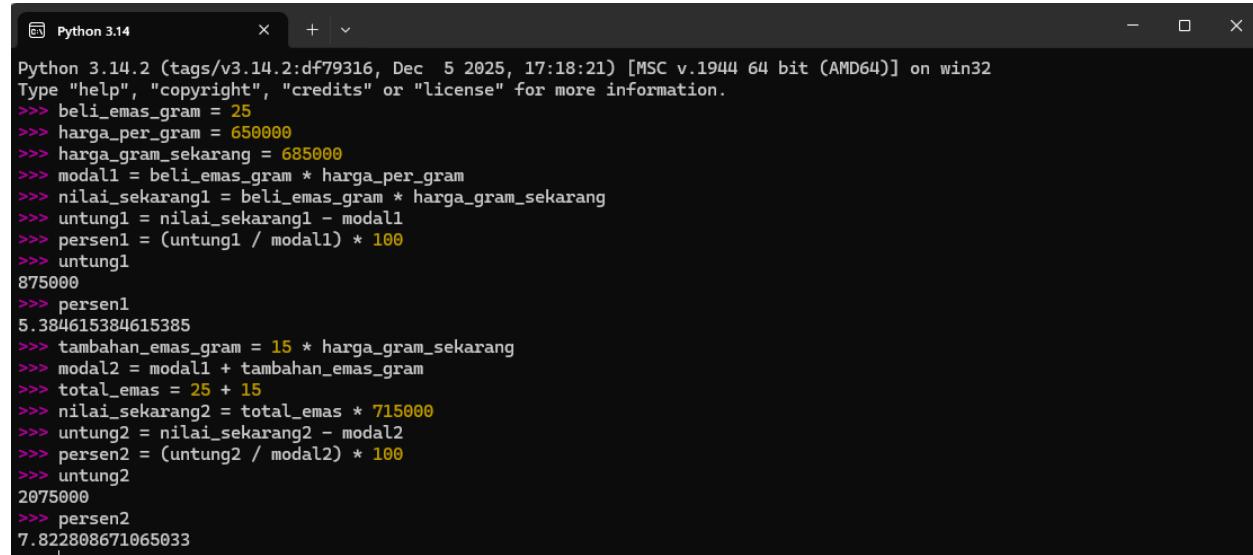
LATIHAN 1.2

Gerard membeli emas 25 gram dengan harga Rp. 650.000/gram. Jika sekarang harga emas menjadi Rp. 685.000/gram,

- Berapa keuntungan yang didapat oleh Gerard (dalam Rp dan dalam %)?
- Jika Gerard kemudian membeli lagi 15 gram emas dengan harga Rp. 685.000, maka Gerard sekarang memiliki total 40 gram emas. Jika kemudian harga emas naik lagi menjadi Rp. 715.000, berapa keuntungan yang didapat oleh Gerard (dalam Rp dan dalam %)?

Gunakan Python mode interaktif untuk menyelesaikan permasalahan tersebut. Tampilkan dan jelaskan secara lengkap langkah-langkah yang anda lakukan, dimulai dari membuka Python mode interaktif di komputer anda masing-masing.

Source Code:



```
Python 3.14.2 (tags/v3.14.2:df79316, Dec  5 2025, 17:18:21) [MSC v.1944 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> beli_emas_gram = 25
>>> harga_per_gram = 650000
>>> harga_gram_sekarang = 685000
>>> modal1 = beli_emas_gram * harga_per_gram
>>> nilai_sekarang1 = beli_emas_gram * harga_gram_sekarang
>>> untung1 = nilai_sekarang1 - modal1
>>> persen1 = (untung1 / modal1) * 100
>>> untung1
875000
>>> persen1
5.384615384615385
>>> tambahan_emas_gram = 15 * harga_gram_sekarang
>>> modal2 = modal1 + tambahan_emas_gram
>>> total_emas = 25 + 15
>>> nilai_sekarang2 = total_emas * 715000
>>> untung2 = nilai_sekarang2 - modal2
>>> persen2 = (untung2 / modal2) * 100
>>> untung2
2075000
>>> persen2
7.82288671065033
```

Gambar 1.17: Source Code untuk Latihan 1.2. Menggunakan Terminal dari Python 3.14

Source code pada Gambar 1.17. Menggunakan variabel untuk menyimpan jumlah emas, harga beli, dan harga jual saat ini. Pertama, dihitung modal awal dengan mengalikan jumlah gram emas dengan harga beli, kemudian nilai emas saat ini dihitung dengan mengalikan jumlah gram dengan harga jual sekarang. Selisih antara nilai saat ini dan modal menghasilkan keuntungan dalam rupiah, sedangkan persentase keuntungan dihitung dengan membagi keuntungan dengan modal lalu dikali 100. Untuk kasus kedua, prosedurnya sama, hanya jumlah emas dan modal total yang ditambahkan setelah pembelian tambahan.

Output:

```
>>> untung1
875000
>>> persen1
5.384615384615385
```

Gambar 1.18: Output 1 dari Gambar 1.17

```
>>> untung2  
2075000  
>>> persen2  
7.822808671065033
```

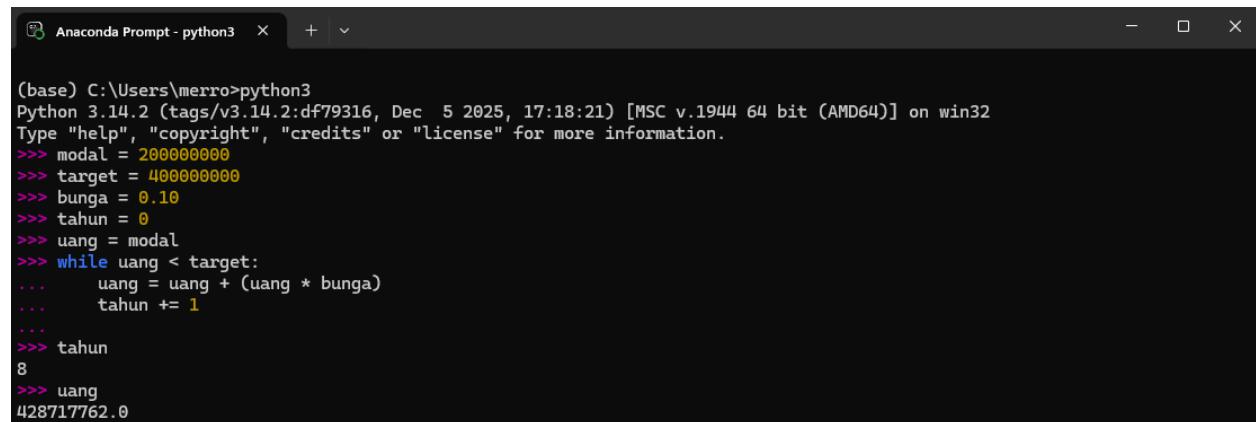
Gambar 1.19: Output 2 dari Gambar 1.17

Pada Gambar 1.18 dan 1.19. Output dari program menunjukkan keuntungan Gerard: untuk pembelian awal 25 gram, keuntungannya Rp875.000 atau 5,38%. Setelah membeli tambahan 15 gram dan harga naik lagi, keuntungan total menjadi Rp2.075.000 atau 7,82%. Dengan cara ini, Python memudahkan perhitungan keuntungan investasi emas secara cepat dan akurat.

LATIHAN 1.3

Berkaitan dengan compound interest pada Contoh 1.2 dan 1.3, jika Erika memiliki uang 200 juta rupiah dan ingin disimpan di deposito Pasti Cuan sampai uangnya menjadi minimal 400 juta, berapa lama waktu yang dibutuhkan? catatan: bunga 10% per-tahun. Gunakan Python mode interaktif untuk menyelesaikan permasalahan tersebut. Tampilkan dan jelaskan secara lengkap langkah-langkah yang anda lakukan, dimulai dari membuka Python mode interaktif di komputer anda masing-masing.

Source Code:



```
Anaconda Prompt - python3  
(base) C:\Users\merro>python3  
Python 3.14.2 (tags/v3.14.2:df79316, Dec 5 2025, 17:18:21) [MSC v.1944 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license" for more information.  
>>> modal = 200000000  
>>> target = 400000000  
>>> bunga = 0.10  
>>> tahun = 0  
>>> uang = modal  
>>> while uang < target:  
...     uang = uang + (uang * bunga)  
...     tahun += 1  
...  
>>> tahun  
8  
>>> uang  
428717762.0
```

Gambar 1.20: Source Code untuk Latihan 1.3. Menggunakan Anaconda Prompt – python3

Pada gambar 1.20. Program ini pakai while loop buat hitung pertumbuhan uang Erika tiap tahun dengan bunga majemuk. Modal awal disimpan di modal, target di target, bunga di bunga. uang

simpan jumlah saat ini, tahun buat hitung berapa lama. Tiap tahun, uang nambah 10% dari jumlah sebelumnya, sampai mencapai atau lewat target.

Output:

```
>>> tahun  
8  
>>> uang  
428717762.0
```

Gambar 1.21: Output dari Gambar 1.20

Pada Gambar 1.21 Erika butuh 8 tahun supaya uangnya minimal Rp400 juta, setelah itu jumlah uang ditampilkan 428717762.0 (program menghitung uang tiap tahun secara tepat, tanpa pembulatan ke jutaan) atau sekitar 429jt (jika dibulatkan jutaan).Jadi, bunga majemuk bikin uang tumbuh makin cepat karena dihitung dari total uang yang terus bertambah.