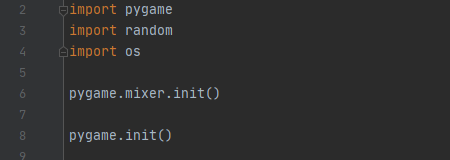
**Joc 2:Snake**

**Prima etapa:Import**

->Librariile de care avem nevoie pentru acest proiect sunt :pygame,random si os.

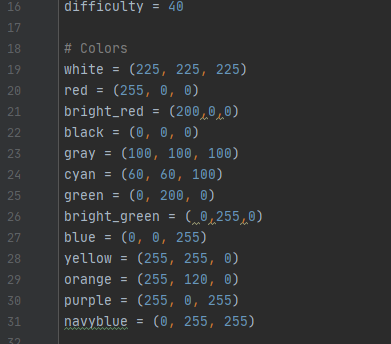


Initializam modulele pygame rexpectiv pygame.mixer pentru a putea folosi suntet de tip mp3.

**A doua etapa:Intializare**

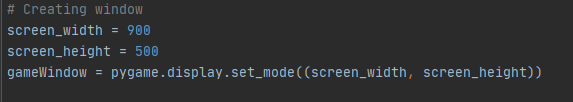
In aceasta etapa vom intializa dificultatea,culorile pe care le folosim,crearea ferestrei pe care se afla jocu,setarea background-ului,setare nume si font-ul acestuia .

Initializare Culorii:



Creare gameWindow:

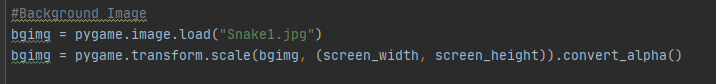
Initializarea latimi si inatlimi ferestrei pe care se va afla jocul nostru.Prin folosirea unei variabile gameWindow se va seta display-ul ca fiind de dimensiunile setate anterior(screen\_width,screen\_hight).



Setare Background:

Utilizam o variabila bgimg prin care se incarca o imagine (pygame.image.load).

Dupa setarea background-ului acesta nu se afla la o dimensiune pe care o dorim(width ,height) pentru a seta aceste dimensiuni background-ului nostru utilizam un modul al pygame-ului numit pygame.transform.scale prin care scalam imaginea la dimensiunie definite de noi anterior, in urma modificari inaltimi si latimi imagini se va face o copie a imagini care va avea formatul droit de pixeli (convert\_alpha).



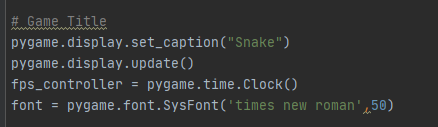
Titlu si fontul acestuia:

In continuare utilizam module pygame prin care setam nume joc(pygame.display.set\_caption(“Snake”)) .

Pygame.display.update->actualizam portiuni ale ecranului pentru afisarea titlului

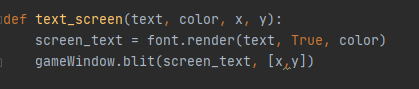
fps\_controller=pygame.time.Clock()->setare unui obiect prin care urmarim trecerea timpului

font=pygame.font.SysFont->Setare unui “Font object from the system fonts”



**A treia etapa:Definire Functii**

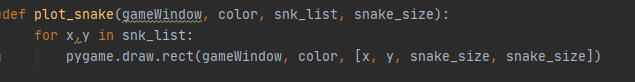
In aceasta etapa se vor define functile prin care se va pune text pe ecran,plotare sarpe,si crearea unui mesaj de intampinare.



Def text\_screen(text,color,x,y)->definire functia text in care se folosesc variabilele text,color,x,y.

Screen\_text=font.render(text,True,color)->randarea font pe text cu o anumita culoare pe care o selectezi in game loop.

gameWindow.blit(screen\_text,[x,y])->afisarea pe ecran a text-ului generat(in ecranu de intampinare) ,[x,y] reprezinta cordonatele unde se afla text-ul pe ecran



Def plot\_snake(gameWindow,color,snk\_list,snake\_size)-definire functie de plotare a sarpelui in care se folsesc variabilele gameWindow,color,snk\_list,snake\_size.

For x,y in snk\_list:

Pygame.draw.rect(gameWindow,color,[x,y,snake\_size,snake\_size])

->pentru cordonatele x si y din snk\_list se va desena un dreptunghi in interiorul gameWindow cu o anumita culoare la anumite cordonate cu marimea de snake\_size

Def welcome():

Exit\_game=False

While not exit\_game-> while not loop executa corpul loopului pana cand conditia este adevarata exit\_game=true

Corp while not exit\_game:

gameWindow.fill((255,255,255)-> umplere ecran cu culoarea alb

gameWindow.blit(bgimg,[0,0])->umplere ecran cu background-ul incarcat anterior la cordonate [0,0] central

text\_screen->adaugare text de culoare alb la cordonatele metionate

for event in pygame.event.get()

if event.type ==pygame.QUIT

exit\_game=true

daca evenimentul este egal cu modulul pygame.QUIT(sarpele a atins unu din pereti sau sa atins pe el insusi) exit\_game devine adevarat

if event.type ==pygame.KEYDOWN: ->daca evenimentul este egal cu modulul de apasare a tastei in pygame

if event.key ==pygame.K\_SPACE:-> iar tasta apasata este space(se selecteaza sa fie space)

pygame.mixer.music.load(‘music.wav’)

pygame.mixer.music.play(-1,0.0)-se porneste musica cu un play de -1 adica aceasta va fi in loop pana cand se indeplinesc conditile ca jocu sa se fi terminat

gameloop()->se apeleaza functia gameloop daca conditile din while not sunt indeplinite se porneste jocu

pygame.display.update()-se actualizeaza fereastra

fps\_controller.tick(60)-frame rate-ul pentru fereastra de welcome

**A patra etapa:GameLoop**

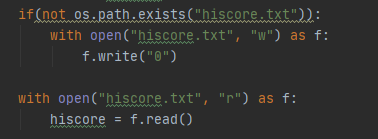
In aceasta etapa au fost initializate niste variabile cum ar fi exit\_game ,game\_over cu conditia initiala de a fi false

Se intilizeaza pozitia din care incepe sarpele(snake\_x,snake\_y)

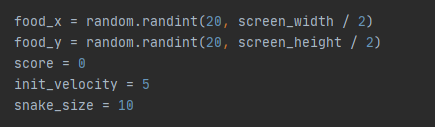
Volacity\_x,volacity\_y initializate cu 0 prin care vom determina directia in care se va deplasa sarpele

Snk\_list=[]-> lista in care sunt posibile poziti in care poate aparea food-ul

Snk\_length=1->lungimea initiala



Verificare daca exista hiscore.text,se va folosi pentru prima data bibloteca os,se verifica daca fisieru .txt exista

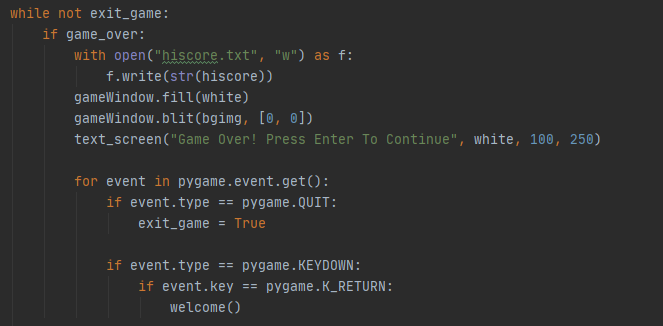


Food\_x=random,randint(20,screen\_width/2)->20 reprezinta pozitia de start,screen\_width/2 pozitia unde aceasta nu poate aparea.

Intializare scor=0

Init\_volacity=5 ->initializate viteza

Snake\_size=10->marime cap sarpe(dreptunghiului)



Loopul while care se executa pana cand conditia lui exit\_game nu devine true

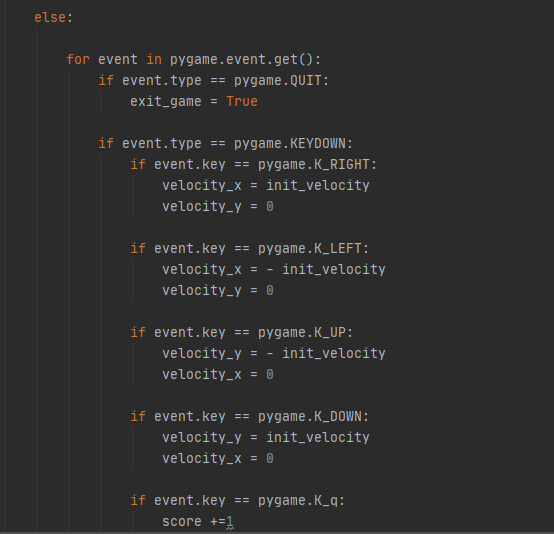
Daca game\_over=true se deschide hiscore.txt si se citeste valoarea acestuia

gameWindow.fill(white)->umplere ecran cu alb

gamewindow.bilt(bgimg,[0,0])->afisearea background incarcat

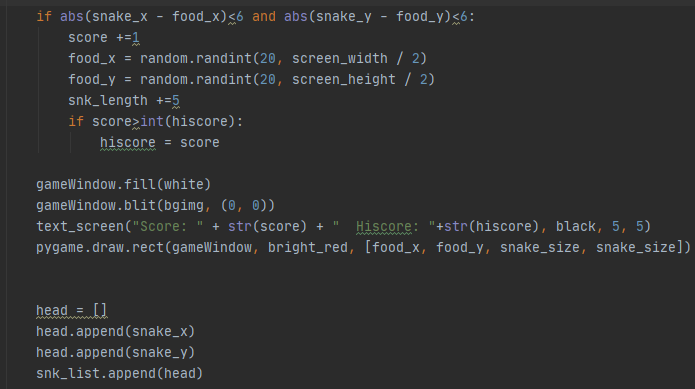
text\_screen->afisare text cu culoarea alb si cu pozitia centrala

for event in pygame.event.get(0….se verifica daca jocu sa sfarsit ,daca da ne intoarcem la ecranu de intampinare(welcome() este apelat)



Daca jocu nu sa sfarsti se defines miscarile noastre stanga ,dreapta sus jos si o tasta q prin care poti adauga la scoru tau 1 (fara ca marimea sarpelui sa creasca)

snake\_x = snake\_x + velocity\_x  
snake\_y = snake\_y + velocity\_y



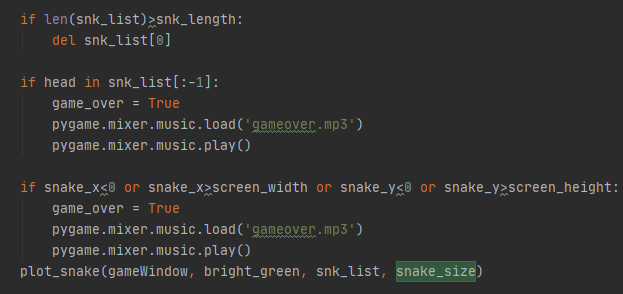
Daca sarpele(capul sarpelui)mananca mancarea se executa urmatorul if prin care se verifica daca sarpele a mancat mancarea ,se creste scoru cu 1 ,se genereaza o noua pozitie a mancari pe ecran si se creste marimea sarpelui cu inca un cub

If score>int(hiscore):

Hiscore=score

Se verifica daca scorul curent este egal cu cel mai mare scor.

Se umple din nou ecranul cu alb si se adauga backgroundul ,se adauga text-ul pentru score si hiscore pe ecran si se deseneaza mancarea cu culoarea rosu deschis cu cordonate food\_x,food\_y(care sunt random pe suprafata create) si se verifica daca capul sarpelui a atins mancarea si se adauga din snk\_list la corpul sarpelui



If len(snk\_list)>snk\_length

Del snk\_list[0] ->se sterge obiectul

If head in snk\_list[:-1]:->daca capu atinge corpu se indepineste conditia pentru game\_over si se aunde muzica incarcata pentru acest lucru

Game\_over=true

If snake\_x<0 snake\_x>screen\_width or snake\_y<0 or snake\_y>screeb\_heighht ->se verifica daca sarpele a atins marginile ecranului ,daca da game \_over=true

Plot\_snake(gameWindow,bright\_green,snk\_list,snake\_size)->plotare sarpe

pygame.display.update()->actualizam protiuni ale ecranului pentru afisarea sarpelui

fps\_controller.tick(difficulty)->actualizare clock(timer) framurile in care se misca sarpele cu valoarea care se afla in difficulty.

Pygame.quit()

Quit()

Welcome()