Snake

1. Primer disenyarem el marc de joc i la serp
2. Posarem les normes per el fi de joc (quan toqui els marges s'acaba)
3. Afagirem el menjar
4. Posarem un marcador amb els resultats /Fins aquí és el que m'ha donat temps/
5. Augmentarà de tamany cada vegada que passi per el menjar
6. (Si dona temps) ho farem que sigui una app

In [1]:

```
#----- libraries -----#
import pygame
import time
import random

#----- Global definitions -----#
#Color definition
BLACK = (0,0,0)
WHITE = (255, 255, 255)
RED = (255, 0, 0)
MAGENTA = (255,0,230)

#----- Class definition -----#
class Snake(object):
    def __init__(self, game):
        self.x1 = game.width/2
        self.y1 = game.hight/2
        self.x1_change = 0
        self.y1_change = 0
    def draw(self, game): #Drawing of the snake
        pygame.draw.rect(game.window,WHITE,[self.x1,self.y1, game.s_block, game.s_blo
    def move(self, game): #Movement of the snake
        self.x1 += self.x1_change
        self.y1 += self.y1_change
    def position(self, valuex, valuey): #position of the snake
        self.x1_change = valuex
        self.y1_change = valuey
    def boundaries(self, game): #boundaries of the game, where the snake cannot go
        if self.x1 >= game.width or self.x1 < 0 or self.y1 >= game.hight or self.y1 <
            return True
        return False
    def eat(self, game):
        if self.x1 == game.apples.fx and self.y1== game.apples.fy:
            game.message("Nyam!!")
            game.apples.fx = round(random.randrange(0, game.width - game.s_block) / 1
            game.apples.fy = round(random.randrange(0, game.width - game.s_block) / 1
            game.score += 1

class Apples(object):
    def __init__(self, game):
        self.fx = round(random.randrange(0, game.width - game.s_block) / 10.0) * 10.0
        self.fy = round(random.randrange(0, game.width - game.s_block) / 10.0) * 10.0
    def draw(self, game):
        pygame.draw.rect(game.window,RED,[self.fx,self.fy, game.s_block, game.s_block


class Game(object):
    width = 600
    hight = 600
    s_block = 10
    s_speed = 15
    def __init__(self):
        self.score = 0
        pygame.init()
        self.window = pygame.display.set_mode((self.width,self.hight))
        pygame.display.set_caption('Snake - Clara Alonso') #Set the current window ca
    def play(self):
        clock = pygame.time.Clock()
```

```
        game_over=False
        game_close = False
        snake = Snake(self)
        self.apples = Apples(self)
        #--------------------- Game Bucle Start --------------#
        while not game_over:
            #--------------------- Game Event Start --------------#
            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:
                        snake.position(-self.s_block,0)
                    elif event.key == pygame.K_RIGHT:
                        snake.position(self.s_block,0)
                    elif event.key == pygame.K_UP:
                        snake.position(0,-self.s_block)
                    elif event.key == pygame.K_DOWN:
                        snake.position(0,self.s_block)

            #--------------------- Game Event End --------------#
            #Moves the snake and draws it
            snake.move(self)
            self.window.fill(BLACK)
            self.apples.draw(self)
            snake.draw(self)
            snake.eat(self)
            self.your_score(self.score)
            #Has the snake collided with the boundaries?
            if snake.boundaries(self) == True:
                game_over = True
            #Has the game ended?
            if game_over == True:
                self.message("You've lost")

            pygame.display.update() #Update portions of the screen for software displ
            clock.tick(self.s_speed)
        #--------------------- Game Bucle End --------------#
        time.sleep(2)
        pygame.quit()
        quit()

    def message(self, text):
        tex = pygame.font.SysFont(None, 50).render(text, True, MAGENTA)
        self.window.blit(tex, [self.width/2,self.hight/2])
        pygame.display.update()
    def your_score(self, score):
        value = pygame.font.SysFont(None, 35).render(f"Your Score: {self.score}", Tru
        self.window.blit(value, [0, 0])

#--------------- Start Game --------#
if __name__ == "__main__":
    Game().play()
```

```
pygame 2.1.0 (SDL 2.0.16, Python 3.8.8)
Hello from the pygame community. https://www.pygame.org/contribute.html
```

In [ ]:

In [ ]: