Tugas grafika computer

Nama : iwan sugiwan

Nim :2211310111

|  |  |
| --- | --- |
|  | 2 |

Gambar lapangan tenis meja

Keterangan;

1.blade no 1(bulat lingkaran merah no 1)

2. blade no 2( bulat lingakaran merah no 2)

3. bola bulat putih

Coding sederhana:

import pygame

import sys

# Inisialisasi Pygame

pygame.init()

# Ukuran layar

WIDTH, HEIGHT = 800, 400

screen = pygame.display.set\_mode((WIDTH, HEIGHT))

pygame.display.set\_caption("Ping Pong Sederhana")

# Warna

BLUE = (0, 0, 255)

RED = (255, 0, 0)

WHITE = (255, 255, 255)

YELLOW = (255, 255, 0)

# Posisi dan ukuran objek

blade1\_pos = [50, HEIGHT // 2 - 50]

blade2\_pos = [WIDTH - 60, HEIGHT // 2 - 50]

blade\_size = [10, 100]

ball\_pos = [WIDTH // 2, HEIGHT // 2]

ball\_radius = 10

ball\_speed = [4, 4]  # Kecepatan bola (x, y)

# Kecepatan blade

blade\_speed = 5

# Loop utama

clock = pygame.time.Clock()

while True:

    screen.fill(BLUE)

    # Event handler

    for event in pygame.event.get():

        if event.type == pygame.QUIT:

            pygame.quit()

            sys.exit()

    # Gerakkan blade 1 (kiri) dengan mouse

    mouse\_y = pygame.mouse.get\_pos()[1]

    blade1\_pos[1] = mouse\_y - blade\_size[1] // 2

    # Gerakkan blade 2 (kanan) otomatis mengikuti bola

    if ball\_pos[1] > blade2\_pos[1] + blade\_size[1] // 2:

        blade2\_pos[1] += blade\_speed

    elif ball\_pos[1] < blade2\_pos[1] + blade\_size[1] // 2:

        blade2\_pos[1] -= blade\_speed

    # Batasi gerakan blade agar tidak keluar layar

    blade1\_pos[1] = max(0, min(HEIGHT - blade\_size[1], blade1\_pos[1]))

    blade2\_pos[1] = max(0, min(HEIGHT - blade\_size[1], blade2\_pos[1]))

    # Pergerakan bola

    ball\_pos[0] += ball\_speed[0]

    ball\_pos[1] += ball\_speed[1]

    # Pantulan bola di atas dan bawah layar

    if ball\_pos[1] - ball\_radius <= 0 or ball\_pos[1] + ball\_radius >= HEIGHT:

        ball\_speed[1] = -ball\_speed[1]

    # Pantulan bola di blade 1

    if (

        ball\_pos[0] - ball\_radius <= blade1\_pos[0] + blade\_size[0]

        and blade1\_pos[1] < ball\_pos[1] < blade1\_pos[1] + blade\_size[1]

    ):

        ball\_speed[0] = -ball\_speed[0]

    # Pantulan bola di blade 2

    if (

        ball\_pos[0] + ball\_radius >= blade2\_pos[0]

        and blade2\_pos[1] < ball\_pos[1] < blade2\_pos[1] + blade\_size[1]

    ):

        ball\_speed[0] = -ball\_speed[0]

    # Reset bola jika keluar layar

    if ball\_pos[0] - ball\_radius <= 0 or ball\_pos[0] + ball\_radius >= WIDTH:

        ball\_pos = [WIDTH // 2, HEIGHT // 2]

        ball\_speed = [4, 4]

    # Gambar net

    pygame.draw.line(screen, WHITE, (WIDTH // 2, 0), (WIDTH // 2, HEIGHT), 2)

    # Gambar blade dan bola

    pygame.draw.rect(screen, RED, (\*blade1\_pos, \*blade\_size))

    pygame.draw.rect(screen, YELLOW, (\*blade2\_pos, \*blade\_size))

    pygame.draw.circle(screen, WHITE, ball\_pos, ball\_radius)

    # Perbarui layar

    pygame.display.flip()

    clock.tick(60)

**Penjelasan Kode:**

1. **Blade No. 1**:
   * Digerakkan dengan mouse (posisi Y mouse menentukan posisi blade).
2. **Blade No. 2**:
   * Bergerak otomatis mengikuti posisi bola.
3. **Bola**:
   * Memantul jika menyentuh blade atau dinding atas/bawah.
   * Kembali ke tengah jika keluar dari sisi kiri/kanan.
4. **Net**:
   * Digambar sebagai garis putih di tengah layar.

Animasi pingpong sederhana interaksi dengan mose di mana "Blade No 1" dan "Blade No 2" dapat digerakkan menggunakan mouse, blade no 1 menyentuh bola akan memantul meyebrang net saat bersentuhan dengan blade no 2 ( pergerakan bola lurus dari sisi kiri ke kanan) [C:\New folder (2)\Animasi interaksi mose pingpong (2).py](animasi)