

<!--Universidad Internacional del Ecuador-->

Pong v1.0 -

AA3 {

<Por="Juan Cabrera"/>

}



# Contenido del videojuego

- Interfaz Gráfica
- Controles con teclado
- Colisiones
- Sistema de vidas

# Interfaz Gráfica

## Librerias {

```
import pygame #libreria util para la interfaz grafica y algunas funciones logicas del juego
import sys #esta libreria sirve para salir del juego
import random #libreria para tener valores random al momento de que la bola sale del centro
import os #util para poder dar direcciones a archivos mas facil
```

## Ventana {

```
#Creamos la interfaz grafica para poder ver
ancho, alto = 800, 600
ventana = pygame.display.set_mode((ancho,alto)) #Le dice que haga la pantalla de las medidas del alto y ancho
pygame.display.set_caption("Pong v.0 Juan Cabrera Aprendizaje Autonomo 2") #Que escriba el titulo en la barra de la ventana
```

## Definición de variables {

```
#Colores, barras, atributos
negro= (0,0,0)
blanco= (255, 255, 255)
ancho_barra = 10
alto_barra = 100
velocidad_barra = 7
bola = 10
velocidad_bola = 5
corazon_altura = 16
corazon_esp = 10
boton_reiniciar = pygame.Rect(250, 350, 250, 60) #x,y,largo,alto
boton_salir = pygame.Rect(275, 420, 200, 60)#x,y,largo,alto
fuente = pygame.font.SysFont("consolas", 40)
```

}

# Objetos claves {

01

```
#Variable de Barra izquierda, su posicion y medidas  
barra_izquierda = pygame.Rect(  
    50, #posicion de x  
    alto // 2 - alto_barra // 2, #posicion de y  
    ancho_barra, #ancho de la barra  
    alto_barra #alto de la barra  
)
```

```
#Variable de Barra derecha, posicion y medidas  
barra_derecha = pygame.Rect(  
    ancho - 50, #posicion de x  
    alto // 2 - alto_barra // 2, #posicion de y  
    ancho_barra, #ancho de la barra  
    alto_barra #alto de la barra  
)
```

02

```
#Variable de Bola, posicion y medidas  
bola_centro = pygame.Rect(  
    ancho - 400, #posicion de x  
    alto - 300, #posicion de y  
    bola, #ancho de la bola  
    bola #alto de la bola  
)
```

03

```
#Creamos variables de las vidas  
vidas_derecha = 3  
vidas_izquierda = 3
```

```
#El os nos sirve para poder poner el directorio de un archivo mas facil  
ruta_base = os.path.dirname(__file__)  
ruta_corazon = os.path.join(ruta_base, "corazon.png")
```

}

# Control de teclado{

```
#Leer el teclado
teclado = pygame.key.get_pressed()

#Aqui el teclado lee que teclas se aplastan y que pasa cuando se aplastan en una direccion x o y (en este caso y)
if teclado [pygame.K_w]:
| barra_izquierda.y -= velocidad_barra
if teclado[pygame.K_s]:
| barra_izquierda.y += velocidad_barra

if teclado[pygame.K_UP]:
| barra_derecha.y -= velocidad_barra
if teclado[pygame.K_DOWN]:
| barra_derecha.y += velocidad_barra
```

# Movimiento de bola, colisión y resta de vidas {

```
#Movimiento de la bola y se movera con movimiento randomizado
bola_centro.x += velocidad_bola_x
bola_centro.y += velocidad_bola_y
```

```
#Rebote con bordes,si la bola toca del borde de arriba o abajo
if bola_centro.top <= 0 or bola_centro.bottom >= alto:
| velocidad_bola_y *=-1 #cambia de direccion
```

```
#Si la bola pasa perdemos vidas
if bola_centro.left <=0:
| vidas_izquierda -=1
| reseteo_bola()

elif bola_centro.right >= ancho:
| vidas_derecha -=1
| reseteo_bola()
```

```
if bola_centro.colliderect(barra_izquierda) and velocidad_bola_x < 0:
| velocidad_bola_x *= -1
| velocidad_bola_x += 1
if bola_centro.colliderect(barra_derecha) and velocidad_bola_x > 0:
| velocidad_bola_x += 1
| velocidad_bola_x *= -1
```

}

# Definimos funciones {

reseteo\_bola():

```
#definimos una funcion que se llama reseteo bola y hace esto
def reseteo_bola():
    global velocidad_bola_x, velocidad_bola_y #cambia estas variables globalmente

    #pone la bola en el centro siempre que se resetee
    bola_centro.center = (ancho - 400, alto - 300)
    #elige al azar si salir izq o der
    dir_x = random.choice([-1,1])
    #elige al azar si salir arriba o abajo
    dir_y = random.choice([-1,1])
    #asignamos velocidades
    velocidad_bola_x = velocidad_bola * dir_x
    velocidad_bola_y = velocidad_bola * dir_y
```

dibujar\_cora():

```
#Definimos funcion para dibujar corazones
def dibujar_cora():
    for i in range(vidas_izquierda):
        x=150+i * (corazon_altura + corazon_esp)
        y=20
        ventana.blit(corazon, (x,y))

    for i in range(vidas_derecha):
        x = ancho - 150 - corazon_altura - i * (corazon_altura + corazon_esp)
        y=20
        ventana.blit(corazon, (x,y))
```

}

# Empezamos la lógica del juego {

```
#Bucle del inicio y juego
correr = True
juego_empezado = False

while correr:

    reloj.tick(FPS)

    for eventos in pygame.event.get():
        if eventos.type == pygame.QUIT:
            correr = False
        elif not juego_empezado:
            if eventos.type == pygame.MOUSEBUTTONDOWN and eventos.button == 1:
                juego_empezado = True
                reseteo_bola()
```

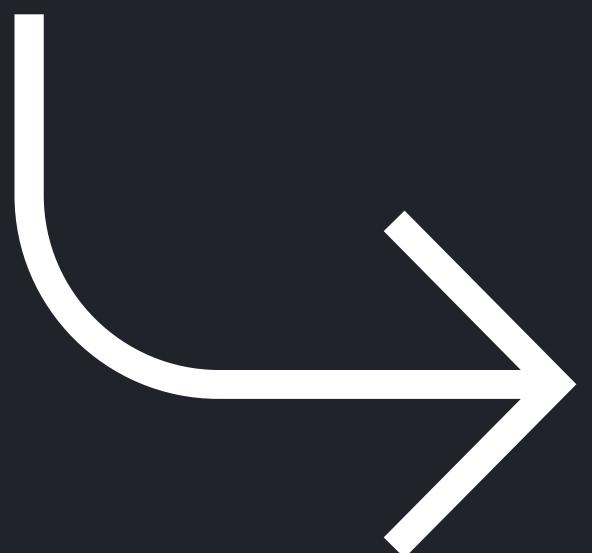
if not juego\_empezado:

```
if not juego_empezado:
    ventana.fill(negro)
    texto_inicio = fuente.render("Clic para empezar", True, blanco)
    ventana.blit(
        texto_inicio,
        (
            ancho // 2 - texto_inicio.get_width() // 2,
            alto // 2 - texto_inicio.get_height() // 2
        )
    )
    pygame.display.flip()
    continue
```

}

## Pantalla final {

```
#Que pasa cuando llegas a 0 vidas?  
if vidas_izquierda <=0 or vidas_derecha <=0:  
    if vidas_izquierda <=0:  
        mensaje = "Jugador de la Derecha ganó!"  
    elif vidas_derecha <=0:  
        mensaje = "Jugador de la Izquierda ganó!"  
    else:  
        mensaje = "Ups.... esto no debería pasar.."
```



```
#Bucle: Entrar a pantalla Final  
esperando = True  
while esperando:  
    for eventos in pygame.event.get():  
        if eventos.type == pygame.QUIT:  
            correr = False  
            esperando = False  
  
        elif eventos.type == pygame.MOUSEBUTTONDOWN and eventos.button == 1:  
            if boton_reiniciar.collidepoint(eventos.pos):  
                # Reiniciar juego  
                vidas_izquierda = 3  
                vidas_derecha = 3  
                barra_izquierda.y = alto // 2 - alto_barra // 2  
                barra_derecha.y = alto // 2 - alto_barra // 2  
                reseteo_bola()  
                mensaje = ""  
                esperando = False # salir de la pantalla de menu  
  
            elif boton_salir.collidepoint(eventos.pos):  
                correr = False # cerrar juego  
                esperando = False
```

}

<!--Universidad Internacional del Ecuador-->

Gracias {

<Por="Juan Cabrera"/>

}