

```

from pygame import *
'''Required classes'''
#parent class for sprites
class GameSprite(sprite.Sprite):
    def __init__(self, player_image, player_x, player_y, player_speed, wight, height):
        super().__init__()
        self.image = transform.scale(image.load(player_image), (wight, height)) #e.g.
55,55 - parameters
        self.speed = player_speed
        self.rect = self.image.get_rect()
        self.rect.x = player_x
        self.rect.y = player_y

    def reset(self):
        window.blit(self.image, (self.rect.x, self.rect.y))

class Player(GameSprite):
    def update_r(self):
        keys = key.get_pressed()
        if keys[K_UP] and self.rect.y > 5:
            self.rect.y -= self.speed
        if keys[K_DOWN] and self.rect.y < win_height - 80:
            self.rect.y += self.speed
    def update_l(self):
        keys = key.get_pressed()
        if keys[K_w] and self.rect.y > 5:
            self.rect.y -= self.speed
        if keys[K_s] and self.rect.y < win_height - 80:
            self.rect.y += self.speed

#game scene:
back = (200, 255, 255) #background color (background)
win_width = 600
win_height = 500
window = display.set_mode((win_width, win_height))
window.fill(back)

#flags responsible for game state
game = True
finish = False
clock = time.Clock()
FPS = 60

#creating ball and paddles
racket1 = Player('racket.png', 30, 200, 4, 50, 150)
racket2 = Player('racket.png', 520, 200, 4, 50, 150)
ball = GameSprite('tenis_ball.png', 200, 200, 4, 50, 50)

font.init()
font = font.Font(None, 35)
lose1 = font.render('PLAYER 1 LOSE!', True, (180, 0, 0))
lose2 = font.render('PLAYER 2 LOSE!', True, (180, 0, 0))

speed_x = 3

```

```
speed_y = 3

while game:
    for e in event.get():
        if e.type == QUIT:
            game = False

    if finish != True:
        window.fill(back)
        racket1.update_l()
        racket2.update_r()
        ball.rect.x += speed_x
        ball.rect.y += speed_y
        if sprite.collide_rect(racket1, ball) or sprite.collide_rect(racket2, ball):
            speed_x *= -1
            speed_y *= 1
        #if the ball reaches screen edges, change its movement direction
        if ball.rect.y > win_height-50 or ball.rect.y < 0:
            speed_y *= -1
        #if ball flies behind this paddle, display loss condition for player 1
        if ball.rect.x < 0:
            finish = True
            window.blit(lose1, (200, 200))
            game_over = True
        #if ball flies behind this paddle, display loss condition for player 2
        if ball.rect.x > win_width:
            finish = True
            window.blit(lose2, (200, 200))
            game_over = True
        racket1.reset()
        racket2.reset()
        ball.reset()
    display.update()
    clock.tick(FPS)
```