

[codingcreativo.it](https://www.codingcreativo.it)

## Pygame per disegnare sullo schermo - Coding Creativo

Autore dell'articolo: Cristina

5-6 minutes

In questa lezione vedremo come utilizzare pygame per disegnare sullo schermo.

Nella scorsa lezione abbiamo visto come importare la libreria pygame e abbiamo creato la nostra prima finestra, adesso vedremo come personalizzarla.

Proviamo quindi a colorare lo sfondo. Ricordiamo alcuni colori in RGB.

Bianco – RGB(255,255,255)

Nero – RGB(0,0,0)

Rosso – RGB(255,0,0)

Verde – RGB(0,255,0)

Blu – RGB(0,0,255)

Giallo – RGB(255,255,0)

Magenta – RGB(255,0,255)

Ciano – RGB(0,255,255)

Quindi supponiamo di voler creare una finestra con lo sfondo giallo.

```
import pygame

pygame.init()

screen=pygame.display.set_mode((640,480))

pygame.display.set_caption('Coding Creativo')

GIALLO=(255, 255, 0)

screen.fill(GIALLO)

pygame.display.flip()
```

In Python il colore si rappresenta con un tupla e quindi per il giallo impostiamo la tupla (255,255,0).

Applichiamo poi il metodo **fill** all'istanza dell'oggetto screen, ovvero la variabile che abbiamo creato in precedenza. In questo modo coloreremo la nostra finestra di giallo.

Utilizziamo poi la funzione **flip** del sotto-modulo *display* affinché le modifiche diventino effettive.

Quando dobbiamo disegnare qualcosa occorre sempre chiamare la funzione **flip** per rendere effettive le modifiche.

### Utilizzare Pygame per disegnare sullo schermo.

Per disegnare sullo schermo possiamo utilizzare il **sotto – modulo draw**.

Il sotto modulo **draw** di pygame contiene alcune funzioni:

**pygame.draw.circle** – disegna un cerchio

pygame.draw.circle (superficie, colore, pos, raggio, larghezza)

**pygame.draw.polygon** – disegna un poligono

pygame.draw.polygon (superficie, colore, lista puntiforme, larghezza)

**pygame.draw.rect** – disegna un rettangolo

pygame.draw.rect (superficie, colore, Rettangolo, larghezza)

**pygame.draw.ellipse** – disegna un'ellisse

pygame.draw.ellipse (superficie, colore, Rettangolo, larghezza)

**pygame.draw.arc** – disegna un arco

pygame.draw.arc (superficie, colore, Rect, start\_angle, stop\_angle, width = 1)

**pygame.draw.line** – disegna un segmento di linea retta

pygame.draw.line (superficie, colore, start\_pos, end\_pos, larghezza = 1)

**pygame.draw.lines** – disegna più segmenti di linea contigui

pygame.draw.lines (superficie, colore, chiuso, lista puntiforme, larghezza = 1)

**pygame.draw.aaline** – disegna una linea sottile antialias

pygame.draw.aaline (superficie, colore, startpos, endpos, blend = 1)

**pygame.draw.aalines** – disegna una sequenza di linee antialias

pygame.draw.aalines (superficie, colore, chiuso, elenco punti, blend = 1)

Dove:

**superficie** – rappresenta la superficie sulla quale disegnare la forma.

**colore** – Rappresenta il colore che si vuole utilizzare. Es. (255,0,0)

**pos** – Rappresenta la posizione del centro del cerchio.

**raggio** – Rappresenta il raggio del cerchio in pixel.

**larghezza** – Rappresenta la posizione della linea o del contorno della forma. Se larghezza è uguale a 0 la forma sarà riempita.

**rettangolo** – area rettangolare dove verrà disegnata la forma.

**lista puntiforme** – Rappresenta un elenco di una quantità arbitraria di punti, in pixel (x, y).

**blend** – verrà disegnata una linea tra il primo e l'ultimo punto se vero.

**star\_angle** – Angolo iniziale in radianti

**stop\_angle** – Angolo finale in radianti

**star\_pos** – Posizione iniziale della linea in pixel

**stop\_angle** – Posizione finale della linea in pixel

Facciamo dunque degli esempi, disegnando alcune forme.

```
import pygame
```

```
pygame.init()
screen=pygame.display.set_mode((640,480),0,32)
pygame.display.set_caption('Coding Creativo')
```

```
RED=(255, 0, 0)
WHITE=(255,255,255)
BLACK=(0,0,0)
GREEN=(0,255,0)
BLUE=(0,0,255)
YELLOW=(255,255,0)
CYAN=(0,255,255)
```

```
screen.fill(YELLOW)
```

```
pygame.draw.circle(screen, GREEN, (30, 30), 10, 1)
pygame.draw.circle(screen, RED, (90, 30), 20, 0)
pygame.draw.line(screen, BLUE, (150, 30), (200, 30))
pygame.draw.rect(screen, BLACK, (230, 20, 30, 30),1)
pygame.draw.rect(screen, BLUE, (300, 10, 70, 60))
pygame.draw.ellipse(screen, CYAN, (390, 10, 80, 40), 0)
```

```
pygame.display.flip()
```

#### Alcuni link utili

[Indice tutorial sul linguaggio python](#)

1 – [Introduzione al linguaggio python](#)

2 – [Le variabili in python](#)

3 – [Operatori aritmetici e di assegnazione in python](#)

4 – [Stringhe in python](#)

5 – [Casting in python](#)

6 – [Input e print in python](#)

7 – [Primi esercizi in python](#)

8 – [Errori in python](#)

9 – [Script python](#)

10 – [Scambio di variabili in python](#)