

## Modul 4

### A. Tujuan

- Mahasiswa mampu memahami dan mengimplementasikan fungsi input/output dalam pygame
- Mahasiswa mampu memahami dan mengimplementasikan kontrol mouse dan keyboard

### B. Dasar teori

#### 1. Implementing mouse and keyboard control

Modul `pygame.locals` berisi sekitar 280 konstanta yang dapat digunakan pygame.

```
import pygame
from pygame.locals import *
```

We find the key modifiers (alt, ctrl, cmd, etc.)

```
KMOD_ALT, KMOD_CAPS, KMOD_CTRL, KMOD_LALT,
KMOD_LCTRL, KMOD_LMETA, KMOD_LSHIFT, KMOD_META,
KMOD_MODE, KMOD_NONE, KMOD_NUM, KMOD_RALT, KMOD_RCTRL,
KMOD_RMETA, KMOD_RSHIFT, KMOD_SHIFT,
```

the number keys:

```
K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_8, K_9,
```

the special character keys:

```
K_AMPERSAND, K_ASTERISK, K_AT, K_BACKQUOTE,
K_BACKSLASH, K_BACKSPACE, K_BREAK,
```

the letter keys of the alphabet:

```
K_a, K_b, K_c, K_d, K_e, K_f, K_g, K_h, K_i, K_j, K_k, K_l, K_m,
K_n, K_o, K_p, K_q, K_r, K_s, K_t, K_u, K_v, K_w, K_x, K_y, K_z,
```

#### 2. Text in Pygame

- Untuk menampilkan pygame text, dapat digunakan modul `pygame.font`.

```
font_color=(0,150,250)
font_obj=pygame.font.Font("C:\Windows\Fonts\segoeprb.ttf",25)
text_obj=font_obj.render("Welcome to Pygame",True,font_color)
```

```
sur_obj.fill((255,255,255))
sur_obj.blit(text_obj,(22,0))
```

Hasil :



Penjelasan program :

Fungsi `pygame.font.Font` (“(direktori font”, `font_size`) memiliki 2 parameter. Parameter pertama merupakan tempat (direktori) font tersimpan, sedangkan parameter kedua adalah ukuran font.

Fungsi `render` (`Text`, `antialiasing`, `font_color`, `background=None`) memiliki 4 parameter. Parameter pertama berisi `text` yang akan di tampilkan pada `pygame` windows. Parameter kedua adalah `antialiasing` memiliki nilai `True/False`. `Antialiasing` bernilai `True` jika ingin memiliki tepi yang halus, `False` jika memiliki tepi yang kasar. Parameter ketiga merupakan warna karakter. Parameter keempat adalah untuk menambahkan background

Fungsi `Fill` untuk mengatur warna background. Sedangkan fungsi `blit` untuk merender (menampilkan) teks pada layar sesuai dengan koordinat yang ditentukan.

- Mengedit Text menggunakan keyboard

```
import pygame
from pygame.locals import *
import time

BLACK = (0,0,0)
RED = (255,0,0)
GRAY = (200,200,200)

pygame.init()
screen = pygame.display.set_mode((640,240))

text = "game development"
font = pygame.font.SysFont(None, 48)
img = font.render(text, True, RED)

rect = img.get_rect()
rect.topleft = (20,20)
cursor = Rect(rect.topright, (3, rect.height))

running = True
background = GRAY

while running:
    for event in pygame.event.get():
        if event.type == QUIT:
            running = False

        if event.type == KEYDOWN:
            if event.key == K_BACKSPACE:
                if len(text)>0:
                    text = text[:-1]

            else:
                text += event.unicode
            img = font.render(text, True, RED)
            rect.size = img.get_size()
            cursor.topleft = rect.topright

    screen.fill(background)
    screen.blit(img,rect)
    if time.time() % 1 > 0.5:
        pygame.draw.rect(screen, RED, cursor)
    pygame.display.update()

pygame.quit()
```

Hasil program :



## C. Contoh Program

- Smooth Movement

### Part A

```
def update(self):
    self.velX = 0
    self.velY = 0
    if self.left_pressed and not self.right_pressed:
        self.velX = -self.speed
    if self.right_pressed and not self.left_pressed:
        self.velX = self.speed
    if self.up_pressed and not self.down_pressed:
        self.velY = -self.speed
    if self.down_pressed and not self.up_pressed:
        self.velY = self.speed

    self.x += self.velX
    self.y += self.velY

    self.rect = pygame.Rect(int(self.x), int(self.y), 32, 32)
```

### Part B

```
player = Player(WIDTH/2, HEIGHT/2)

while True:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()
            sys.exit()
        if event.type == pygame.KEYDOWN:
            if event.key == pygame.K_LEFT:
                player.left_pressed = True
            if event.key == pygame.K_RIGHT:
                player.right_pressed = True
            if event.key == pygame.K_UP:
                player.up_pressed = True
            if event.key == pygame.K_DOWN:
                player.down_pressed = True
        if event.type == pygame.KEYUP:
            if event.key == pygame.K_LEFT:
                player.left_pressed = False
            if event.key == pygame.K_RIGHT:
                player.right_pressed = False
            if event.key == pygame.K_UP:
                player.up_pressed = False
            if event.key == pygame.K_DOWN:
                player.down_pressed = False
```

### Part C

```
win.fill((12, 24, 36))
player.draw(win)

player.update()
pygame.display.flip()

clock.tick(120)
```

### Part D

```
class Player:
    def __init__(self, x, y):
        self.x = int(x)
        self.y = int(y)
        self.rect = pygame.Rect(self.x, self.y, 32, 32)
        self.color = (250, 120, 60)
        self.velX = 0
        self.velY = 0
        self.left_pressed = False
        self.right_pressed = False
        self.up_pressed = False
        self.down_pressed = False
        self.speed = 4
```

### Part E

```
import pygame, sys

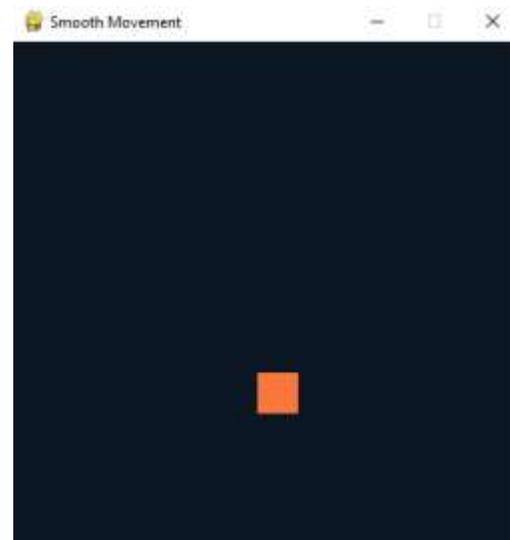
WIDTH, HEIGHT = 400, 400
TITLE = "Smooth Movement"

pygame.init()
win = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption(TITLE)
clock = pygame.time.Clock()
```

### Part F

```
def draw(self, win):
    pygame.draw.rect(win, self.color, self.rect)
```

## Hasil Program



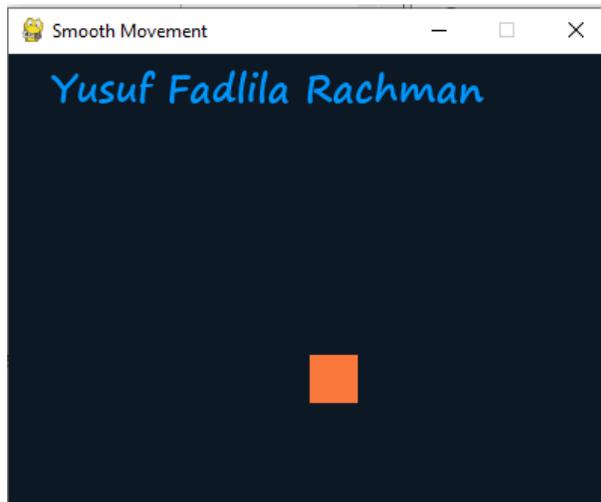
Penjelasan program :

Jika dijalankan, maka program akan menerima inputan dari keyboard. Inputan yang dibuat adalah key\_up, key\_down, key\_left, key\_right. Objek yang ditampilkan pada pygame window (kotak) dapat bergerak sesuai dengan inputan yang diberikan. Jika ditekan tombol atas, maka kotak akan bergerak keatas. Jika ditekan tombol bawah, kotak bergerak kebawah begitu seterusnya.

#### **D. TUGAS INDIVIDU**

1. Cobalah program pada poin C. Kode program pada poin C terdiri dari beberapa bagian (6 part) acak. Susun bagian-bagian kode tersebut sehingga dapat menampilkan hasil program seperti pada contoh. Kemudian:

- Ubah objek yang ditampilkan, sehingga dapat menampilkan karakter game yang dibuat pada mata kuliah GameDev.
- Tampilkan sebuah text isinya “Nama Lengkap anda” pada pygame windows.
- Berikan fungsi pembatas pada pergerakan objek yang dibuat, sehingga objek tersebut tidak bisa bergerak melebihi ukuran panjang dan lebar dari pygame windows.
- Contoh hasil :



## 2. Pengumpulan Tugas Praktikum.

- Tempatkan kode program, penjelasan tiap bagian program (penjelasan part A, part B, dst), dan screenshot hasil program di akun github masing-masing. Kemudian kumpulkan link akun github di SPADA
- **Untuk kelas TI E** Tugas Praktikum di kumpulkan paling lambat tanggal **21 September 2021 jam 23.59**  
**Untuk kelas TI D** Tugas Praktikum di kumpulkan paling lambat tanggal **22 September 2021 jam 23.59**
- Format penamaan file SKD\_namakelas\_nim\_nama