

Snake game with Pygame

installing Pygame

Pip install Pygame

importing Pygame

import Pygame

or

import Pygame as Pg

other modul

import OS

import Random

initializing Pygame module

Pygame.init()

for all module

Pygame.mixer.init()

for ~~mod~~ music

function of Pygame :

⇒ to create game display
`Pygame.display.set mode ((width, height))`

⇒ to load image

`bgimg = Pygame.image.load ("img Path")` # it returns value so

store it in variable

`bgimg = Pygame.transform.scale (filename, (width, height))`
 • convert alpha)

⇒ # to transform image according to window size,

⇒ to set game title

`pygame.display.set_caption ("Snakegame.mj")`

⇒ to update display



`pygame.display.update()`

⇒ to set refresh rate (FPS)

`clock = Pygame.time.Clock()`

`clock.tick (60)`

⇒ to set font

font fontsize
 ↓ ↓

`font = Pygame.font.Sysfont (name, size)`

⇒ to draw rectangle

`pygame.draw.rect(gamewindow, color [x, y, size, size])`

⇒ to fill game window with color

`gamewindow.fill((233, 210, 229))`

⇒ to show text on screen

`text_screen("text", color, x, y)` coordinate
↙ ↘

⇒ to trace event

`pygame.event.get()`

⇒ to ~~show~~ check event type & action according to event

if event.type == Pygame.KEY :

⇒ to load music

`pygame.mixer.music.load('music file Path')`

⇒ to play music

`pygame.mixer.music.play()`



⇒ to create random coordinate (food)

~~random~~ ~~randint~~ randint (x, y)

⇒ to set Photo on game window

gamewindow.blit (filename , 0, 0) ↗ coordinate

⇒ to quit Pygame

Pygame.quit()

⇒ to quit Python program

quit()

Snake game . PY

PAGE NO.:

```
import Pygame  
import random  
import os
```

} importing modules

```
pygame.mixer.init() # for music  
pygame.init() # initialize Pygame module
```

defining colors

```
white = (255, 255, 255)  
red = (255, 0, 0)  
black = (0, 0, 0)
```

} colors

creating game window

```
screen_width = 900  
screen_height = 600
```

```
game_window = pygame.display.set_mode((screen_width, screen_height))
```

creating game window & assigning it to variable

Background image

```
bgimg = pygame.image.load("1.jpg")
```

```
bgimg = pygame.transform.scale(bgimg, (screen_width, screen_height)).convert_alpha()
```



Game title

```
Pygame.display.set_caption("Snake MJ")
```

```
Pygame.display.update() # to update it on window
```

to set clock(FPS) & font

```
clock = Pygame.time.clock()
```

```
font = Pygame.font.Sysfont (None, 55)
```

defining functions (user defined)

```
→ def text_screen(text, color, x, y):  
    screen_text = font.render(text, True, color)  
    gamewindow.blit(screen_text, [x, y])
```

it is used to show text on gaming window, welcome window, gameover window.

```
→ def plot_snake(gamewindow, color, snk_list, snake_size)  
    for x, y in snk_list:  
        Pygame.draw.rect(gamewindow, color, [x, y,  
            snake_size, snake_size])
```

to draw snake

⇒ def welcome():

exit_game = False

while not exit_game:

gamewindow.fill((233, 210, 229))

textscreen

f" call

{ textscreen("welcome to snakes", Black, 260, 250)

{ textscreen("Press Space Bar to Play", Black, 230, 250)

for event in pygame.event.get():

to quit

game

{

if event.type == pygame.QUIT:

exit_game = True

Keyboard input

space input

music

{

if event.type == Py. KEYDOWN:

if event.key == pygame.K_SPACE:

pygame.mixer.music.load('1.mp3')

pygame.mixer.music.play()

Start gameloop

update display

FPS (60)

pygame.display.update()

clock.tick(60)

gameloop()

⇒

def gameloop():

exit_game = False

game_over = False

Snake_x = 45

Snake_y = 55

velocity_x = 0

velocity_y = 0

snk_list = []

snk_length = 1

} game specific
variable

creating highscore file

```
if (not os.path.exists("hiscore.txt")):
    with open("hiscore.txt", "w") as f:
        f.write("0")
```

```
with open("hiscore.txt", "r") as f:
    hiscore = f.read()
```

Some more variable

```
food_x = random.randint(20, screen_width/2)
food_y = random.randint(20, screen_height/2)
score = 0
init_velocity = 5
snake_size = 30
fps = 60
```

game loop

while not exit game:

write high
score in file

if game over:

```
with open("hiscore.txt", "w") as f:
    f.write(str(hiscore))
```

game window fill white

```
text screen("Game over! Enter to continue",
            rect, 100, 250)
```

if game
over

show
msg on
screen

get
event
if quit

```
for event in pygame.event.get():
    if event.type == pygame.QUIT:
        exitgame = True
```

out from game

if enter
restart game

```
if event.type == pygame.KEYDOWN:
    if event.key == pygame.K_RETURN
```

welcome


```
# welcome()
```

fⁿ call

else: # game not over

if user hit
exit ☒

[for event in pygame.event.get():
if event.type == pygame.QUIT:
exit_game = True

Keyboard 

[if event.type == pygame.KEYDOWN:
if event.key == pygame.K_RIGHT:
velocity_x = init_velocity
velocity_y = 0



[if event.key == pygame.K_LEFT:
velocity_x = -init_velocity
velocity_y = 0



[if event.key == pygame.K_UP:
velocity_y = -init_velocity
velocity_x = 0



[if event.key == pygame.K_DOWN:
velocity_y = init_velocity
velocity_x = 0

Cheat code

[if event.key == pygame.K_g:
score += 10

dirn't
speed to
Snake

[Snake_x = Snake_x + velocity_x
Snake_y = Snake_y + velocity_y

eat food

[if abs(Snake_x - food_x) < 16 and
abs(Snake_y - food_y) < 16:
score += 10

Score 100

New

Food created [food_x = random.randint(20, screen_width)
food_y = random.randint(20, screen_height/2)



```

Snk_length += 5 # length Tes
if score > int(hiscore): ] High score
    hiscore = score      updated

```

By Photo
display
score & High Score
draw food

```

gamewindow.fill(white)
# game window .blit (bgimg, (0,0))
text_screen("Score: " + str(score) + " Hiscore " +
            str(hiscore), red, 5, 5)
pygame.draw.rect(gamewindow, red, [food_x, food_y,
Snake_size, Snake_size])

```

```

head = []
head.append(Snake_x)
head.append(Snake_y)
Snk_list.append(head)

```

} list to manage
length of snake

```

if len(Snk_list) > Snk_length: { if length Tes
    del Snk_list[0]              } then delete
                                first node

```

Snake
Bite itself
game over

```

if head in Snk_list[:-1]:
    gameover = True
    pygame.mixer.music.load('2.mp3')
    pygame.mixer.music.play()

```

Snake
hitted
Boundary
game over

```

if Snake_x < 0 or Snake_x > Screen_width or
Snake_y < 0 or Snake_y > Screen_height:
    gameover = True
    pygame.mixer.music.load('2.mp3')
    pygame.mixer.music.play()

```

Not Snake

```

plot_snake(gamewindow, black, Snk_list, Snake_size)
pygame.display.update() # update display

```


pygame.quit() # exit from Pygame
quit() # exit program

welcome() # calling welcome fⁿ