

BAB III

GRAFIK DAN PYTHON SCRIPTS

I. TUJUAN

- Mahasiswa dapat menggunakan *library* Python untuk pembuatan grafik.
- Mahasiswa dapat membuat *Python Scripts* sederhana.
- Dapat menjalankan, membuat fungsi dan menjalankan fungsi dengan *Python Scripts*.
- Mahasiswa dapat membuat program kompleks menggunakan Python.

II. PENDAHULUAN

Grafik

Matplotlib adalah *library* Python untuk membuat visualisasi data. *Library* ini sangat fleksibel dan dapat digunakan untuk membuat berbagai jenis grafik, seperti grafik garis, batang, pie chart, histogram, scatter plot, dan grafik 3D. *Library* ini menyediakan kontrol penuh atas elemen-elemen grafik, seperti sumbu, label, grid, warna, legenda, dan gaya garis.

Dalam Python, *library* **Matplotlib** digunakan untuk menampilkan grafik dari suatu fungsi. Grafik 2 dimensi dapat dibuat menggunakan fungsi *plot()* dari modul *matplotlib.pyplot*. Berikut ini adalah cara untuk membuat grafik dan menambahkan berbagai keterangan:

- ***plt.plot()***, untuk membuat grafik.
- ***plt.title('teks')***, untuk menampilkan judul pada grafik.
- ***plt.xlabel('Sumbu X')***, untuk memberi label pada sumbu-x grafik.
- ***plt.ylabel('Sumbu Y')***, untuk memberi label pada sumbu-y grafik.
- ***plt.show()***, untuk menampilkan grafik yang telah dibuat.
- ***plt.text(koord1, koord2, 'teks')***, untuk menambahkan teks pada lokasi tertentu.

Dalam *library* **Matplotlib**, terdapat beberapa fungsi yang dapat digunakan untuk membuat berbagai jenis grafik diantaranya sebagai berikut:

- *plt.plot()*, secara *default* akan membuat sebuah *Line Plot*.
- *plt.scatter()*, untuk membuat grafik *Scatter Plot*.
- *plt.bar()*, untuk membuat grafik *Bar Plot*.
- *plt.hist()*, untuk membuat grafik *Histogram*.
- *plt.pie()*, untuk membuat grafik *Pie Chart*.

Matplotlib sangat fleksibel dan memungkinkan pengguna untuk menyesuaikan tampilan grafik sesuai kebutuhan, seperti mengubah warna grafik, mengatur gaya garis, menambahkan marker, mengatur lebar, dan mengatur ukuran. Berikut beberapa gaya warna dan garis di Matplotlib:

1. Warna

Nama Warna	Kode Singkat	Kode Hex
Biru	b	#0000FF
Hijau	g	#008000
Merah	r	#FF0000
Cyan	c	#00FFFF
Magenta	m	#FF00FF
Kuning	y	#FFFF00
Hitam	k	#000000
Putih	w	#FFFFFF
Abu-abu	gray	#808080

2. Gaya Garis

Deskripsi	Kode Gaya Garis
Garis lurus	'_'
Garis putus-putus	'--'
Garis titik-titik	'.'
Garis titik-garis	'-.'
Tidak ada garis	" atau ''

3. Marker

Nama Marker	Kode Marker
Tidak ada marker	None
Titik kecil	'.'
Lingkaran	'o'
Silang	'x'
Silang miring	'X'

Plus	'+'
Bintang	'*'
Persegi	's'
Persegi panjang	'P'
Segitiga ke atas	'^'
Segitiga ke bawah	'v'
Segitiga ke kiri	'<'
Segitiga ke kanan	'>'
Pentagon	'p'
Hexagon	'h'
Diamond (wajik)	'd'

Scripts

Script dalam Python adalah sekumpulan instruksi atau kode yang disimpan dalam file teks dengan ekstensi `.py`. File ini digunakan untuk menjalankan berbagai perintah secara langsung. Konsep script pada Python serupa dengan M-file di MATLAB, tetapi dengan beberapa perbedaan pada gaya penulisan dan cara menjalankannya. Berikut beberapa karakteristik dari script python:

- Semua script Python harus disimpan dengan ekstensi **.py**.
- Script dapat dijalankan melalui terminal atau command prompt dengan mengetik:

```
python nama_file.py
```
- Script juga dapat dijalankan menggunakan IDE seperti PyCharm, VSCode, atau Jupyter Notebook.
- Python tidak membutuhkan tanda titik koma (;) di akhir setiap perintah.
- Variabel yang dibuat di dalam script hanya akan tersedia selama script tersebut dijalankan. Setelah eksekusi selesai, variabel akan hilang kecuali disimpan dalam file eksternal atau database.

Di dalam script Python, terdapat elemen penting yang akan memiliki warna tertentu:

- Hijau menunjukkan komentar dalam kode.
- Biru menunjukkan kata kunci Python seperti **if**, **for**, dan **def**
- Hitam menunjukkan variabel atau perintah umum.

Script Python dengan Fungsi

Script Python dapat berisi fungsi yang dirancang untuk melakukan tugas tertentu. Fungsi ini membuat kode lebih modular, terorganisir, dan mudah digunakan kembali. Dalam fungsi Python, terdapat tiga komponen utama:

1. Parameter Input (*Input Arguments*): Argumen yang diberikan saat fungsi dipanggil untuk memproses data.
2. Proses (*Logic*): Kode atau algoritma yang menjalankan logika tertentu berdasarkan argumen input.
3. Parameter Output (*Output*): Nilai yang dihasilkan dari fungsi dan dapat digunakan lebih lanjut.

Fungsi Python didefinisikan menggunakan kata kunci `def`. Format umum penulisan fungsi adalah sebagai berikut:

```
def nama_fungsi(argumen):  
    """  
    Penjelasan fungsi ini.  
    """  
    # Logika atau proses  
    hasil = argumen ** 2  
  
    # Mengembalikan hasil  
    return hasil
```

III. LANGKAH PRAKTIKUM

Instalasi Library Matplotlib

Sebelum menggunakan library Matplotlib, kita perlu menginstal *library/package* tersebut dalam Python lokal kita dengan perintah sebagai berikut:

```
pip install matplotlib
```

Jika instalasi gagal, perbarui pip dengan perintah sebagai berikut:

```
pip install --upgrade pip
```

Membuat Grafik Sederhana

```
import matplotlib.pyplot as plt
x = [1, 2, 3]
y = [2, 4, 6]
plt.plot(x, y)
plt.show()
```

Menambahkan Judul & Label

```
x = [1, 2, 3]
y = [2, 4, 6]
plt.plot(x, y)
plt.title('Grafik Sederhana')
plt.xlabel('Sumbu X')
plt.ylabel('Sumbu Y')
plt.show()
```

Memberikan Warna Pada Grafik

```
# Menggunakan Nama Warna
plt.plot(x, y, color='red')
# Menggunakan Kode HEX
plt.plot(x, y, color='#1f77b4')
# Menggunakan RGB
plt.plot(x, y, color=(0.5, 0.2, 0.8))
```

Mengatur Gaya Garis

```
plt.plot(x, y, linestyle='--')
```

Menambahkan Marker

```
plt.plot(x, y, marker='o') # Lingkaran
plt.plot(x, y, marker='x') # Silang
```

Menggabungkan Gaya, Warna, dan Marker

```
plt.plot(x, y, 'r--o') # Garis merah, putus-putus, dengan marker
lingkaran
```

Mengatur Lebar dan Ukuran

```
plt.plot(x, y, color='blue', linestyle='-', marker='s', linewidth=2,
markersize=8)
```

Menampilkan Grafik Kurva $y = x^3$ dengan rentang $x=-3$ hingga $x=+3$.

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-3, 3.1, 0.1) # Inkremen 0.1 agar grafik terlihat mulus
y = x ** 3
plt.plot(x, y)
plt.xlabel('Sumbu X')
plt.ylabel('Sumbu Y')
plt.title('Kurva \ (Y = X^3\ )')
plt.grid(True)
plt.show()
```

Menampilkan Beberapa Kurva Dalam Satu Grafik

```
import matplotlib.pyplot as plt
import numpy as np

# Data untuk grafik
x = np.linspace(0, 10, 100) # Rentang nilai x dari 0 hingga 10

# Beberapa kurva
y1 = x                # Garis lurus
y2 = x**2             # Parabola
y3 = np.sqrt(x)       # Akar kuadrat
y4 = np.log1p(x)      # Logaritma natural (ln(1+x))

# Plot semua kurva
plt.plot(x, y1, label='y = x')          # Kurva 1
plt.plot(x, y2, label='y = x^2')        # Kurva 2
plt.plot(x, y3, label='y = √x')         # Kurva 3
plt.plot(x, y4, label='y = ln(1+x)')    # Kurva 4

# Menambahkan elemen grafik
plt.title('Beberapa Kurva dalam Satu Grafik')
plt.xlabel('Sumbu X')
plt.ylabel('Sumbu Y')
plt.legend() # Menampilkan legenda untuk setiap kurva
plt.grid(True) # Menampilkan grid
plt.show()
```

Membuat dan Menyimpan Script Python

```
# Program untuk menghitung rata-rata
a, b, c, d, e = 10, 20, 30, 40, 50
rata_rata = (a + b + c + d + e) / 5
print("Rata-rata:", rata_rata)
```

Simpan file dengan nama `hitung_rata2.py` dan jalankan di terminal IDE.

Menggunakan Fungsi dalam Script

```
def hitung_persegi_panjang(panjang, lebar):
    luas = panjang * lebar
    keliling = 2 * (panjang + lebar)
    return luas, keliling

panjang = 5
lebar = 3
luas, keliling = hitung_persegi_panjang(panjang, lebar)
print(f"Luas: {luas}, Keliling: {keliling}")
```

Menggunakan Input dari Pengguna

```
import math

jari_jari = float(input("Masukkan jari-jari lingkaran: "))
luas = math.pi * jari_jari ** 2
print(f"Luas lingkaran dengan jari-jari {jari_jari} adalah {luas:.2f}")
```

Menggunakan Script di Script Lain

File: `lingkaran_utils.py`

```
import math

def hitung_lingkaran(jari_jari):
    luas = math.pi * jari_jari ** 2
    keliling = 2 * math.pi * jari_jari
    return luas, keliling
```

Buat script baru untuk menggunakan fungsi ini, misalnya `script_utama.py`.

```
# Import fungsi dari circle_utils
from circle_utils import hitung_lingkaran

# Input jari-jari lingkaran
jari_jari = float(input("Masukkan jari-jari lingkaran: "))
```

```
# Menggunakan fungsi hitung_lingkaran
luas, keliling = hitung_lingkaran(jari_jari)

# Menampilkan hasil
print(f"Luas Lingkaran: {luas:.2f}")
print(f"Keliling Lingkaran: {keliling:.2f}")
```

Script dengan Banyak Fungsi

```
# bangun_datar.py

import math

def hitung_persegi_panjang(panjang, lebar):
    luas = panjang * lebar
    keliling = 2 * (panjang + lebar)
    return luas, keliling

def hitung_lingkaran(jari_jari):
    luas = math.pi * jari_jari ** 2
    keliling = 2 * math.pi * jari_jari
    return luas, keliling

def faktorial(n):
    if n < 0:
        raise ValueError("Faktorial hanya untuk bilangan bulat positif.")
    hasil = 1
    for i in range(1, n + 1):
        hasil *= i
    return hasil
```

Buat script baru, misalnya script_utama.py, dan impor fungsi dari bangun_datar.py.

```
from math_utils import hitung_persegi_panjang, hitung_lingkaran,
faktorial, hitung_hipotenusa

panjang = 10
lebar = 5
luas, keliling = hitung_persegi_panjang(panjang, lebar)
print(f"Luas Persegi Panjang: {luas}, Keliling: {keliling}")

jari_jari = 7
luas, keliling = hitung_lingkaran(jari_jari)
print(f"Luas Lingkaran: {luas:.2f}, Keliling: {keliling:.2f}")

n = 5
print(f"Faktorial dari {n}: {faktorial(n)}")
```


IV. TUGAS PRAKTIKUM

1. Buat grafik bar untuk menampilkan jumlah penjualan barang selama 5 bulan dengan data berikut:

- Bulan: Januari, Februari, Maret, April, Mei
- Penjualan: 150, 200, 180, 220, 170

Berikan judul grafik, label sumbu X (Bulan), dan label sumbu Y (Jumlah Penjualan).

2. Dari data yang sama, buatlah diagram pie untuk menunjukkan persentase kontribusi penjualan tiap bulan. Pastikan setiap bagian memiliki label bulan.
3. Buat grafik scatter plot untuk menampilkan hubungan antara usia dan penghasilan seseorang berdasarkan data berikut:
 - Usia: [20, 25, 30, 35, 40, 45, 50]
 - Penghasilan: [2000, 3200, 4500, 5500, 6100, 5800, 5000]

Tambahkan:

- Warna marker hijau ('g') dan bentuk lingkaran ('o').
- Judul grafik "Hubungan Usia dan Penghasilan".
- Label sumbu X dengan "Usia (Tahun)" dan sumbu Y dengan "Penghasilan (Ribu USD)".

4. Untuk absen Ganjil

Buatlah suatu script python untuk menghitung volume dan luas permukaan balok dengan spesifikasi:

masukan fungsi : panjang, lebar, dan tinggi balok

keluaran fungsi : volume, dan luas permukaan balok.

Beri nama fungsi ini dengan `hitung_balok.py`

Untuk absen Genap

Buatlah suatu script python untuk menghitung volume dan luas permukaan dari suatu prisma segiempat dengan spesifikasi:

masukan fungsi : panjang dan lebar alas prisma, serta tinggi prisma

keluaran fungsi : volume, dan luas permukaan prisma

Beri nama fungsi ini dengan `hitung_prisma.m`

5. Buatlah sebuah fungsi untuk menghitung jumlah hari di antara dua tanggal. Spesifikasi dari fungsi tersebut ialah:

masukan : tanggal, bulan, dan tahun awal, serta tanggal, bulan, dan tahun akhir.

keluaran : jumlah hari di antara dua tanggal tersebut.

Beri nama fungsi ini dengan `hitung_hari.py`.

Misalkan akan menghitung jumlah hari antara 2 Januari 2004 hingga 5 November 2006, maka tulis:

```
jml_hari = hitung_hari(2,1,2004,5,11,2006)
jml_hari = 1038
```