# LAPORAN PRAKTIKUM

## GRAFIKA KOMPUTER

### **(Dosen : *Rio Priantama S.T., M.T.I.*)**

**Modul 2**



**Nama : Muhammad Rizal Nurfirdaus**

**NIM : 20230810088**

**Kelas : TINFC-2023-04**

# TEKNIK INFORMATIKA

## FAKULTAS ILMU KOMPUTER

**UNIVERSITAS KUNINGAN**

**Praktikum**

Praktikum : Menggambar Kotak

1. Praktikum 1  
   Source Code:

import matplotlib.pyplot as plt

import matplotlib.patches as patches

# Membuat sebuah figure dan axis

fig, ax = plt.subplots()

# Fungsi untuk menggambar kotak

def gambar\_kotak(ukuran):

    """

    Menggambar kotak dengan panjang sisi yang ditentukan oleh parameter 'ukuran'.

    Parameter:

        ukuran (int): Panjang sisi kotak.

    """

    # Menggambar persegi dengan ukuran tertentu

    kotak = patches.Rectangle((0, 0), ukuran, ukuran, edgecolor='blue', facecolor='none', linewidth=2)

    ax.add\_patch(kotak)

    # Mengatur batas-batas tampilan sumbu agar sesuai dengan ukuran kotak

    ax.set\_xlim(-10, ukuran + 10)

    ax.set\_ylim(-10, ukuran + 10)

    # Menampilkan grid dan kotak

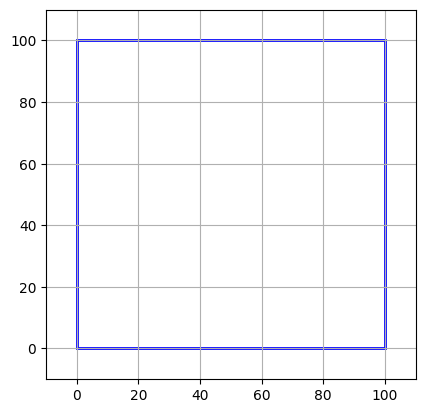
    plt.grid(True)

    plt.gca().set\_aspect('equal', adjustable='box')

    plt.show()

# Memanggil fungsi untuk menggambar kotak dengan ukuran sisi 100 unit

gambar\_kotak(100)

Hasil Run:  


Analisis: Kode di atas merupakan program Python yang menggunakan pustaka **matplotlib** untuk menggambar sebuah **persegi** dengan panjang sisi tertentu. Program dimulai dengan membuat *figure* dan *axis* menggunakan plt.subplots(). Selanjutnya, fungsi gambar\_kotak(ukuran) didefinisikan untuk menggambar persegi berdasarkan parameter ukuran, yang menentukan panjang sisi persegi. Di dalam fungsi tersebut, digunakan patches.Rectangle((0, 0), ukuran, ukuran, ...) untuk membuat persegi dengan titik awal di koordinat (0, 0). Warna garis tepi persegi ditetapkan menjadi biru (edgecolor='blue'), tanpa warna isi (facecolor='none'), dan dengan ketebalan garis 2 (linewidth=2). Kemudian, batas tampilan sumbu X dan Y diatur agar persegi terlihat jelas dengan menambahkan jarak 10 unit dari tepi melalui ax.set\_xlim dan ax.set\_ylim. Grid diaktifkan menggunakan plt.grid(True), dan rasio aspek diatur agar proporsional menggunakan set\_aspect('equal'), sehingga persegi tidak tampak terdistorsi. Akhirnya, plt.show() digunakan untuk menampilkan hasil gambar. Saat fungsi dipanggil dengan gambar\_kotak(100), program akan menampilkan sebuah persegi berukuran **100 unit** dengan garis tepi berwarna biru pada bidang koordinat.

1. Praktikum 2 : Membuat Lingkaran  
   Source Code:

import matplotlib.pyplot as plt

import matplotlib.patches as patches

# Membuat sebuah figure dan axis

fig, ax = plt.subplots()

# Fungsi untuk menggambar lingkaran

def gambar\_lingkaran(radius):

    """

    Menggambar lingkaran dengan radius yang ditentukan oleh parameter 'radius'.

    Parameter:

        radius (int): Radius lingkaran.

    """

    # Menggambar lingkaran dengan radius tertentu

    lingkaran = patches.Circle((0, 0), radius, edgecolor='blue', facecolor='none', linewidth=2)

    ax.add\_patch(lingkaran)

    # Mengatur batas-batas tampilan sumbu agar sesuai dengan radius lingkaran

    ax.set\_xlim(-radius - 10, radius + 10)

    ax.set\_ylim(-radius - 10, radius + 10)

    # Menampilkan grid dan lingkaran

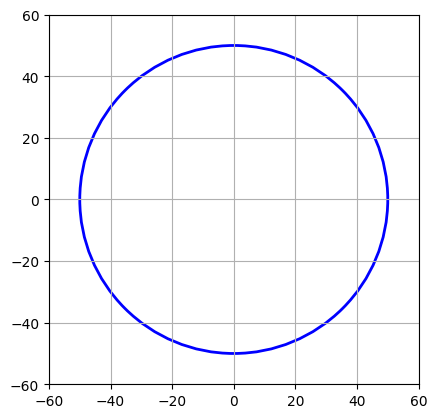
    plt.grid(True)

    plt.gca().set\_aspect('equal', adjustable='box')

    plt.show()

# Memanggil fungsi untuk menggambar lingkaran dengan radius 50 unit

gambar\_lingkaran(50)

Hasil Run:  


Analisis: Kode di atas merupakan program Python yang menggunakan pustaka **matplotlib** untuk menggambar sebuah **lingkaran** dengan jari-jari tertentu. Program dimulai dengan membuat *figure* dan *axis* menggunakan plt.subplots(). Selanjutnya, didefinisikan fungsi gambar\_lingkaran(radius) yang berfungsi untuk menggambar lingkaran berdasarkan nilai parameter radius. Di dalam fungsi ini, objek patches.Circle((0, 0), radius, ...) digunakan untuk membuat lingkaran dengan pusat di titik (0, 0) dan jari-jari sebesar nilai radius. Warna tepi lingkaran diatur menjadi biru (edgecolor='blue'), tanpa warna isi (facecolor='none'), dan dengan ketebalan garis sebesar 2 (linewidth=2). Setelah lingkaran ditambahkan ke bidang gambar menggunakan ax.add\_patch(lingkaran), batas tampilan sumbu X dan Y diatur dengan menambahkan jarak 10 unit di luar radius agar lingkaran terlihat utuh menggunakan ax.set\_xlim dan ax.set\_ylim. Selain itu, plt.grid(True) digunakan untuk menampilkan garis grid, dan plt.gca().set\_aspect('equal', adjustable='box') memastikan bahwa sumbu X dan Y memiliki skala yang sama sehingga lingkaran tidak tampak elips. Terakhir, plt.show() digunakan untuk menampilkan hasil gambar. Saat fungsi dipanggil dengan gambar\_lingkaran(50), program akan menampilkan sebuah **lingkaran dengan jari-jari 50 unit** dan garis tepi berwarna biru secara proporsional pada bidang koordinat.

1. Praktikum 3 : Membuat Segitiga  
   Source Code:

import matplotlib.pyplot as plt

import matplotlib.patches as patches

# Membuat sebuah figure dan axis

fig, ax = plt.subplots()

# Fungsi untuk menggambar segitiga

def gambar\_segitiga(panjang\_sisi):

    # Menghitung koordinat titik-titik segitiga

    t1 = (0, 0)

    t2 = (panjang\_sisi, 0)

    t3 = (panjang\_sisi / 2, (panjang\_sisi \* (3 \*\* 0.5)) / 2) # Tinggi segitiga sama sisi

    # Membuat objek segitiga dengan koordinat yang dihitung

    segitiga = patches.Polygon([t1, t2, t3], closed=True, edgecolor='blue', facecolor='none', linewidth=2)

    ax.add\_patch(segitiga)

    # Mengatur batas-batas tampilan sumbu agar sesuai dengan ukuran segitiga

    ax.set\_xlim(-10, panjang\_sisi + 10)

    ax.set\_ylim(-10, (panjang\_sisi \* (3 \*\* 0.5)) / 2 + 10)

    # Menampilkan grid dan segitiga

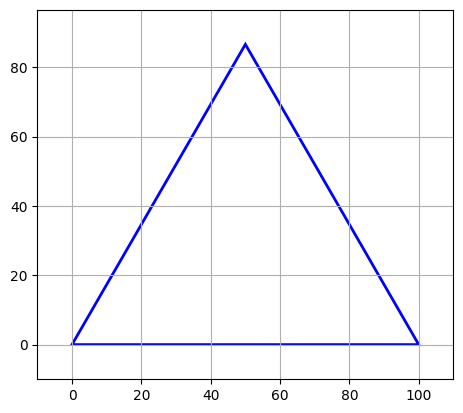
    plt.grid(True)

    plt.gca().set\_aspect('equal', adjustable='box')

    plt.show()

# Memanggil fungsi untuk menggambar segitiga dengan panjang sisi 100 unit

gambar\_segitiga(100)

Hasil Run:  


Analisis: Kode di atas merupakan program Python yang menggunakan pustaka **matplotlib** untuk menggambar sebuah **segitiga sama sisi** dengan panjang sisi tertentu. Program diawali dengan membuat *figure* dan *axis* menggunakan plt.subplots(). Kemudian, fungsi gambar\_segitiga(panjang\_sisi) didefinisikan untuk menggambar segitiga berdasarkan nilai parameter panjang\_sisi. Di dalam fungsi tersebut, tiga titik utama segitiga dihitung: titik pertama (0, 0) sebagai alas kiri, titik kedua (panjang\_sisi, 0) sebagai alas kanan, dan titik ketiga (panjang\_sisi / 2, (panjang\_sisi \* (3 \*\* 0.5)) / 2) sebagai puncak segitiga, di mana rumus (panjang\_sisi \* √3) / 2 digunakan untuk menentukan tinggi segitiga sama sisi. Selanjutnya, objek segitiga dibuat menggunakan patches.Polygon() dengan parameter koordinat ketiga titik tersebut. Warna tepi segitiga diatur menjadi biru (edgecolor='blue'), tanpa warna isi (facecolor='none'), dan dengan ketebalan garis 2 (linewidth=2). Setelah itu, segitiga ditambahkan ke bidang gambar menggunakan ax.add\_patch(segitiga). Batas tampilan sumbu X dan Y disesuaikan agar segitiga terlihat proporsional dengan menambahkan jarak 10 unit di sekelilingnya. Grid diaktifkan melalui plt.grid(True), dan aspek rasio dibuat sama (set\_aspect('equal')) agar bentuk segitiga tidak terdistorsi. Terakhir, plt.show() digunakan untuk menampilkan hasil gambar. Saat fungsi dipanggil dengan gambar\_segitiga(100), program akan menampilkan sebuah **segitiga sama sisi berukuran 100 unit** dengan garis tepi berwarna biru pada bidang koordinat.

1. Praktikum 4 : Membuat Persegi Panjang  
   Source Code:

import matplotlib.pyplot as plt

import matplotlib.patches as patches

# Membuat sebuah figure dan axis

fig, ax = plt.subplots()

# Fungsi untuk menggambar persegi panjang

def gambar\_persegi\_panjang(lebar, tinggi):

    """

    Menggambar persegi panjang dengan lebar dan tinggi yang ditentukan.

    Parameter:

        lebar (int): Lebar persegi panjang.

        tinggi (int): Tinggi persegi panjang.

    """

    # Membuat objek persegi panjang dengan lebar dan tinggi tertentu

    persegi\_panjang = patches.Rectangle((0, 0), lebar, tinggi, edgecolor='blue', facecolor='none', linewidth=2)

    ax.add\_patch(persegi\_panjang)

    # Mengatur batas-batas tampilan sumbu agar sesuai dengan ukuran persegi panjang

    ax.set\_xlim(-10, lebar + 10)

    ax.set\_ylim(-10, tinggi + 10)

    # Menampilkan grid dan persegi panjang

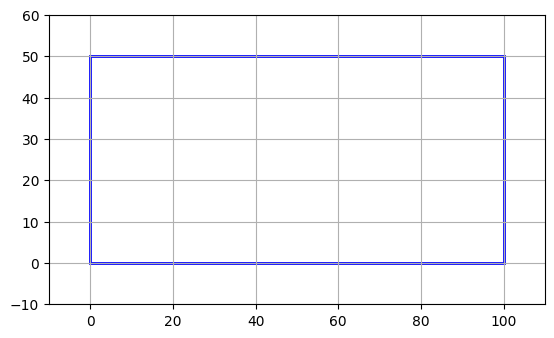
    plt.grid(True)

    plt.gca().set\_aspect('equal', adjustable='box')

    plt.show()

# Memanggil fungsi untuk menggambar persegi panjang dengan lebar 100 unit dan tinggi 50 unit

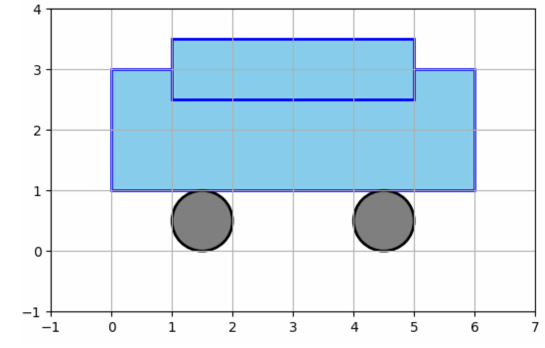
gambar\_persegi\_panjang(100, 50)

Hasil Run :  
 

Analisis: Kode di atas merupakan program Python yang menggunakan pustaka **matplotlib** untuk menggambar sebuah **persegi panjang** dengan ukuran lebar dan tinggi yang dapat ditentukan. Program dimulai dengan membuat *figure* dan *axis* menggunakan plt.subplots(). Selanjutnya, fungsi gambar\_persegi\_panjang(lebar, tinggi) didefinisikan untuk menggambar persegi panjang berdasarkan parameter lebar dan tinggi. Di dalam fungsi tersebut, objek persegi panjang dibuat menggunakan patches.Rectangle((0, 0), lebar, tinggi, ...), di mana titik awal berada pada koordinat (0, 0), dan persegi panjang terbentuk dengan ukuran sesuai nilai parameter. Warna garis tepi diatur menjadi biru (edgecolor='blue'), tanpa warna isi (facecolor='none'), serta memiliki ketebalan garis sebesar 2 (linewidth=2). Setelah objek ditambahkan ke bidang gambar menggunakan ax.add\_patch(persegi\_panjang), batas tampilan sumbu X dan Y diatur agar persegi panjang terlihat proporsional dan tidak terpotong, dengan memberikan jarak tambahan 10 unit pada setiap sisi melalui ax.set\_xlim dan ax.set\_ylim. Garis grid diaktifkan menggunakan plt.grid(True), dan aspek rasio diatur sama dengan plt.gca().set\_aspect('equal', adjustable='box') agar bentuk persegi panjang tampak sesuai skala sebenarnya. Terakhir, plt.show() digunakan untuk menampilkan hasil gambar. Ketika fungsi dipanggil dengan gambar\_persegi\_panjang(100, 50), program akan menampilkan **persegi panjang berukuran lebar 100 unit dan tinggi 50 unit** dengan garis tepi berwarna biru pada bidang koordinat.

**LATIHAN/TUGAS**

1. Buatlah kode program sederhana untuk menggambar mobil seperti gamabr berikut:

  
Source Code:

# Tugas 1

import matplotlib.pyplot as plt

import matplotlib.patches as patches

# Membuat figure dan axis untuk menggambar

fig, ax = plt.subplots(figsize=(8, 6)) # Ukuran figure agar lebih proporsional

# --- Menggambar Bagian Badan Mobil (Persegi Panjang Besar) ---

# Dimulai dari (x, y) = (0, 1)

# Lebar = 6, Tinggi = 2

badan\_mobil = patches.Rectangle(

    (0, 1),

    6,

    2,

    edgecolor='blue',

    facecolor='skyblue', # Menggunakan skyblue seperti pada gambar

    linewidth=2

)

ax.add\_patch(badan\_mobil)

# --- Menggambar Bagian Kabin Mobil (Persegi Panjang Kecil di Atas) ---

# Dimulai dari (x, y) = (1.5, 2.5)

# Lebar = 3, Tinggi = 1

kabin\_mobil = patches.Rectangle(

    (1.5, 2.5),

    3,

    1,

    edgecolor='blue',

    facecolor='skyblue', # Menggunakan skyblue

    linewidth=2

)

ax.add\_patch(kabin\_mobil)

# --- Menggambar Roda Kiri (Lingkaran) ---

# Pusat = (1.5, 0.5), Radius = 0.5

roda\_kiri = patches.Circle(

    (1.5, 0.5),

    0.5,

    edgecolor='black',

    facecolor='dimgrey', # Abu-abu gelap untuk roda

    linewidth=2

)

ax.add\_patch(roda\_kiri)

# --- Menggambar Roda Kanan (Lingkaran) ---

# Pusat = (4.5, 0.5), Radius = 0.5

roda\_kanan = patches.Circle(

    (4.5, 0.5),

    0.5,

    edgecolor='black',

    facecolor='dimgrey',

    linewidth=2

)

ax.add\_patch(roda\_kanan)

# --- Pengaturan Tampilan Plot ---

# Mengatur batas-batas sumbu X dan Y agar sesuai dengan gambar dan grid

ax.set\_xlim(-1, 7) # Dari -1 sampai 7 seperti pada gambar

ax.set\_ylim(-1, 4) # Dari -1 sampai 4 seperti pada gambar

# Menampilkan grid

ax.grid(True, linestyle='-', linewidth=0.5, color='lightgrey') # Grid yang lebih halus

# Mengatur aspek rasio agar sumbu X dan Y memiliki skala yang sama

# Ini penting agar lingkaran dan persegi panjang terlihat proporsional

ax.set\_aspect('equal', adjustable='box')

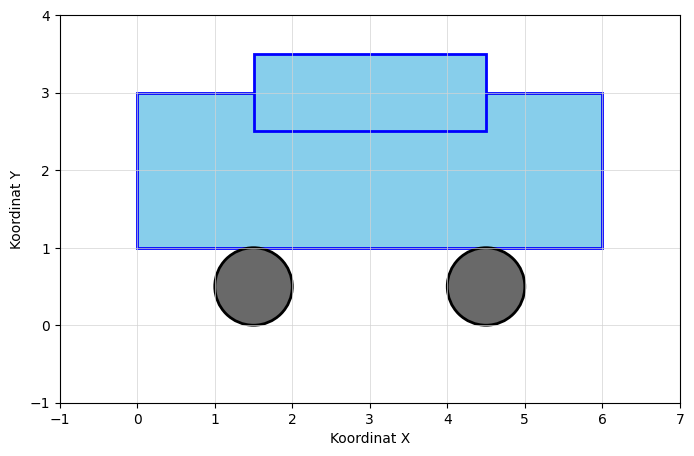
# Menambahkan label sumbu

plt.xlabel("Koordinat X")

plt.ylabel("Koordinat Y")

# Menampilkan plot

plt.show()

Hasil Run :  


Analisis: Kode di atas merupakan program Python yang menggunakan pustaka **matplotlib** untuk menggambar sebuah **mobil sederhana dua dimensi** menggunakan kombinasi bentuk dasar seperti **persegi panjang** dan **lingkaran**. Program diawali dengan pembuatan *figure* dan *axis* berukuran 8×6 inci agar gambar terlihat proporsional. Selanjutnya, bagian **badan mobil** dibuat dengan objek persegi panjang berwarna biru muda menggunakan patches.Rectangle((0, 1), 6, 2, ...), yang merepresentasikan bagian utama mobil. Di atasnya, dibuat **kabin mobil** yang juga berbentuk persegi panjang dengan ukuran lebih kecil ((1.5, 2.5), 3, 1), menggambarkan bagian tempat pengemudi.

Selain itu, dua **roda mobil** digambar menggunakan patches.Circle() dengan pusat di koordinat (1.5, 0.5) dan (4.5, 0.5), masing-masing memiliki jari-jari 0.5 dan diberi warna abu-abu gelap agar tampak seperti roda asli. Setelah semua komponen ditambahkan, tampilan sumbu diatur agar proporsional melalui ax.set\_xlim, ax.set\_ylim, dan ax.set\_aspect('equal'), sementara grid ditampilkan agar posisi objek lebih jelas. Akhirnya, program menampilkan visualisasi mobil sederhana dengan proporsi yang seimbang dan elemen-elemen yang tersusun rapi pada bidang koordinat, lengkap dengan warna, garis tepi, dan label sumbu yang membuat tampilannya informatif serta mudah dipahami.