```
# importing libraries
Import pygame
Import time
Import random

Snake_speed = 15

# Window size
Window_x = 720
Window_y = 480

# defining colors
Black = pygame.Color(0, 0, 0)
White = pygame.Color(255, 255, 255)
Red = pygame.Color(255, 0, 0)
Green = pygame.Color(0, 255, 0)
Blue = pygame.Color(0, 0, 255)
# Initialising pygame
Pygame.init()

# Initialise game window
Pygame.display.set_caption('GeeksforGeeks Snakes')
Game_window = pygame.display.set_mode((window_x, window_y))

# FPS (frames per second) controller
Fps = pygame.time.Clock()
# defining snake default position
Snake_position = [100, 50]

# defining first 4 blocks of snake
# body
```

```
Snake_body = [ [100, 50],
                              [90, 50],
                              [80, 50],
                              [70, 50]
                            ]
# fruit position
Fruit_position = [random.randrange(1, (window_x//10)) * 10,
                              Random.randrange(1, (window_y//10)) * 10]
Fruit_spawn = True

# setting default snake direction
# towards right
Direction = 'RIGHT'
Change_to = direction
# initial score
Score = 0

# displaying Score function
Def show_score(choice, color, font, size):

        # creating font object score_font
        Score_font = pygame.font.SysFont(font, size)

        # create the display surface object
        # score_surface
        Score_surface = score_font.render('Score : ' + str(score), True, color)

        # create a rectangular object for the
        # text surface object
        Score_rect = score_surface.get_rect()
```

```
        # displaying text
        Game_window.blit(score_surface, score_rect)
# game over function
Def game_over():

        # creating font object my_font
        My_font = pygame.font.SysFont('times new roman', 50)

        # creating a text surface on which text
        # will be drawn
        Game_over_surface = my_font.render('Your Score is : ' + str(score), True, red)

        # create a rectangular object for the text
        # surface object
        Game_over_rect = game_over_surface.get_rect()

        # setting position of the text
        Game_over_rect.midtop = (window_x/2, window_y/4)

        # blit will draw the text on screen
        Game_window.blit(game_over_surface, game_over_rect)
        Pygame.display.flip()

        # after 2 seconds we will quit the
        # program
        Time.sleep(2)

        # deactivating pygame library
        Pygame.quit()

        # quit the program
```

```
        Quit()
        # Main Function
While True:

        # handling key events
        For event in pygame.event.get():
                If event.type == pygame.KEYDOWN:
                        If event.key == pygame.K_UP:
                                Change_to = 'UP'
                        If event.key == pygame.K_DOWN:
                                Change_to = 'DOWN'
                        If event.key == pygame.K_LEFT:
                                Change_to = 'LEFT'
                        If event.key == pygame.K_RIGHT:
                                Change_to = 'RIGHT'

        # If two keys pressed simultaneously
        # we don't want snake to move into two directions
        # simultaneously
        If change_to == 'UP' and direction != 'DOWN':
                Direction = 'UP'
        If change_to == 'DOWN' and direction != 'UP':
                Direction = 'DOWN'
        If change_to == 'LEFT' and direction != 'RIGHT':
                Direction = 'LEFT'
        If change_to == 'RIGHT' and direction != 'LEFT':
                Direction = 'RIGHT'

        # Moving the snake
        If direction == 'UP':
                Snake_position[1] -= 10
```

```
If direction == ‘DOWN’:
        Snake_position[1] += 10
If direction == ‘LEFT’:
        Snake_position[0] -= 10
If direction == ‘RIGHT’:
        Snake_position[0] += 10

# Snake body growing mechanism
# if fruits and snakes collide then scores will be
# incremented by 10
Snake_body.insert(0, list(snake_position))
If snake_position[0] == fruit_position[0] and snake_position[1] == fruit_position[1]:
        Score += 10
        Fruit_spawn = False
Else:
        Snake_body.pop()

If not fruit_spawn:
        Fruit_position = [random.randrange(1, (window_x//10)) * 10,
                                        Random.randrange(1, (window_y//10)) * 10]

Fruit_spawn = True
Game_window.fill(black)

For pos in snake_body:
        Pygame.draw.rect(game_window, green, pygame.Rect(
        Pos[0], pos[1], 10, 10))

Pygame.draw.rect(game_window, white, pygame.Rect(
Fruit_position[0], fruit_position[1], 10, 10))
```

```
# Game Over conditions
If snake_position[0] < 0 or snake_position[0] > window_x-10:
        Game_over()
If snake_position[1] < 0 or snake_position[1] > window_y-10:
        Game_over()


# Touching the snake body
For block in snake_body[1:]:
        If snake_position[0] == block[0] and snake_position[1] == block[1]:
                Game_over()


# displaying score continuously
Show_score(1, white, 'times new roman', 20)


# Refresh game screen
Pygame.display.update()


# Frame Per Second /Refresh Rate
Fps.tick(snake_speed)
```