import pygame

import random

pygame.init()

# Set the window size

screen\_width = 800

screen\_height = 600

screen = pygame.display.set\_mode((screen\_width, screen\_height))

pygame.display.set\_caption("Health Hero")

# Load the images and sounds

bg\_image = pygame.image.load("background.png").convert()

object\_image = pygame.image.load("object.png").convert\_alpha()

provider\_image = pygame.image.load("provider.png").convert\_alpha()

syrup\_image = pygame.image.load("syrup.png").convert\_alpha()

honey\_image = pygame.image.load("honey.png").convert\_alpha()

cough\_sound = pygame.mixer.Sound("cough.wav")

success\_sound = pygame.mixer.Sound("success.wav")

failure\_sound = pygame.mixer.Sound("failure.wav")

# Set the object's initial position

object\_x = random.randint(50, screen\_width - 50)

object\_y = random.randint(50, screen\_height - 50)

# Set the provider's initial position

provider\_x = screen\_width // 2

provider\_y = screen\_height - 100

# Set the treatment mode variables

treatment\_mode = False

treatment\_text = ""

treatment\_image = None

treatment\_sound = None

# Set the game loop variables

game\_running = True

game\_success = False

# The game loop

while game\_running:

# Handle events

for event in pygame.event.get():

if event.type == pygame.QUIT:

game\_running = False

elif event.type == pygame.MOUSEBUTTONDOWN:

# Check if the provider is near the object

if abs(provider\_x - object\_x) < 50 and abs(provider\_y - object\_y) < 50:

treatment\_mode = True

# Set the treatment mode variables based on the object's symptoms

if random.random() < 0.5:

treatment\_text = "Cough"

treatment\_image = syrup\_image

treatment\_sound = cough\_sound

else:

treatment\_text = "Fever"

treatment\_image = honey\_image

treatment\_sound = None

# Move the provider based on the arrow keys

keys = pygame.key.get\_pressed()

if keys[pygame.K\_LEFT] and provider\_x > 0:

provider\_x -= 5

elif keys[pygame.K\_RIGHT] and provider\_x < screen\_width - 50:

provider\_x += 5

elif keys[pygame.K\_UP] and provider\_y > 0:

provider\_y -= 5

elif keys[pygame.K\_DOWN] and provider\_y < screen\_height - 50:

provider\_y += 5

# Draw the background and objects

screen.blit(bg\_image, (0, 0))

screen.blit(object\_image, (object\_x, object\_y))

screen.blit(provider\_image, (provider\_x, provider\_y))

# Draw the treatment mode interface

if treatment\_mode:

pygame.draw.rect(screen, (255, 255, 255), (50, 50, screen\_width - 100, screen\_height - 100))

font = pygame.font.Font(None, 50)

text = font.render("Treatment mode: " + treatment\_text, True, (0, 0, 0))

screen.blit(text, (100, 100))

if treatment\_image:

screen.blit(treatment\_image, (screen\_width // 2, screen\_height // 2))

if treatment\_sound:

treatment\_sound.play()

# Check if the game is over

if game\_success or pygame.time.get\_ticks() > 10000:

game\_running