

# Curso de Introducción a Python

*Creando videojuegos con Python*



Álvaro Menéndez  
miniprogramadores.es

MiniProgramadores



- 2025

# Índice

---

<b>Introducción</b>	<b>3</b>
<b>Organización del curso</b>	<b>3</b>
<b>Objetivos</b>	<b>3</b>
<b>Contenidos</b>	<b>3</b>
1. Configuración del Entorno . . . . .	3
2. Nociones Básicas de programación . . . . .	4
3. Game Loop . . . . .	4
4. Sprites y Animaciones . . . . .	4
<b>Proyecto Final</b>	<b>5</b>



## Introducción

---

PyGame es un **conjunto de módulos Python** diseñado específicamente para desarrollar videojuegos. A diferencia de Scratch, permite crear juegos más complejos y profesionales usando un lenguaje de programación real. Su dificultad es notoriamente más alta que la de Scratch, así que se recomienda que los alumnos tengan **más de 12 años**.

Su importancia radica en cómo combina la potencia de Python con una API sencilla para manejar gráficos, sonido y controles. PyGame sirve de puente perfecto entre la programación visual y el código textual, permitiendo a los estudiantes aplicar conceptos fundamentales de programación en proyectos tangibles y motivadores.

## Organización del curso

---

El curso tiene una duración de **32 horas**, divididas en **sesiones de 1 hora**. La estructura se adapta progresivamente:

- Primeras 15 sesiones: Fundamentos de PyGame y Python
- Sigüientes 15 sesiones: Desarrollo de proyecto individual
- Últimas 2 sesiones: Optimización y preparación de presentación

## Objetivos

---

Al finalizar el curso, los estudiantes habrán aprendido lo siguiente:

- Como utilizar un IDE (VScode)
- Nociones básicas de Python
- Crear videojuegos 2D con PyGame

## Contenidos

---

### 1 Configuración del Entorno

Instalación de Python y PyGame. Configuración de IDE (VSCode/Cursor). Estructura básica de un proyecto PyGame.

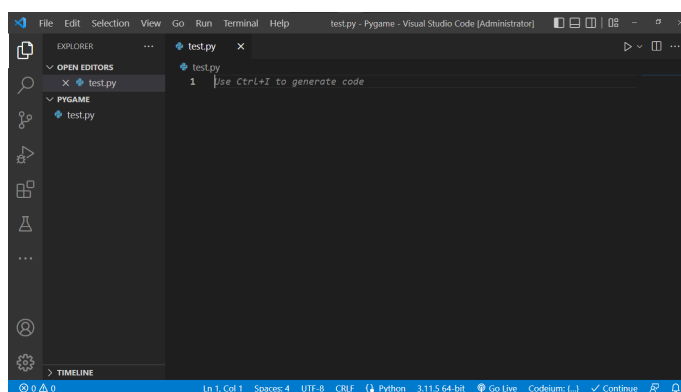


Figura 1: Proyecto vacío en VScode



## 2 Nociones Básicas de programación

**NOTA:** Esta sección es muy parecida a los contenidos del curso anterior de Scratch

Fundamentos esenciales para programar en PyGame: **Variables:** Almacenan datos. En PyGame se usan para:

- Posiciones (x, y)
- Puntuación (score = 0)
- Estados del juego (game\_active = True)

**Condicionales (if):** Toman decisiones:

```
if player_rect.colliderect(enemy_rect):
    vidas -= 1
elif powerup_active:
    score += 50
else:
    player_speed = 5
```

**Bucles (while/for):** Repiten acciones:

```
# Bucle principal del juego
running = True
while running:
    for event in pygame.event.get():
        if event.type == QUIT:
            running = False
```

Estos elementos se combinan con la lógica de PyGame para crear mecánicas interactivas.

## 3 Game Loop

El corazón de cualquier juego en pygame. Componentes principales:

- Inicialización
- Bucle principal
- Manejo de eventos
- Actualización de estado
- Renderizado

## 4 Sprites y Animaciones

Un sprite es una imagen que da vida a objetos en un videojuego. Para crear animaciones, de movimiento por ejemplo, se necesitan un conjunto de sprites que vienen en una Spritesheet:





Figura 2: Ejemplo de sprite sheet

## Proyecto Final

---

Los estudiantes desarrollarán un juego completo que incluya los contenidos explicados

