import pygame  
from pygame.locals import \*  
import time  
import random  
  
SIZE = 40  
  
  
class Apple:  
 def \_\_init\_\_(self, parent\_screen):  
 self.parent\_screen = parent\_screen  
 self.image = pygame.image.load("apple.jpg").convert()  
 self.x = 120  
 self.y = 120  
  
 def draw(self):  
 self.parent\_screen.blit(self.image, (self.x, self.y))  
 pygame.display.flip()  
  
 def move(self):  
 self.x = random.randint(1, 25) \* SIZE  
 self.y = random.randint(1, 20) \* SIZE  
  
  
class Snake:  
 def \_\_init\_\_(self, parent\_screen, length):  
 self.parent\_screen = parent\_screen  
 self.image = pygame.image.load("block.jpg").convert()  
 self.direction = 'down'  
  
 self.length = length  
 self.x = [40] \* length  
 self.y = [40] \* length  
  
 def move\_left(self):  
 self.direction = 'left'  
  
 def move\_right(self):  
 self.direction = 'right'  
  
 def move\_up(self):  
 self.direction = 'up'  
  
 def move\_down(self):  
 self.direction = 'down'  
  
 def walk(self):  
 # update body  
 for i in range(self.length - 1, 0, -1):  
 self.x[i] = self.x[i - 1]  
 self.y[i] = self.y[i - 1]  
  
 # update head  
 if self.direction == 'left':  
 self.x[0] -= SIZE  
 if self.direction == 'right':  
 self.x[0] += SIZE  
 if self.direction == 'up':  
 self.y[0] -= SIZE  
 if self.direction == 'down':  
 self.y[0] += SIZE  
  
 self.draw()  
  
 def draw(self):  
 self.parent\_screen.fill((110, 110, 5))  
  
 for i in range(self.length):  
 self.parent\_screen.blit(self.image, (self.x[i], self.y[i]))  
 pygame.display.flip()  
  
 def increase\_length(self):  
 self.length += 1  
 self.x.append(-1)  
 self.y.append(-1)  
  
  
def is\_collision(x1, y1, x2, y2):  
 if x1 >= x2 < x2 + SIZE:  
 if y2 <= y1 < y2 + SIZE:  
 return True  
 return False  
  
  
class Game:  
 def \_\_init\_\_(self):  
 pygame.init()  
 self.surface = pygame.display.set\_mode((1000, 800))  
 self.snake = Snake(self.surface, 2)  
 self.snake.draw()  
 self.apple = Apple(self.surface)  
 self.apple.draw()  
  
 def display\_score(self):  
 font = pygame.font.SysFont('arial', 30)  
 score = font.render(f"Score: {self.snake.length}", True, (200, 200, 200))  
 self.surface.blit(score, (850, 10))  
  
 def play(self):  
 self.snake.walk()  
 self.apple.draw()  
 self.display\_score()  
 pygame.display.flip()  
  
 if is\_collision(self.snake.x[0], self.snake.y[0], self.apple.x, self.apple.y):  
 self.snake.increase\_length()  
 self.apple.move()  
  
 def run(self):  
 running = True  
  
 while running:  
 for event in pygame.event.get():  
 if event.type == KEYDOWN:  
 if event.key == K\_ESCAPE:  
 running = False  
  
 if event.key == K\_LEFT:  
 self.snake.move\_left()  
  
 if event.key == K\_RIGHT:  
 self.snake.move\_right()  
  
 if event.key == K\_UP:  
 self.snake.move\_up()  
  
 if event.key == K\_DOWN:  
 self.snake.move\_down()  
  
 elif event.type == QUIT:  
 running = False  
  
 self.play()  
  
 time.sleep(.2)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 game = Game()  
 game.run()