



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 | 71251192 |
| Nama Lengkap | Albert Gerald Pratama Mesakh |
| Minggu ke / Materi | 01 / Bahasa Pemrograman Python |

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

BAGIAN 1

Bahasa Pemrograman Python

Pada praktikum minggu ini, saya sudah mempelajari dasar-dasar bahasa pemrograman Python. Materi yang diberikan yaitu baru awal pengenalan, tetapi sangat penting karena menjadi dasar untuk memahami materi pemrograman untuk kedepannya. Di minggu ini saya belajar mulai dari apa itu Python (pengertian), kenapa Python digunakan, bagaimana cara menginstalnya, menjalankan Python dalam berbagai mode, menulis program sederhana, sampai memahami kesalahan program dan cara memperbaiknya. Seluruh materi disampaikan secara bertahap sehingga mudah diikuti, terutama bagi pemula yang baru pertama kali belajar pemrograman.

1. Pengenalan Bahasa Pemrograman Python

Python adalah bahasa pemrograman tingkat tinggi yang dirancang agar mudah dibaca dan ditulis oleh manusia. Saat pertama kali mempelajari Python, saya langsung merasakan bahwa penulisan kodennya tidak rumit dan tidak banyak simbol yang membuat bingung. Python banyak digunakan di berbagai bidang seperti pengolahan data, web, kecerdasan buatan, dan otomasi karena sintaksnya sederhana.



Salah satu alasan utama Python cocok untuk pemula adalah karena satu perintah bisa langsung menghasilkan output tanpa harus menulis banyak baris kode. Contohnya untuk menampilkan tulisan **Hello World**, Python hanya membutuhkan satu baris saja.

```
print("Hello World!")
```

Pada bahasa pemrograman Python:

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

Output:

Hello World!

Jika dibandingkan dengan bahasa lain seperti Bahasa C atau Java, Python jauh lebih singkat. Hal ini membuat proses belajar logika pemrograman menjadi lebih fokus ke cara berpikir, bukan ke penulisan yang rumit.

Ilustrasi perbandingan tingkat kesederhanaan kode:

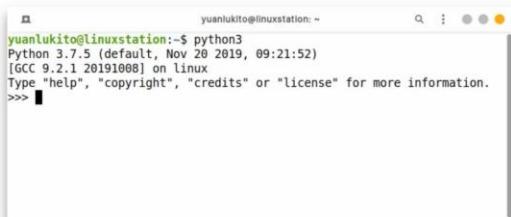
C / Java : banyak baris kode

Python : sedikit baris kode

2. Instalasi Python

Sebelum bisa menggunakan Python, langkah pertama yang harus dilakukan adalah menginstal Python di komputer. Pada praktikum dijelaskan bahwa Python yang digunakan adalah python versi 3. Untuk pengguna Windows, instalasi yang disarankan adalah menggunakan Anaconda, karena sudah menyediakan Python beserta tools pendukung seperti Jupyter Notebook. Sedangkan untuk sistem operasi Linux, Python biasanya sudah terpasang secara otomatis.

Setelah proses instalasi selesai, Python dapat dicek melalui Command Prompt atau Terminal dengan mengetik perintah: python atau python3. Jika muncul informasi versi Python, maka Python sudah siap digunakan.



```
yuanlukito@linuxstation:~$ python3
Python 3.7.5 (default, Nov 28 2019, 09:21:52)
[GCC 9.2.1 20191008] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> [REDACTED]
```

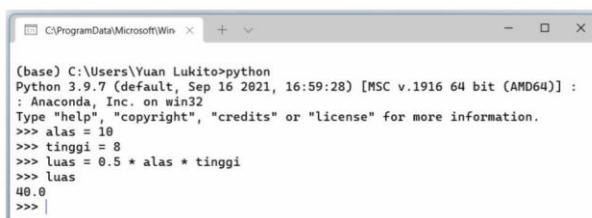
Ilustrasi proses instalasi:

Download Python/Anaconda kemudian Install lalu Buka Terminal / CMD maka Python siap

3. Menjalankan Python Mode Interaktif

Mode interaktif adalah cara menjalankan Python dengan mengetik perintah satu per satu, lalu hasilnya langsung muncul. Mode ini sangat membantu saat mau mencoba perhitungan yang sederhana atau mengecek apakah suatu perintah sudah berjalan dengan benar.

Contoh yang dilakukan pada praktikum adalah menghitung luas segitiga.



```
(base) C:\Users\Yuan Lukito>python
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)] :
: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> alas = 10
>>> tinggi = 8
>>> luas = 0.5 * alas * tinggi
>>> luas
40.0
>>> [REDACTED]
```

Output:

40.0

Dari contoh ini saya belajar Python menggunakan variabel untuk menyimpan nilai, dan hasil perhitungan bisa langsung dilihat tanpa perlu membuat program yang panjang. Mode interaktif ini cocok digunakan untuk belajar, tetapi kurang cocok untuk program yang besar karena perintah harus diketikkan ulang setiap kali keluar dari Python.

Ilustrasi mode interaktif:

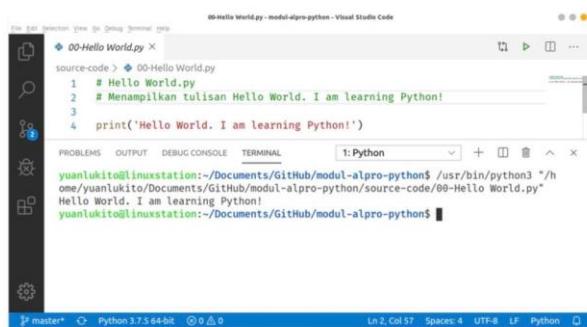
Ketik perintah, kemudian Tekan Enter, lalu Hasil langsung muncul

4. Menggunakan Editor Python

Untuk membuat program yang lebih panjang dan rapi, dibutuhkan editor. Pada praktikum ditunjukan editor Python seperti Visual Studio Code karena tampilannya sederhana dan mudah digunakan.



Dengan editor, program disimpan dalam file berekstensi .py, sehingga bisa dijalankan kapan saja tanpa perlu mengetik ulang perintah. Ini contoh tampilan Visual Studio code saat menjalankan script Phyton:



Ilustrasi penggunaan editor:

Tulis kode di editor, kemudian Simpan sebagai .py dan Jalankan

5. Menjalankan Script Python

Selain mode interaktif, Python juga bisa dijalankan dalam bentuk script. Program ditulis di file .py lalu dijalankan melalui terminal atau bisa juga langsung dari editor.

Contoh program yang dipelajari adalah konversi nilai dollar ke rupiah.

```
1 # nilai kurs 1 US$ ke IDR
2 kursusd = 13950
3
4 # informasi program
5 print('Program konversi US$ ke IDR')
6 print('Kurs saat ini 1 US$ = ,kursusd, 'Rupiah')
7 # input jumlah US$ yang mau ditukar
8 jumlahusd = float(input('Masukkan jumlah uang yang mau ditukar ke Rupiah: '))
9 # hitung nilainya dalam Rupiah
10 dalamrupiah = jumlahusd * kursusd
11 # tampilkan hasilnya
12 print('Hasil konversi = Rp. ', dalamrupiah)
13
```

Contoh Output:

Program konversi US\$ ke IDR

Masukkan jumlah dollar: 10

Hasil konversi = Rp. 139500

Dari program ini saya belajar cara menerima input dari pengguna, mengubah tipe data, serta menampilkan hasil perhitungan ke layar.

6. Bug dan Debugging

The screenshot shows the Visual Studio Code interface with the file 'nilaitukardollar.py' open. The code defines a variable 'kursusd' (13950), prints a welcome message, takes user input for 'jumlahusd', calculates 'dalamrupiah', and prints the result. A red circle highlights the 'Tombol untuk Run script' (Run script button) in the top right corner of the terminal tab.

```
# 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)
```

The screenshot shows a terminal window with the command 'python3 contohdebug.py' run. It outputs an error message: 'NameError: name 'nominaldiskon' is not defined'. Annotations explain that there is a mistake in line 11 ('ada kesalahan di baris 11') and that a variable is not defined ('ada variabel yang belum didefinisikan').

```
yuanlukito@linuxstation:~/Documents/python-codes$ python3 contohdebug.py
Traceback (most recent call last):
File "contohdebug.py", line 11, in <module>
    bayar = nominaldiskon - belanja
NameError: name 'nominaldiskon' is not defined
```

Bug adalah sebuah kesalahan dalam program. Pada praktikum dijelaskan bahwa bug dibagi menjadi dua jenis utama, yaitu syntax error dan runtime error.

Syntax error bisa terjadi karena salah menulis kode, misalnya salah nama variabel. Sedangkan runtime error terjadi ketika program berjalan tetapi hasilnya tidak sesuai dengan yang diharapkan karena kesalahan dalam logika.

Contoh kesalahan logika:

bayar = nominaldiskon - belanja

Seharusnya:

bayar = belanja - nominaldiskon

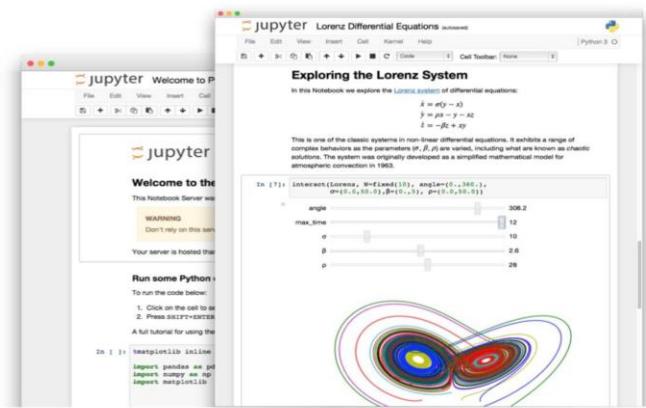
Kesalahan seperti ini mengajarkan saya bahwa program bisa saja berjalan tanpa error, tetapi tetap menghasilkan hasil yang salah jika logikanya juga salah.

Ilustrasi debugging:

Program error terus Baca pesan yang error, kemudian Cek kode lalu Perbaiki

7. Jupyter Notebook

Jupyter Notebook adalah aplikasi berbasis web yang digunakan untuk menjalankan Python secara interaktif di browser. Jupyter sangat cocok untuk pembelajaran karena kode dan output bisa ditampilkan bersamaan.



Di Jupyter, program dijalankan per *cell*, sehingga tidak harus menjalankan seluruh kode sekaligus.

Ilustrasi Jupyter Notebook:

Cell kode → Run → Output muncul di bawahnya

SUMBER YANG DIGUNAKAN:

☒ **Downey, A. B. (2015).** *Think Python: How to Think Like a Computer Scientist* (2nd ed.). O'Reilly Media.

Buku ini digunakan sebagai referensi dasar untuk memahami konsep pemrograman Python, terutama bagi pemula. Penjelasan mengenai variabel, mode interaktif, dan logika pemrograman mengacu pada buku ini.

☒ **Python Software Foundation. (2023).** *Python Documentation*.

Sumber resmi Python.

<https://docs.python.org/3/>

Dokumentasi resmi Python yang menjadi acuan utama dalam penggunaan sintaks, fungsi dasar seperti `print()` dan `input()`, serta cara menjalankan Python di berbagai mode.

 **Lutz, M. (2013).** *Learning Python* (5th ed.). O'Reilly Media.

Digunakan sebagai referensi pendukung untuk penjelasan konsep dasar Python, penggunaan script .py, serta perbedaan mode interaktif dan mode script.

 **Anaconda, Inc. (2023).** *Anaconda Documentation*.

<https://docs.anaconda.com/>

Referensi untuk proses instalasi Python menggunakan Anaconda dan pengenalan tools pendukung seperti Jupyter Notebook.

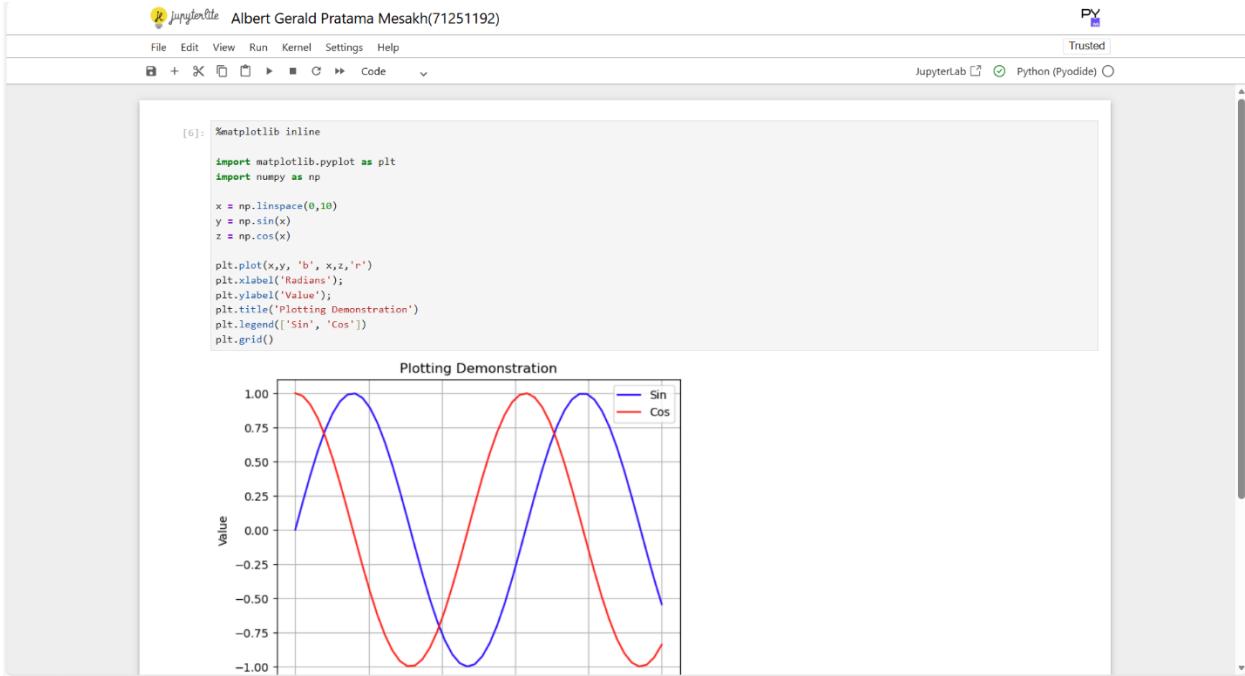
 **Project Jupyter. (2023).** *Jupyter Notebook Documentation*.

<https://jupyter.org/documentation>

Digunakan sebagai sumber penjelasan mengenai penggunaan Jupyter Notebook sebagai lingkungan interaktif untuk menjalankan Python.

BAGIAN 2: LATIHAN MANDIRI

SOAL 1



1. kita memanggil matplotlib untuk menggambar grafik
2. selanjutnya kita memanggil NumPy agar mengerjakan soal matematika dalam codingan
3. Setelah itu kita menyimpan data yang ada dalam soal, pada variable x kita membuat angka otomatis yang jaraknya sama seperti di atas, (gunakan linspace)
4. Variable y kita hitung dengan nilai sinus dan variable z kita hitung dengan nilai cosinus
5. Selanjutnya untuk gambar warna biru (variable x,y) dan warna merah (variable x,z)
6. Dan menambahkan garis radians untuk bagian horizontal, dan value untuk garis vertical.
7. Dan menambahkan kotak atau code (legend), untuk memberitau warna garis pada grafik
8. Dan grid untuk garis belakang Kotak-Kotak agar mudah dibaca

SOAL 2

1. Keuntungan yang didapat Gerard

```
[7]: gram_awal = 25
      harga_beli_awal = 650000
      modal_awal = gram_awal * harga_beli_awal
      modal_awal
      harga_sekarang = 685000
      nilai_sekarang = gram_awal * harga_sekarang
      nilai_sekarang
      keuntungan = nilai_sekarang - modal_awal
      keuntungan
      keuntungan_persen = (keuntungan / modal_awal) * 100
      keuntungan_persen
```

```
[7]: 5.384615384615385
```

- Variabel gram_awal adalah jumlah gram yang dimiliki.
- Variabel harga_beli_awal adalah harga beli per gram.
- Variabel modal_awal adalah total modal pembelian, yang merupakan hasil perkalian antara gram_awal dengan harga_beli_awal.
- Baris modal_awal untuk menampilkan nilai dari variabel modal_awal.
- Variabel harga_sekarang adalah harga per gram saat ini.
- Variabel nilai_sekarang adalah total nilai saat ini.
- Baris nilai_sekarang digunakan untuk menampilkan nilai dari variabel nilai_sekarang.
- Variabel keuntungan adalah selisih antara nilai_sekarang dan modal_awal, menunjukkan keuntungan dalam rupiah.
- Baris keuntungan digunakan untuk menampilkan nilai keuntungan.
- Variabel keuntungan_persen adalah persentase keuntungan, yang diperoleh dari hasil pembagian keuntungan dengan modal_awal, kemudian dikalikan 100.
- Baris keuntungan_persen digunakan untuk menampilkan nilai persentase keuntungan

1. Keuntungan yang didapat Gerard

2. Keuntungan yang didapat Gerard

```
[10]: gram_tambahan = 15
      harga_beli_kedua = 685000
      modal_kedua = gram_tambahan * harga_beli_kedua
      modal_kedua
      total_modal = modal_awal + modal_kedua
      total_modal
      total_gram = 40
      harga_terbaru = 715000
      nilai_total = total_gram * harga_terbaru
      nilai_total
      keuntungan_total = nilai_total - total_modal
      keuntungan_total
      keuntungan_total = nilai_total - total_modal
      keuntungan_total
```

```
[10]: 2075000
```

- Variabel gram_tambahan adalah jumlah gram tambahan yang dibeli.
- Variabel harga_beli_kedua adalah harga beli per gram pada pembelian kedua.
- Variabel modal_kedua adalah total modal dari pembelian kedua.
- Baris modal_kedua menampilkan nilai dari total modal pembelian kedua.
- Variabel total_modal adalah keseluruhan modal yang dikeluarkan.
- Baris total_modal digunakan untuk menampilkan nilai total modal.
- Variabel total_gram adalah jumlah keseluruhan gram yang dimiliki setelah pembelian tambahan.
- Variabel harga_terbaru adalah harga per gram terbaru saat ini.
- Variabel nilai_total adalah total nilai aset saat ini diperoleh dari hasil perkalian antara total_gram dan harga_terbaru.
- Baris nilai_total digunakan untuk menampilkan nilai total aset.
- Variabel keuntungan_total adalah total keuntungan keseluruhan, yang diperoleh dari hasil pengurangan nilai_total dengan total_modal.
- Baris keuntungan_total digunakan untuk menampilkan nilai keuntungan.

SOAL 3

```
[12]: # Modal awal dan target
modal = 200_000_000
target_uang = 400_000_000

# Suku bunga per tahun
suku_bunga = 0.10

from math import log

# Menghitung waktu (dalam Tahun)
waktu = log(target_uang / modal) / log(1 + suku_bunga)
print("Waktu yang dibutuhkan:", waktu, "tahun")

jumlah_tahun = int(waktu)
jumlah_bulan = (waktu - jumlah_tahun) * 12

jumlah_tahun, jumlah_bulan
```

Waktu yang dibutuhkan: 7.272540897341713 tahun

[12]: (7, 3.270490768100558)

1. modal = 200_000_000 menyatakan jumlah modal awal
2. target_uang = 400_000_000 menyatakan target dana
3. suku_bunga = 0.10 menyatakan tingkat bunga per tahun sebesar 10%
4. from math import log mengimpor fungsi log() dari library math.
5. waktu = log(target_uang / modal) / log(1 + suku_bunga) Variabel waktu digunakan untuk menghitung lama waktu yang dibutuhkan dalam tahun
6. print("Waktu yang dibutuhkan:", waktu, "tahun") menampilkan hasil waktu yang dibutuhkan untuk mencapai target
7. jumlah_tahun = int(waktu) menyimpan nilai tahun dalam bentuk bilangan bulat.
8. jumlah_bulan = (waktu - jumlah_tahun) * 12 menghitung sisa waktu dalam bulan.
9. jumlah_tahun, jumlah_bulan menampilkan hasil akhir yaitu 7,27 tahun.

Link Github: <https://github.com/geraldmesakh/Prak.Alpro-B.git>