import pygame

import time

import os

# Initialize Pygame

pygame.init()

# Set up the screen

screen = pygame.display.set\_mode((0, 0), pygame.FULLSCREEN)

screen\_width, screen\_height = screen.get\_size()

# Load digit images

image\_folder = os.path.join(os.path.dirname(\_\_file\_\_), "assets")

digit\_images = {str(i): pygame.image.load(os.path.join(image\_folder, f"{i}.png")) for i in range(10)}

# Scale images to fit the screen

digit\_width = screen\_width // 10

digit\_height = screen\_height // 2

digit\_images = {k: pygame.transform.scale(img, (digit\_width, digit\_height)) for k, img in digit\_images.items()}

# Define functions

def render\_time(current\_time):

"""Displays the current time on the screen."""

x\_position = screen\_width // 2 - digit\_width \* 3 # Start drawing in the middle of the screen

for char in current\_time:

if char == ":":

x\_position += digit\_width // 2 # Adjust for the colon spacing

continue

digit = digit\_images[char]

screen.blit(digit, (x\_position, screen\_height // 2 - digit\_height // 2))

x\_position += digit\_width # Move to the next position

def flip\_animation(old\_digit, new\_digit, x, y):

"""Creates a flipping animation for digit changes."""

for i in range(1, 11):

# Clear the area

screen.fill((0, 0, 0), (x, y, digit\_width, digit\_height))

# Top half of the old digit

scale = digit\_height \* (11 - i) // 10

offset = (digit\_height - scale) // 2

old\_part = pygame.transform.scale(old\_digit, (digit\_width, scale))

screen.blit(old\_part, (x, y + offset))

pygame.display.flip()

pygame.time.delay(50)

# Bottom half of the new digit

new\_part = pygame.transform.scale(new\_digit, (digit\_width, scale))

screen.blit(new\_part, (x, y + offset))

pygame.display.flip()

pygame.time.delay(50)

# Main game loop

running = True

last\_time = "" # Store the last displayed time for comparison

while running:

# Clear the screen

screen.fill((0, 0, 0)) # Black background

# Get current time as HHMMSS

current\_time = time.strftime("%H%M%S")

# Check for time changes and animate flips

if last\_time:

for i, char in enumerate(current\_time):

if last\_time[i] != char: # If the digit has changed

x = screen\_width // 2 - digit\_width \* 3 + i \* digit\_width

y = screen\_height // 2 - digit\_height // 2

flip\_animation(digit\_images[last\_time[i]], digit\_images[char], x, y)

# Render the current time

render\_time(current\_time)

pygame.display.flip()

# Update last time

last\_time = current\_time

# Handle events

for event in pygame.event.get():

if event.type == pygame.QUIT:

running = False

elif event.type == pygame.KEYDOWN:

if event.key == pygame.K\_ESCAPE: # Exit on Escape key

running = False

pygame.time.delay(100) # Delay to control frame rate

pygame.quit()