



SECRETARÍA DE
INNOVACIÓN

Agenda

Sesión 18/18

1. Punto 1: PYGAME
2. Punto 2: Controlador de eventos
3. Punto 3: Personajes
4. Punto 4: Movimientos
5. Punto 5: Posiciones
6. Punto 6: Colisiones
7. Punto 7: Marcador

PROGRAMACIÓN CON PYTHON

PYGAME

PYGAME

¿Qué es?

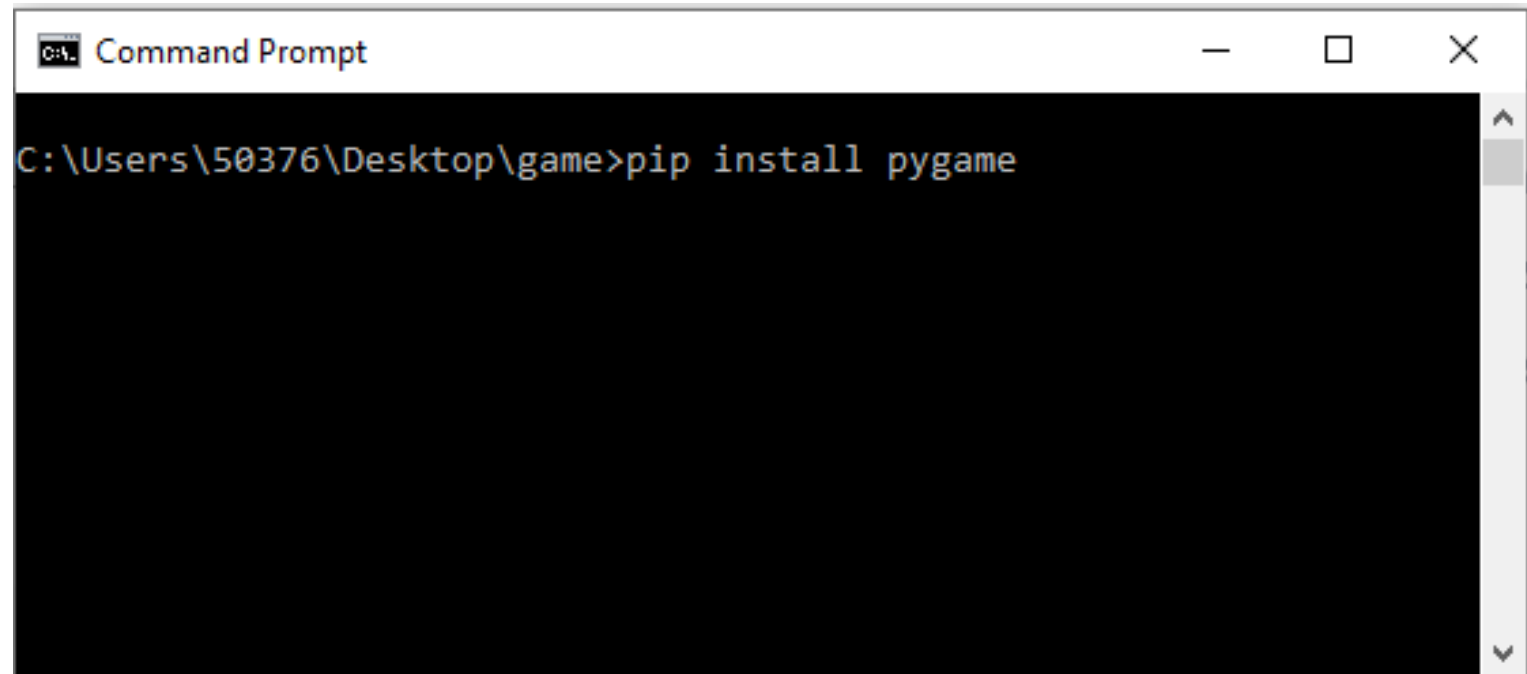
PyGame es un módulo del lenguaje de programación Python que permiten la creación de videojuegos en dos dimensiones de una manera sencilla. Mediante PyGame podemos utilizar sprites (objetos), cargar y mostrar imágenes en diferentes formatos, sonidos, etc.

PYGAME

<https://www.pygame.org/docs/>

PYGAME

Instalación



```
Command Prompt

C:\Users\50376\Desktop\game>pip install pygame
```

PYGAME

Verificar instalación

```
juego1.py x
1 import pygame|
2
```

PYGAME

Controlador de eventos

```
juego1.py x
1  import pygame
2  import sys
3
4  #construyendo una ventana
5  #800,600 es una tupla (ancho y alto)
6  ventana = pygame.display.set_mode((800,600))
7
8  #bucle principal del juego
9  #aquí se registran todos los eventos
10 fin_del_juego = False
11
12 while not fin_del_juego:
13     for el_evento in pygame.event.get():
14         #si presionamos el botón cerrar
15         if el_evento.type == pygame.QUIT:
16             sys.exit()
17
```


PYGAME

Verificando eventos

```
juego1.py
1  import pygame
2  import sys
3
4  #construyendo una ventana
5  #800,600 es una tupla (ancho y alto)
6  ventana = pygame.display.set_mode((800,600))
7
8  #bucle principal del juego
9  #aquí se registran todos los eventos
10 fin_del_juego = False
11
12 while not fin_del_juego:
13     for el_evento in pygame.event.get():
14         print(el_evento) #imprime los eventos
15         #si presionamos el botón cerrar
16         if el_evento.type == pygame.QUIT:
17             sys.exit()
18
```

PYGAME

Creando al personaje principal

```
juego1.py x
1  import pygame
2  import sys
3
4  #construyendo una ventana
5  #800,600 es una tupla (ancho y alto)
6  ventana = pygame.display.set_mode((800,600))
7
8  fin_del_juego = False
9
10 while not fin_del_juego:
11     for el_evento in pygame.event.get():
12         if el_evento.type == pygame.QUIT:
13             sys.exit()
14     #personaje principal
15     pygame.draw.rect(ventana, (255,0,0), (400,400,50,50))
16     #ventana / color RGB / posición inicial / tamaño
17     pygame.display.update()
18
```

PYGAME

Variables y constantes

```
juego1.py x
1 import pygame
2 import sys
3
4 #definiendo constantes
5 ALTO = 800
6 ANCHO = 600
7 rojo = (255,0,0)
8
9 #personaje principal
10 posicion_personaje = [400,400]
11 tamaño_personaje = [50,50]
12
13 ventana = pygame.display.set_mode((ALTO,ANCHO))
14
15 fin_del_juego = False
16
17 while not fin_del_juego:
18     for el_evento in pygame.event.get():
19         if el_evento.type == pygame.QUIT:
20             sys.exit()
21     #personaje principal
22     pygame.draw.rect(ventana, rojo,(posicion_personaje,tamaño_personaje))
23     #ventana / color RGB / posición inicial / tamaño
24     pygame.display.update()
```

PYGAME

Movimiento

```
juego1.py x game.py x
1 import pygame
2 import sys
3
4 #definiendo constantes
5 ALTO = 800
6 ANCHO = 600
7 rojo = (255,0,0)
8
9 #personaje principal
10 posicion_personaje = [400,400]
11 tamano_personaje = 50
12
13 ventana = pygame.display.set_mode((ALTO,ANCHO))
14
15 fin_del_juego = False
16
17 while not fin_del_juego:
18     #for que identifica evento
19     for el_evento in pygame.event.get():
20         if el_evento.type == pygame.QUIT:
21             sys.exit()
22
23     #movimiento del personaje
24     if el_evento.type == pygame.KEYDOWN:
25         ejex = posicion_personaje[0] #indice 0 = 50
26         if el_evento.key == pygame.K_LEFT:
27             ejex -= tamano_personaje #disminuye
28         if el_evento.key == pygame.K_RIGHT:
29             ejex += tamano_personaje #aumenta
30
31         posicion_personaje[0] = ejex #actualiza la coordenada
32
33     pygame.draw.rect(ventana, rojo, (posicion_personaje[0], posicion_personaje[1], tamano_personaje, tamano_personaje))
34     pygame.display.update()
35
```

PYGAME

Movimiento

```
1  import pygame
2  import sys
3
4  #definiendo constantes
5  ALTO = 800
6  ANCHO = 600
7  rojo = (255,0,0)
8  negro = (0,0,0)
9
```

```
ventana.fill(negro) #actualiza el color de la ventana
pygame.draw.rect(ventana, rojo, (posicion_personaje[0], posicion_personaje[1], tamano_personaje, tamano_personaje))
pygame.display.update()
```

PYGAME

Posición

```
#personaje principal  
tamano_personaje = 50  
posicion_personaje = [ANCHO / 2, ALTO - tamano_personaje]
```

PYGAME

```
10 #personaje principal
11 tamaño_personaje = 50
12 posicion_personaje = [ANCHO / 2, ALTO - tamaño_personaje]
13
14 #enemigo
15 tamaño_enemigo = 50
16 posicion_enemigo = [400,0]
17 verde = [30,166,48]
18
```

Enemigo

```
40 ventana.fill(negro) #actualiza el color de la ventana
41 pygame.draw.rect(ventana, verde, (posicion_enemigo[0], posicion_enemigo[1], tamaño_enemigo, tamaño_enemigo))
42 pygame.draw.rect(ventana, rojo, (posicion_personaje[0], posicion_personaje[1], tamaño_personaje, tamaño_personaje))
43 pygame.display.update()
44
```

PYGAME

Enemigo aleatorio

```
1  import pygame
2  import sys
3  import random
4
5
6  #definiendo constantes
7  ALTO = 600
8  ANCHO = 800
9  rojo = (255,0,0)
10 negro = (0,0,0)
11
12 #personaje principal
13 tamaño_personaje = 50
14 posicion_personaje = [ANCHO / 2, ALTO - tamaño_personaje]
15
16 #enemigo
17 tamaño_enemigo = 50
18 posicion_enemigo = [random.randint(0, ANCHO - tamaño_enemigo),0]
19 verde = [30,166,48]
20
```

Ejecutamos varias veces para verificar

PYGAME

Movimiento enemigo

```
41
42     ventana.fill(negro) #actualiza el color de la ventana
43
44     if posicion_enemigo[1] >=0 and posicion_enemigo[1] < ALTO:
45         posicion_enemigo[1] += 30
46     else:
47         posicion_enemigo[1] = 0
48
```

PYGAME

Movimiento enemigo

```
41
42     ventana.fill(negro) #actualiza el color de la ventana
43
44     if posicion_enemigo[1] >=0 and posicion_enemigo[1] < ALTO:
45         posicion_enemigo[1] += 30
46     else:
47         posicion_enemigo[0] = random.randint(0,ANCHO-tamano_enemigo)
48         posicion_enemigo[1] = 0
49
```

PYGAME

Movimiento enemigo

```
23
24 fin_del_juego = False
25 reloj = pygame.time.Clock() ←
26
27 ▼ while not fin_del_juego:
28     #for que identifica evento
29 ▼     for el_evento in pygame.event.get():
```

```
pygame.draw.rect(ventana, verde, (posicion_enemigo[0], posicion_enemigo[1], ancho_enemigo, alto_enemigo))
pygame.draw.rect(ventana, rojo, (posicion_personaje[0], posicion_personaje[1], ancho_personaje, alto_personaje))
reloj.tick(20) ←
pygame.display.update()
```

PYGAME

Colisiones

```
23
24 fin_del_juego = False
25 reloj = pygame.time.Clock()
26
27 def colision(posicion_personaje, posicion_enemigo):
28     px = posicion_personaje[0]
29     py = posicion_personaje[1]
30     ex = posicion_enemigo[0]
31     ey = posicion_enemigo[1]
32
33     if (ex >= px and ex < (px + tamano_personaje)) or (px >= ex and px < (ex + tamano_enemigo)):
34         if (ey >= py and ey < (py + tamano_personaje)) or (py >= ey and py < (ey + tamano_enemigo)):
35             return True
36     return False
37
38
```

```
posicion_enemigo[1]
0

if colision(posicion_personaje, posicion_enemigo):
    fin_del_juego = True

pygame.draw.rect(ventana, verde, (posicion_enemigo[0], posicion_enemigo[1], tam
pygame.draw.rect(ventana, rojo, (posicion_personaje[0], posicion_personaje[1],
reloj.tick(20)
pygame.display.update()
```

PYGAME

Marcador

```
21
22 ventana = pygame.display.set_mode((ANCHO, ALTO))
23
24 fin_del_juego = False
25 reloj = pygame.time.Clock()
26 score = 0 ←
27
```

```
57
58 if posicion_enemigo[1] >=0 and posicion_enemigo[1] < ALTO:
59     posicion_enemigo[1] += 30
60 else:
61     posicion_enemigo[0] = random.randint(0,ANCHO-tamano_enemigo)
62     posicion_enemigo[1] = 0
63     score += 10 ←
64     print(score)
65
66 if colision(posicion_personaje,posicion_enemigo):
67     fin_del_juego = True
68
```

PYGAME

Marcador

```
26 reloj = pygame.time.Clock()
27 score = 0
28
29 def marcador(como, texto, tamano, x, y): ←
30     font = pygame.font.SysFont(None, tamano)
31     text_como = font.render(texto, True, (30,166,48))
32     text_forma = text_como.get_rect()
33     text_forma.midtop = (x,y) #midtop posicion
34     como.blit(text_como, text_forma)#lo muestra
35
36 def colision(posicion_personaje, posicion_enemigo):
37     px = posicion_personaje[0]
38     py = posicion_personaje[1]
```

```
77
78     #el marcador
79     marcador(ventana, str(score), 25, ANCHO // 2, 10) ←
80
81     pygame.draw.rect(ventana, verde, (posicion_enemigo[0], posicion_enemigo[1], 20, 20))
82     pygame.draw.rect(ventana, rojo, (posicion_personaje[0], posicion_personaje[1], 20, 20))
83     reloj.tick(20)
84     pygame.display.update()
85
86
```

RESUMEN DE SESIÓN



SECRETARÍA DE
INNOVACIÓN