|  |
| --- |
| **Write a Python Program to Create a Snake Game** |

**Aim:**

The aim of this Python script is to implement a simple Snake Game using the Pygame library.

**Objective:**

1.Initialize Pygame: Begin by initializing the Pygame library for game development.

2. Define Colors: Set up color codes for various elements in the game, such as the snake, background, and food.

3. Set Up Display: Create a game window with specified dimensions and a caption.

4. Initialize Clock: Set up a clock to control the game's frames per second (FPS).

5. Define Snake Properties:

- Set the initial position of the snake's head.

- Set initial movement direction (x1\_change

and y1\_change).

- Create lists to store the snake's body parts and

initialize the snake's length.

6. Generate Initial Food Position: Randomly position the first food item within the game window.

7. Main Game Loop:

- Handle events, such as user input and window

close events.

- Check for collisions with the screen boundaries.

- Update snake's position based on user input.

- Draw the game window with a filled background.

- Draw the snake and the food on the screen.

- Check for collisions with the snake's body.

- Update the display and continue the loop.

8. Game Over Handling:

- If the snake collides with the screen boundaries or

itself, initiate the game over sequence.

- Display the player's score.

9. Game Over Screen Loop:

- While in the game over state, wait for player input.

- Allow the player to either quit the game or restart.

10. Quit the Game: Properly quit Pygame and exit the script.

**Program:**

import pygame

import time

import random

pygame.init()

white = (255, 255, 255)

yellow = (255, 255, 102)

black = (0, 0, 0)

red = (213, 50, 80)

green = (0, 255, 0)

blue = (50, 153, 213)

dis\_width = 600

dis\_height = 400

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

clock = pygame.time.Clock()

snake\_block = 10

snake\_speed = 15

font\_style = pygame.font.SysFont(None, 50)

def our\_snake(snake\_block, snake\_list):

for x in snake\_list:

pygame.draw.rect(dis, white, [x[0], x[1], snake\_block, snake\_block])

def Your\_score(score):

value = font\_style.render("Your Score: " + str(score), True, white)

dis.blit(value, [0, 0])

def gameLoop():

game\_over = False

game\_close = False

x1 = dis\_width / 2

y1 = dis\_height / 2

x1\_change = 0

y1\_change = 0

snake\_List = [ ]

Length\_of\_snake = 1

foodx = round(random.randrange(0, dis\_width - snake\_block) / 10.0) \* 10.0

foody = round(random.randrange(0, dis\_height - snake\_block) / 10.0) \* 10.0

while not game\_over:

while game\_close:

dis.fill(blue)

Your\_score(Length\_of\_snake - 1)

pygame.display.update()

for event in pygame.event.get():

if event.type == pygame.KEYDOWN:

if event.key == pygame.K\_q:

game\_over = True

game\_close = False

if event.key == pygame.K\_c:

gameLoop()

for event in pygame.event.get():

if event.type == pygame.QUIT:

game\_over = True

if event.type == pygame.KEYDOWN:

if event.key == pygame.K\_LEFT:

x1\_change = -snake\_block

y1\_change = 0

elif event.key == pygame.K\_RIGHT:

x1\_change = snake\_block

y1\_change = 0

elif event.key == pygame.K\_UP:

y1\_change = -snake\_block

x1\_change = 0

elif event.key == pygame.K\_DOWN:

y1\_change = snake\_block

x1\_change = 0

if x1 >= dis\_width or x1 < 0 or y1 >= dis\_height or y1 < 0:

game\_close = True

x1 += x1\_change

y1 += y1\_change

dis.fill(blue)

pygame.draw.rect(dis, green, [foodx, foody, snake\_block, snake\_block])

snake\_Head = []

snake\_Head.append(x1)

snake\_Head.append(y1)

snake\_List.append(snake\_Head)

if len(snake\_List) > Length\_of\_snake:

del snake\_List[0]

for x in snake\_List[:-1]:

if x == snake\_Head:

game\_close = True

our\_snake(snake\_block, snake\_List)

Your\_score(Length\_of\_snake - 1)

pygame.display.update()

if x1 == foodx and y1 == foody:

foodx = round(random.randrange(0, dis\_width - snake\_block) / 10.0) \* 10.0

foody = round(random.randrange(0, dis\_height - snake\_block) / 10.0) \* 10.0

Length\_of\_snake += 1

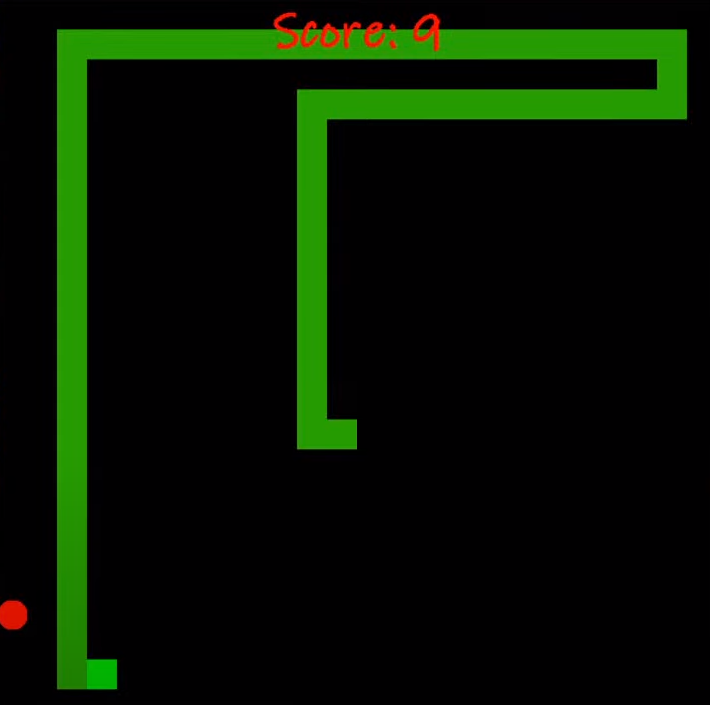
clock.tick(snake\_speed)

pygame.quit()

quit()

gameLoop()

**Output:**



**Conclusion:**

The script achieves its objectives by creating a functional Snake Game with basic features. It implements game initialization, user input handling, collision detection, scoring, and a game over sequence. The player can control the snake, eat food to grow longer, and the game provides a simple yet engaging interactive experience.

# **Github Link:**

# https://github.com/BharathRaja18/ProjectBca