

GUÍA RÁPIDA DE PYGAME

Inicialización básica

```
import pygame, sys
pygame.init()

# Pantalla
screen = pygame.display.set_mode((800, 600))
pygame.display.set_caption("Guía Rápida Pygame")

# Reloj para FPS
clock = pygame.time.Clock()
```

1. Crear y mostrar una imagen (con blit)

```
imagen = pygame.image.load("imagen.png")    # Carga una imagen
imagen_rect = imagen.get_rect(center=(400, 300))

screen.blit(imagen, imagen_rect)  # Muestra la imagen
```

2. Crear y mostrar un rectángulo

```
rect = pygame.Rect(100, 100, 120, 80)  # x, y, ancho, alto
pygame.draw.rect(screen, (255, 0, 0), rect)
```

3. Crear y mostrar un círculo

```
pygame.draw.circle(screen, (0, 255, 0), (400, 300), 50)
```

4. Crear y mostrar texto

```
fuente = pygame.font.Font(None, 48)
texto = fuente.render("¡Hola, Pygame!", True, (255, 255, 255))
screen.blit(texto, (250, 250))
```

5. Crear un menú con selección por cursores + ENTER

```
opciones = ["Jugar", "Opciones", "Salir"]
seleccion = 0
fuente = pygame.font.Font(None, 60)
```

```

def dibujar_menu():
    screen.fill((0, 0, 0))
    for i, texto in enumerate(opciones):
        color = (255, 255, 0) if i == seleccion else (255, 255, 255)
        render = fuente.render(texto, True, color)
        screen.blit(render, (350, 200 + i * 70))

running = True
while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT: sys.exit()
        elif event.type == pygame.KEYDOWN:
            if event.key == pygame.K_UP: seleccion = (seleccion - 1) % len(opciones)
            if event.key == pygame.K_DOWN: seleccion = (seleccion + 1) % len(opciones)
            if event.key == pygame.K_RETURN:
                print("Has seleccionado:", opciones[seleccion])

    dibujar_menu()
    pygame.display.flip()
    clock.tick(30)

```

6. Menú con selección también por ratón

```

def dibujar_menu_mouse():
    screen.fill((0, 0, 0))
    mouse = pygame.mouse.get_pos()
    for i, texto in enumerate(opciones):
        rect_texto = pygame.Rect(350, 200 + i * 70, 200, 60)
        color = (255, 255, 0) if rect_texto.collidepoint(mouse) else (255, 255, 255)
        render = fuente.render(texto, True, color)
        screen.blit(render, rect_texto.topleft)

    pygame.display.flip()

while True:
    for event in pygame.event.get():
        if event.type == pygame.QUIT: sys.exit()
        if event.type == pygame.MOUSEBUTTONDOWN:
            for i, texto in enumerate(opciones):
                rect_texto = pygame.Rect(350, 200 + i * 70, 200, 60)
                if rect_texto.collidepoint(event.pos):
                    print("Click en:", opciones[i])

    dibujar_menu_mouse()
    clock.tick(30)

```

7. Control de colisiones (rect vs rect)

```

jugador = pygame.Rect(100, 100, 50, 50)
enemigo = pygame.Rect(200, 150, 50, 50)

if jugador.colliderect(enemigo):
    print("Colisión detectada!")

```



8. Crear disparos

```
disparos = []
vel_disparo = 10

# Crear disparo al pulsar ESPACIO
for event in pygame.event.get():
    if event.type == pygame.KEYDOWN and event.key == pygame.K_SPACE:
        disparos.append(pygame.Rect(jugador.centerx, jugador.top, 5,
10))

# Mover y dibujar disparos
for d in disparos[:]:
    d.y -= vel_disparo
    if d.bottom < 0:
        disparos.remove(d)
    pygame.draw.rect(screen, (255, 255, 0), d)
```



9. Comprobar si un disparo toca un elemento

```
for d in disparos[:]:
    if d.colliderect(enemigo):
        print("Enemigo alcanzado!")
        disparos.remove(d)
```



Estructura básica del loop

```
running = True
while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT: running = False

    screen.fill((0, 0, 0))
    # Aquí dibujas todo lo anterior

    pygame.display.flip()
    clock.tick(60)
pygame.quit()
```