import pygame

import pygame\_gui

pygame.init()

# Initialize Pygame

pygame.display.set\_caption('Main Game Window')

window\_size = (800, 600)

window = pygame.display.set\_mode(window\_size)

# Initialize a separate window for the control panel

panel\_size = (400, 200)

panel\_window = pygame.display.set\_mode(panel\_size)

pygame.display.set\_caption('Control Panel')

panel\_manager = pygame\_gui.UIManager(panel\_size)

# Your Pygmy program variables

default\_variable1 = 50

default\_variable2 = 100

variable1 = default\_variable1

variable2 = default\_variable2

# Create sliders in the separate panel

slider1 = pygame\_gui.elements.UIHorizontalSlider(relative\_rect=pygame.Rect((50, 50), (300, 20)),

start\_value=variable1, value\_range=(0, 100),

manager=panel\_manager)

slider2 = pygame\_gui.elements.UIHorizontalSlider(relative\_rect=pygame.Rect((50, 100), (300, 20)),

start\_value=variable2, value\_range=(0, 100),

manager=panel\_manager)

reset\_button = pygame\_gui.elements.UIButton(relative\_rect=pygame.Rect((200, 150), (100, 40)),

text='Reset', manager=panel\_manager)

clock = pygame.time.Clock()

is\_running = True

while is\_running:

time\_delta = clock.tick(60) / 1000.0

for event in pygame.event.get():

if event.type == pygame.QUIT:

is\_running = False

# Handle slider events

if event.type == pygame.USEREVENT:

if event.user\_type == pygame\_gui.UI\_HORIZONTAL\_SLIDER\_MOVED:

if event.ui\_element == slider1:

variable1 = int(event.value)

elif event.ui\_element == slider2:

variable2 = int(event.value)

# Handle button click

elif event.user\_type == pygame\_gui.UI\_BUTTON\_PRESSED:

if event.ui\_element == reset\_button:

variable1 = default\_variable1

variable2 = default\_variable2

slider1.set\_value(variable1)

slider2.set\_value(variable2)

panel\_manager.process\_events(event)

# Update Pygmy program with the adjusted variables

# Clear the main game window

window.fill((255, 255, 255))

# Draw or update your Pygmy program based on variable1 and variable2

# Draw the separate control panel window

panel\_window.fill((200, 200, 200))

panel\_manager.update(time\_delta)

panel\_manager.draw\_ui(panel\_window)

pygame.display.flip()

pygame.quit()