



g2d

animazioni

```

import g2d                                     # modulo per la grafica 2d

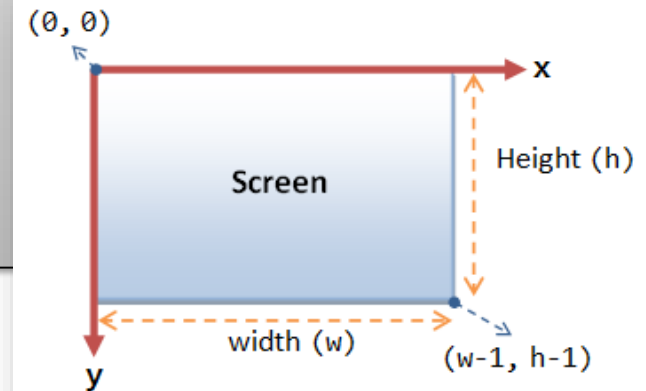
# finestra grafica (canvas), larghezza=600, altezza=400
g2d.init_canvas((600, 400))

# rettangolo((r,g,b), (left,top,w,h))
# giallo (255,255,0)
# spigolo x=150, y=100, larghezza=250, altezza=200
g2d.draw_rect((255, 255, 0), (150, 100, 250, 200))

# cerchio((r,g,b), (x_centro,y_centro), raggio)
g2d.draw_circle((0, 0, 255), (400, 300), 20)

# Ciclo gestione eventi finestra
g2d.main_loop()

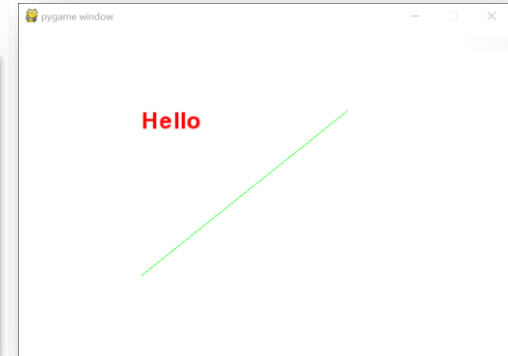
```



The 2D Screen Coordinates: The origin is located at the top-left corner, with x-axis pointing left and y-axis pointing down.

```
# linea (r,g,b) , (x0,y0) , (x1,y1)
g2d.draw_line((0, 255, 0), (150, 300), (400, 100))

# testo(stringa, (r,g,b) , (x_left,y_top) ,dimensione)
g2d.draw_text("Hello", (255, 0, 0), (150, 100), 40
)
```



```
i = 0
while i < 10:
    x = i * 25
    y = i * 25
    red = i * 25
    g2d.draw_rect((red, 0, 0), (x, y, 100, 100))
    i += 1
g2d.main_loop()
```



```

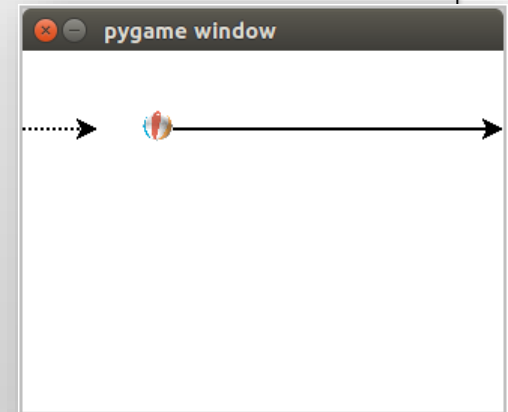
import g2d                                     # modulo per la grafica 2d

def disegna():
    '''
    imbianca il canvas
    disegna una immagine alle coordinate x,50
    sposta x di 5 pixel
    '''
    global x
    g2d.fill_canvas((255, 255, 255))          # sfondo bianco
    g2d.draw_image(imm, (x, 50))              # disegna pallina
    x = (x + 5) % 320                          # modifica x pallina

# finestra grafica (canvas), larghezza=320, altezza=200
g2d.init_canvas((320, 200))

imm = g2d.load_image("ball.png")             # carica immagine in variabile imm
x = 50                                         # posizione iniziale

# chiama la funzione disegna 30 volte al secondo
g2d.main_loop(disegna, 1000 // 30)
    
```



```
def tasto_giu(tasto: str):  
    print("premuto: ", tasto)  
  
def tasto_su(tasto: str):  
    print("rilasciato: ", tasto)  
  
# gestione pressione tasto  
g2d.handle_keyboard(tasto_giu, tasto_su)
```



python
introduzione e primi programmi
esercizi



- ***d.1 quadrati casuali***

- chiedere all'utente un numero n
- disegnare n quadrati
 - tutti con lato di 100 pixel
 - ciascuno in posizione casuale
 - ciascuno con un colore casuale

cominciare a disegnare un solo quadrato grigio, in posizione casuale



- ***d.2 orologio classico***

- disegnare 12 tacche a raggiera, come in un orologio classico
- miglioramento: disegnare anche le tacche dei minuti, più piccole

usare `math.sin` e `math.cos` per determinare le posizioni in cui disegnare

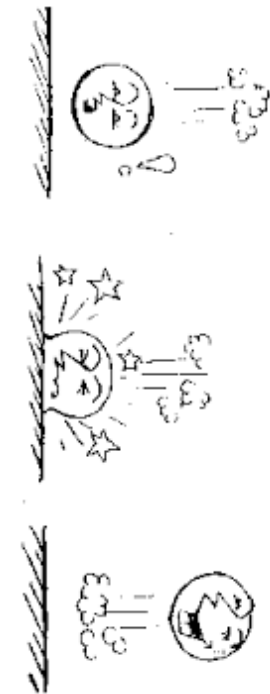


d.3 movimento orizzontale

- mostrare una *pallina* che si muove in *orizzontale*
- la pallina *rimbalza* sui bordi

memorizzare in una variabile dx lo spostamento orizzontale da effettuare ad ogni ciclo

cambiare segno a dx quando $x < 0$ oppure $x + w > screen_width$



d.4 movimento a serpentina

- mostrare una *pallina* che si muove a *serpentina*
- partire dall'esercizio precedente
- al momento del **rimbalzo**, imporre un spostamento **verticale**
- fare in modo che, in ogni frame, lo spostamento sia *solo orizzontale*, o *solo verticale*, ma *non diagonale*

