Chapter 5: Snake

Description: Snake game to grow the snake by eating food.

Why? This chapter will mix what we have learned in the previous chapters. We will use complex structures again as well as imported functions and several things tied in.

Resources:

- Python 3

Why? Use pygame more with different inputs as well as having dynamic(Growing and shrinking) structures.

To-do:

- Print board and snake
- Move snake
- Random food
- Score + grow snake
- Dead when snake touches the snake

Code:

Step 1:

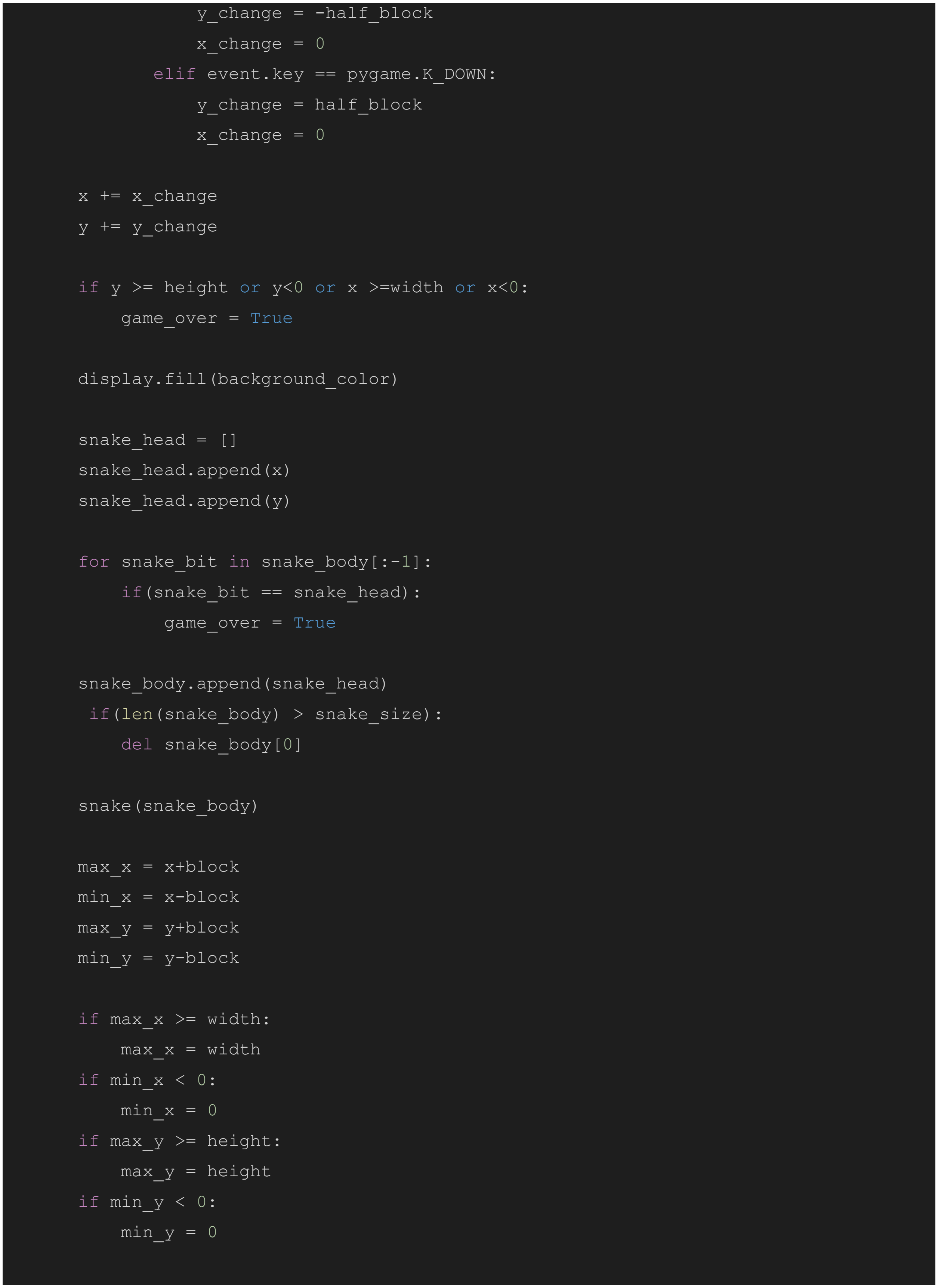
```
import pygame
import time
import random
pygame.init()
background_color = (255, 255, 255)
food_color = (213, 50, 80)
snake_color = (0, 255, 0)
width = 1000
height = 1000
display = pygame.display.set_mode((width, height))
pygame.display.set_caption('Snake!!!!!!')
clock = pygame.time.Clock()
snake_speed = 25

def gameLoop():
   game_over = False
   x = 500
   y = 500
   snake_List = []
   Length_of_snake = 1
   while not game_over:
       for event in pygame.event.get():
           if event.type == pygame.QUIT:
               game_over = True

       display.fill(background_color)
       pygame.display.update()
       clock.tick(snake_speed)
   pygame.quit()
   quit()
gameLoop()
```

```
Import pygame, time and random
Initiate pygame
Set colors of different objects in game
Set window width and height
Add title to window “Snake!!!!”
Grab clock value from pygame
Set speed snake will move

Define game loop
    Status if game is over
    X and y coordinate of snake
    Snake parts list
    Current length of snake
    While not over
        For each event
            If event is quit
                Then game is over
        Fill in background color
        Update pygame display
        Control time for next window update by snake speed
    Quit game
Start game loop
```

Step 2:

```
import pygame
import time
import random
pygame.init()
background_color = (255, 255, 255)
food_color = (213, 50, 80)
snake_color = (0, 255, 0)
width = 1000
```

```
height = 700
block = 20
half_block = block/2
snake_speed = 25

display = pygame.display.set_mode((width, height))
pygame.display.set_caption('Snake!!!!!!')
clock = pygame.time.Clock()
def snake(snake_body):
  for x in snake_body:
      pygame.draw.rect(display, snake_color, [x[0], x[1], block, block])

def gen_food():
   food = random.randint(block,width-block), random.randint(block,height-block)
   return food

def game():

   game_over = False
   x = 500
   y = 500
   snake_body = []
   snake_size = 1
   food = gen_food()

   x_change = 0
   y_change = 0

   while not game_over:

       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:
                  x_change = -half_block
                  y_change = 0
              elif event.key == pygame.K_RIGHT:
                  x_change = half_block
                  y_change = 0
              elif event.key == pygame.K_UP:
```



```
           y_change = -half_block
           x_change = 0
       elif event.key == pygame.K_DOWN:
           y_change = half_block
           x_change = 0

x += x_change
y += y_change

if y >= height or y<0 or x >=width or x<0:
    game_over = True

display.fill(background_color)

snake_head = []
snake_head.append(x)
snake_head.append(y)

for snake_bit in snake_body[:-1]:
    if(snake_bit == snake_head):
        game_over = True

snake_body.append(snake_head)
 if(len(snake_body) > snake_size):
    del snake_body[0]

snake(snake_body)

max_x = x+block
min_x = x-block
max_y = y+block
min_y = y-block

if max_x >= width:
    max_x = width
if min_x < 0:
    min_x = 0
if max_y >= height:
    max_y = height
if min_y < 0:
    min_y = 0
```

```
        if(food[0] >= min_x and food[0] <= max_x and food[1] >= min_y and food[1] <=
max_y):
            snake_size+=1
            food= gen_food()

        pygame.draw.rect(display, food_color, [food[0], food[1], block, block])

        pygame.display.update()
        clock.tick(snake_speed)

    print("Score: " +str(snake_size))
    pygame.quit()
    quit()
game()
```

* Remembering the limits of whatever object you are using is important to remember as it will be constant constraint in most things you work on

```
Import pygame, time and random
Initiate pygame
Set colors of different objects in game
Set window width and height
Set size of a block
Add title to window “Snake!!!!!”
Grab clock value from pygame
Set speed snake will move

Define snake accepting parameter snake body
        For each value in snake body
                Draw a rectangle with snake color at snake body coordinates size block by block (20 x 20)

Define gen food
        Food x and y position will randomly be generated
        Return food position

Define game
        Status if game is over
        X and y coordinate of snake
        Snake parts list
        Current length of snake
        Generate food
        X and y change variables
```

```
    While not over
        For each event
            If event is quit
                Then game is over
            If event is up, down, left or right update x and y coordinates (stop vertical if going horizontal and vice versa
        If x or y coordinates leave frame of window then game is over (crashed)
        Update x and y coordinates based on x and y changes
        Update snake position per body part piece
        Remove excess variable (add the head and remove the tail)
        If any body part touches snake head then game over
        Create a range for touching a block along x and y coords (blocks are 20 pixels big)
        If max or min values are too large or too small round to window limits
        If current position of snake matches range of food coordinates
            Add to snake size and generate food
        Draw snake
        Draw food
        Fill in background color
        Update pygame display
        Control time for next window update by snake speed
    Quit game
```

Extra:
For extra stuff look here:
https://github.com/DownRamp/Games/blob/master/snake.py
****
THIS IS THE IMPORTANT PART PLEASE DON’T SKIP
****
Next steps:

- Upgrades (faster moving, fat snake, invulnerable)
- Versus mode
- Visual
- Random moving another snake