import math  
import random  
  
import pygame  
from pygame import mixer  
  
# Intialize the pygame  
pygame.init()  
  
# create the screen  
screen = pygame.display.set\_mode((800, 600))  
  
  
# Background  
background = pygame.image.load('background (1).png')  
  
  
# Sound  
mixer.music.load('background.wav')  
mixer.music.play(-1)  
  
# Caption and Icon  
pygame.display.set\_caption("Chick Invader")  
icon = pygame.image.load('chick (1).png')  
pygame.display.set\_icon(icon)  
  
# Player  
playerImg = pygame.image.load('man.png')  
playerX = 370  
playerY = 480  
playerX\_change = 0  
  
# Enemy  
enemyImg = []  
enemyX = []  
enemyY = []  
enemyX\_change = []  
enemyY\_change = []  
num\_of\_enemies = 6  
  
for i in range(num\_of\_enemies):  
 enemyImg.append(pygame.image.load('fox (1).png'))  
 enemyX.append(random.randint(0, 736))  
 enemyY.append(random.randint(50, 150))  
 enemyX\_change.append(4)  
 enemyY\_change.append(40)  
  
# Bullet  
  
# Ready - You can't see the bullet on the screen  
# Fire - The bullet is currently moving  
  
bulletImg = pygame.image.load('water-balloons.png')  
bulletX = 0  
bulletY = 480  
bulletX\_change = 0  
bulletY\_change = 10  
bullet\_state = "ready"  
  
# Score  
  
score\_value = 0  
font = pygame.font.Font('freesansbold.ttf', 32)  
  
textX = 10  
testY = 10  
  
# Game Over  
over\_font = pygame.font.Font('freesansbold.ttf', 64)  
  
  
def show\_score(x, y):  
 score = font.render("Score : " + str(score\_value), True, (255, 255, 255))  
 screen.blit(score, (x, y))  
  
  
def game\_over\_text():  
 over\_text = over\_font.render("GAME OVER", True, (255, 255, 255))  
 screen.blit(over\_text, (200, 250))  
  
  
def player(x, y):  
 screen.blit(playerImg, (x, y))  
  
  
def enemy(x, y, i):  
 screen.blit(enemyImg[i], (x, y))  
  
  
  
  
def fire\_bullet(x, y):  
 global bullet\_state  
 bullet\_state = "fire"  
 screen.blit(bulletImg, (x + 16, y + 10))  
  
  
def isCollision(enemyX, enemyY, bulletX, bulletY):  
 distance = math.sqrt(math.pow(enemyX - bulletX, 2) + (math.pow(enemyY - bulletY, 2)))  
 if distance < 27:  
 return True  
 else:  
 return False  
  
  
# Game Loop  
running = True  
while running:  
 # Background Image  
 screen.blit(background, (0, 0))  
  
 # RGB = Red, Green, Blue  
 screen.fill((0, 100, 0))  
 for event in pygame.event.get():  
 if event.type == pygame.QUIT:  
 running = False  
  
 # if keystroke is pressed check whether its right or left  
 if event.type == pygame.KEYDOWN:  
 if event.key == pygame.K\_LEFT:  
 playerX\_change = -5  
 if event.key == pygame.K\_RIGHT:  
 playerX\_change = 5  
 if event.key == pygame.K\_SPACE:  
 if bullet\_state == "ready":  
 bulletSound = mixer.Sound("laser.wav")  
 bulletSound.play()  
 # Get the current x cordinate of the spaceship  
 bulletX = playerX  
 fire\_bullet(bulletX, bulletY)  
  
 if event.type == pygame.KEYUP:  
 if event.key == pygame.K\_LEFT or event.key == pygame.K\_RIGHT:  
 playerX\_change = 0  
  
 # 5 = 5 + -0.1 -> 5 = 5 - 0.1  
 # 5 = 5 + 0.1  
  
 playerX += playerX\_change  
 if playerX <= 0:  
 playerX = 0  
 elif playerX >= 736:  
 playerX = 736  
  
 # Enemy Movement  
 for i in range(num\_of\_enemies):  
  
 # Game Over  
 if enemyY[i] > 440:  
 for j in range(num\_of\_enemies):  
 enemyY[j] = 2000  
 game\_over\_text()  
 break  
  
 enemyX[i] += enemyX\_change[i]  
 if enemyX[i] <= 0:  
 enemyX\_change[i] = 2.5  
 enemyY[i] += enemyY\_change[i]  
 elif enemyX[i] >= 736:  
 enemyX\_change[i] = -2.5  
 enemyY[i] += enemyY\_change[i]  
  
 # Collision  
 collision = isCollision(enemyX[i], enemyY[i], bulletX, bulletY)  
 if collision:  
 explosionSound = mixer.Sound("explosion.wav")  
 explosionSound.play()  
 bulletY = 480  
 bullet\_state = "ready"  
 score\_value += 1  
 enemyX[i] = random.randint(0, 736)  
 enemyY[i] = random.randint(50, 150)  
  
 enemy(enemyX[i], enemyY[i], i)  
  
 # Bullet Movement  
 if bulletY <= 0:  
 bulletY = 480  
 bullet\_state = "ready"  
  
 if bullet\_state == "fire":  
 fire\_bullet(bulletX, bulletY)  
 bulletY -= bulletY\_change  
  
 player(playerX, playerY)  
 show\_score(textX, testY)  
 pygame.display.update()