



# Pygame Zero

Un motore per fare giochi (tutto incluso, no sbatti)

 III Liceo Scientifico Biella - Scienze Applicate

 Python Biella Group



## Perché Pygame Zero?

Pygame classico: più di 10 righe per far apparire un quadrato

Pygame Zero: 5 righe per far apparire un alieno che si muove

*"La vita è troppo breve per scrivere `pygame.init()` ogni volta"*

*— Qualcuno, probabilmente*

Perfetto per chi vuole **creare**, non passare ore a configurare.



# Installazione

Con Thonny (raccomandato per chi inizia):

1. Apri Thonny
2. Vai su **Tools** → **Manage packages...**
3. Cerca `pgzero` nella barra di ricerca
4. Clicca su **Install**

È tutto qui. Thonny si occupa di tutto il resto.

Se funziona: sei fortunato (e probabilmente succederà).

Se non funziona: chiedi aiuto, non sei solo in questa battaglia.



## La struttura base

Ogni gioco Pygame Zero ha **tre funzioni magiche**:

- `draw()` → disegna tutto (chiamata 60 volte al secondo!)
- `update()` → aggiorna la logica di gioco
- Eventi → `on_mouse_down()`, `on_key_press()`, ecc.

**Fun fact:** Non devi chiamarle tu. Pygame Zero è come quel compagno di scuola che aiuta senza che te ne accorgi.



## Sfida


*Cosa fa `draw()` 60 volte al secondo?*

- A) Ti disegna un caffè ☕
- B) Ridisegna tutto lo schermo
- C) Manda messaggi alla tua ex



## Sfida

*Cosa fa `draw()` 60 volte al secondo?*

- A) Ti disegna un caffè 
- B) Ridisegna tutto lo schermo
- C) Manda messaggi alla tua ex

**Risposta:** B (ma dopo questo corso, forse anche A)



# Il nostro primo codice

```
from pgzero.actor import Actor
import pgzrun

TITLE = "Il mio primo gioco"
WIDTH = 800
HEIGHT = 600

def draw():
    screen.clear()
    screen.fill(color=(128, 0, 0)) # Rosso scuro

pgzrun.go()
```

**Risultato:** Una finestra rossa. Minimalista. Artistica. Inutile.



## Aggiungiamo un protagonista: l'Actor

```
alieno = Actor("alieno") # Carica l'immagine alieno.png

def draw():
    screen.clear()
    screen.fill(color=(128, 0, 0))
    alieno.draw() # Disegna l'alieno
```

**Actor** = sprite con superpoteri (posizione, collisioni, tutto incluso)

**Nota bene:** L'immagine deve essere nella cartella `images/` o Pygame Zero ti ignora come fai tu con lo stato WA di uno che non sopporti.



## Posizionamento: coordinate style

In Pygame Zero, il sistema di coordinate è come quello matematico... **MA INVERTITO** sull'asse Y!

- $(0, 0)$  = angolo in alto a sinistra
- X aumenta verso destra ✓
- Y aumenta verso il **BASSO** ↓ (tradimento!)

```
alieno.x = 400 # Centro orizzontale  
alieno.y = 300 # Centro verticale
```



## Sfida

*Se WIDTH=800 e HEIGHT=600, dove si trova il punto (800, 600)?*

- A) Al centro dello schermo
- B) Nell'angolo in basso a destra
- C) In una dimensione parallela, fuori schermo



## Sfida

*Se WIDTH=800 e HEIGHT=600, dove si trova il punto (800, 600)?*

- A) Al centro dello schermo
- B) Nell'angolo in basso a destra
- C) In una dimensione parallela, fuori schermo

**Risposta:** C! L'ultimo pixel valido è (799, 599). Solo fuori di uno...



# Randomizziamo! 🎲

```
from random import randint

def piazza_alieno():
    alieno.x = randint(50, WIDTH-50)
    alieno.y = randint(50, HEIGHT-50)
```

**Perché 50?** Perché l'alieno è 64x64 pixel e non vogliamo che finisca mezzo fuori schermo come i vostri screenshot mal tagliati.

**Matematica applicata:** Evitare che le cose vadano fuori dai bordi.



## Eventi: ascoltare i click

```
def on_mouse_down(pos):  
    if alieno.collidepoint(pos):  
        print("COLPITO!")  
    else:  
        print("HAI MANCATO, NOOB")
```

`collidepoint(pos)` → verifica se il punto cade dentro l'Actor.

**Magia della fisica computazionale (o quasi).**



## Variabili globali: il male necessario

```
messaggio = ""

def on_mouse_down(pos):
    global messaggio # Senza global, la variabile è solo locale
    if alieno.collidepoint(pos):
        messaggio = "Bel colpo!"
    else:
        messaggio = "Mancato..."
```

**Global = peccato veniale** in un gioco piccolo.

In progetti grandi = ricetta per il disastro. Usare con moderazione.



## Mostrare testo sullo schermo

```
def draw():  
    screen.clear()  
    screen.fill(color=(128, 0, 0))  
    alieno.draw()  
    screen.draw.text(messaggio,  
                     center=(400, 40),  
                     fontsize=60)
```

**Pro tip:** center= centra il testo. Niente calcoli manuali. Niente pianti.



## Sfida

*Cosa succede se dimentichi global messaggio ?*

- A) Python crea una variabile locale che muore subito
- B) Il messaggio rimane vuoto
- C) Il prof ti guarda male
- D) Tutte le precedenti



## Sfida

*Cosa succede se dimentichi global messaggio ?*

- A) Python crea una variabile locale che muore subito
- B) Il messaggio rimane vuoto
- C) Il prof ti guarda male
- D) Tutte le precedenti

**Risposta:** D! Classico errore Python.



## Il Clock: programmare eventi

```
from pgzero.clock import clock  
  
clock.schedule_interval(piazza_alieno, 1.0)
```

Ogni 1 secondo → chiama `piazza_alieno()`

L'alieno si sposterà automaticamente!



## Cambiare l'immagine dell'Actor

```
if alieno.collidepoint(pos):  
    alieno.image = "esplosione" # BOOM! 💣  
else:  
    alieno.image = "alieno" # Torna normale
```

Pygame Zero ricarica l'immagine al volo. Niente complicazioni.

**Requisito:** Devi avere `esplosione.png` nella cartella `images/`.



## Struttura file del progetto

```
il_mio_gioco/  
├── gioco.py           # Il tuo codice  
├── images/            # Le tue immagini  
│   ├── alieno.png  
│   └── esplosione.png  
└── sounds/           # I tuoi suoni (opzionale)  
    └── colpo.wav
```

**Regola d'oro:** Pygame Zero cerca automaticamente in `images/` e `sounds/`. Non cambiare i nomi delle cartelle o il gioco implode.



## Il codice completo

```
from pgzero.actor import Actor
from pgzero.clock import clock
from random import randint
import pgzrun

TITLE = "Colpisci l'alieno"
WIDTH = 800
HEIGHT = 600
messaggio = ""
alieno = Actor("alieno")

def draw():
    screen.clear()
    screen.fill(color=(128, 0, 0))
    alieno.draw()
    screen.draw.text(messaggio, center=(400, 40), fontsize=60)
```



## Il codice completo 🎯 (parte 2)

```
def piazza_alieno():  
    '''  
    Il limite di 50 pixel è definito per evitare che l'immagine  
    sia parzialmente fuori schermo  
    Alieno ha size 64x64  
    '''  
    alieno.x = randint(50, WIDTH-50)  
    alieno.y = randint(50, HEIGHT-50)  
    alieno.image = "alieno"  
  
def on_mouse_down(pos):  
    global messaggio  
    if alieno.collidepoint(pos):  
        messaggio = "Bel colpo!"  
        alieno.image = "esplosione"  
    else:  
        messaggio = "Mancato..."  
  
piazza_alieno()  
clock.schedule_interval(piazza_alieno, 1.0)  
pgzrun.go()
```



## Sfida finale

*Cosa si impara davvero da questo gioco?*

- A) Gestione eventi
- B) Coordinate 2D
- C) Funzioni e variabili globali
- D) Che colpire alieni è terapeutico



## Sfida finale

*Cosa si impara davvero da questo gioco?*

- A) Gestione eventi
- B) Coordinate 2D
- C) Funzioni e variabili globali
- D) Che colpire alieni è terapeutico

**Risposta:** Tutte! (Soprattutto D)



## Possibili miglioramenti 🚀

Ora che avete la base, potete aggiungere:

- **Punteggio:** Conta i colpi riusciti
- **Timer:** 30 secondi per fare più punti possibile
- **Velocità crescente:** Alieno si muove più veloce ogni 10 colpi
- **Suoni:** `sounds.colpo.play()` quando colpisci
- **Vite:** 3 errori e game over

Il limite è la vostra creatività.



## Debugging tips 🐛

**Problema:** L'alieno non appare

→ Controlla che l'immagine sia in `images/` e si chiami esattamente come scritto nel codice

**Problema:** Il messaggio non cambia

→ Hai dimenticato `global` messaggio

**Problema:** Il gioco va lento

→ Stai facendo troppi calcoli in `draw()`. Sposta la logica in `update()`

**Problema:** Crash inspiegabili

→ Benvenuto nella programmazione. Leggi l'errore. Usa `print()`. Respira.



## Risorse utili

- Documentazione ufficiale: [pygame-zero.readthedocs.io](https://pygame-zero.readthedocs.io)
- Esempi: Nella cartella di installazione di Pygame Zero
- Asset gratuiti: [opengameart.org](https://opengameart.org), [itch.io](https://itch.io), [kenney.nl](https://kenney.nl)
- Quando nulla funziona: AI o Stack Overflow (ovviamente)



# Conclusioni

## Avete imparato:

- Struttura base di Pygame Zero
- Actor, coordinate e collisioni
- Eventi mouse e timer
- Gestione stato del gioco

**Next step:** Modificate il codice, rompetelo, aggiustatelo. È così che si impara.

*"L'unico modo per imparare a programmare è programmare"*  
— Qualche guru della Silicon Valley

Ora andate e create qualcosa di epico! 🎮✨



## Domande?

Se non ci sono domande, iniziate a codare.

Se ci sono domande, probabilmente le risposte sono su Stack Overflow o ChatGPT.

Buon coding! 🐙

*"C'è sempre qualcosa da imparare per migliorarci e crescere...**insieme!**" 🌟*