

ANIMATION D'UN MOBILE

L'application suivante met un œuvre un mobile (un carré rouge) qui part du sommet haut gauche de la surface de l'écran , descend en diagonale puis longera successivement les 4 bords de l'écran.

Le mouvement durera tant qu'on ne cliquera pas sur le bouton central, auquel cas le mouvement cessera.

```
import pygame
import sys
import time
pygame.init()                                # initialise

res = (1080,600)                             # screen resolution
screen = pygame.display.set_mode(res)
pygame.display.set_caption("Ma belle fenêtre")

color = (255,255,255)                        #blanc
vert=(0,255,0)                               #vert
color_light = (170,170,170)                  # pâle
rouge = (255,0,0)                            # rouge
color_dark = (100,100,100)                  # foncé

width = screen.get_width()                   # largeur de l'écran
height = screen.get_height()                 # hauteur de l'écran

smallfont = pygame.font.SysFont('Corbel',35) # fonte
text = smallfont.render('Fin ', True , color)

# carré rouge mobile
rect =pygame.draw.rect(screen,rouge, [10,10,10,10])

hh=1                                          # coordonnées initiales
vv=1

while True:

    for ev in pygame.event.get():
        if ev.type == pygame.QUIT:
            pygame.quit()
        if ev.type == pygame.MOUSEBUTTONDOWN: # click souris
            if width/2 <= mouse[0] <= width/2+160 and height/2 <= mouse[1] <=
                                                height/2+40:
                pygame.quit()                # sur le bouton
```

```
time.sleep(0.010)
screen.fill((60,25,60))
```

```
if vv >= 590 :
```

```
    vv=590
```

```
    hh+=1
```

```
if hh >= 1060 :
```

```
    hh=1060
```

```
    vv-=1
```

```
if vv <= 0 :
```

```
    vv=0
```

```
    hh-=1
```

```
if hh <= 0 :
```

```
    hh=0
```

```
    vv+=1
```

```
if hh != 1060 and vv != 590 and vv != 0 and hh != 0:
```

```
    hh+=1
```

```
    vv+=1
```

```
rect=pygame.draw.rect(screen,rouge, [hh,vv,10,10])
```

```
pygame.draw.rect(screen,vert, [1070,10,10,580])
```

```
mouse = pygame.mouse.get_pos()
```

```
# coord souris
```

```
# quand souris passe sur le bouton
```

```
if width/2 <= mouse[0] <= width/2+140 and height/2 <= mouse[1] <= height/2+40:
```

```
    pygame.draw.rect(screen,color_light,[width/2,height/2,160,40])
```

```
else:
```

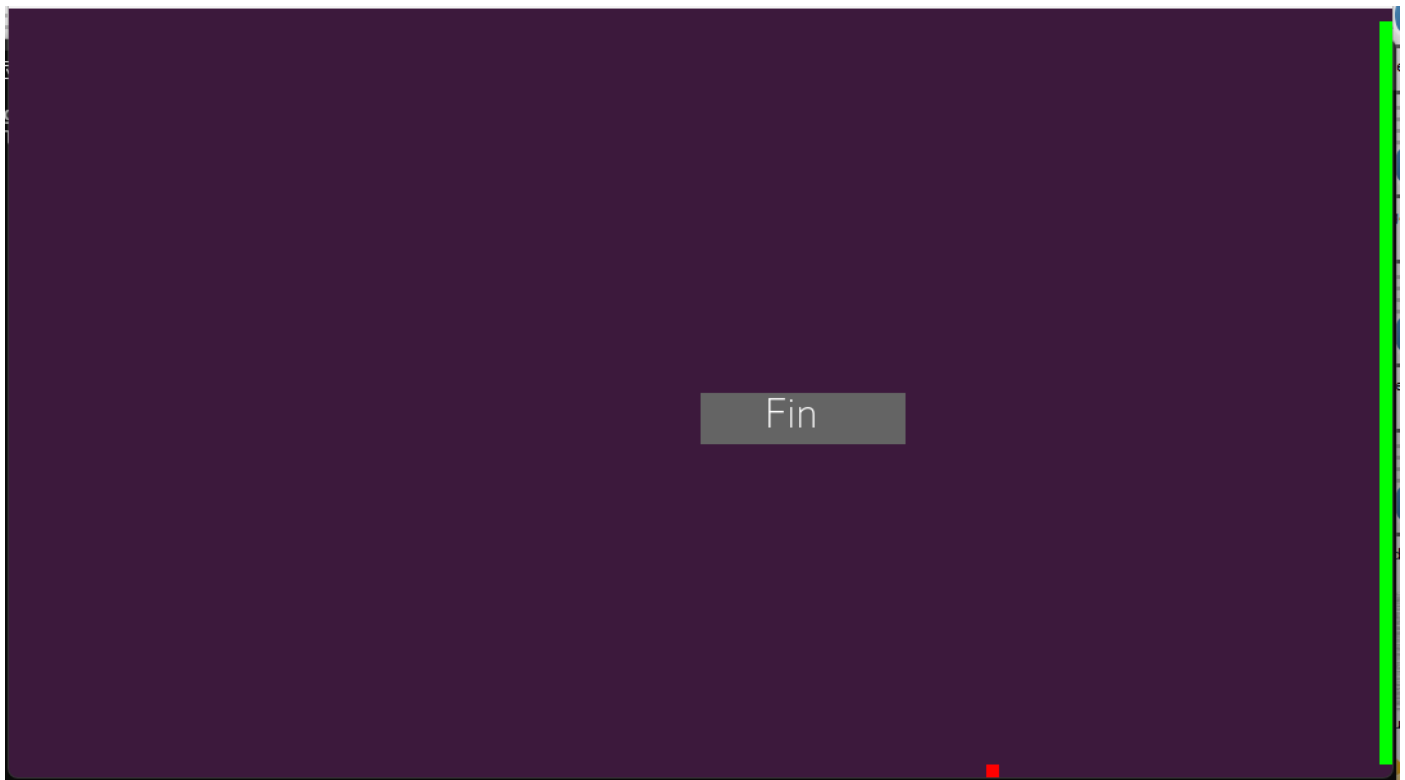
```
    pygame.draw.rect(screen,color_dark,[width/2,height/2,160,40])
```

```
# affichage du tete sur le bouton
```

```
screen.blit(text , (width/2+50,height/2))
```

```
# update
```

```
pygame.display.update()
```



Notez que si l'on passe le curseur de la souris sur le bouton Fin, sa couleur s'éclaircit.

Notez aussi que :

Si l'on clique sur le bouton Fin, on sort de l'application.

Si l'on désire simplement qu'en cliquant sur le bouton on mette fin au mouvement du carré rouge, sans sortir de l'application , il faudra modifier le code ainsi (on sortira alors de l'application par la méthode classique : click sur la croix en haut à droite) :

```
import pygame
```

```
import sys
```

```
import time
```

```
pygame.init()          # initialise
```

```
res = (1080,600)       # screen resolution
```

```
screen = pygame.display.set_mode(res)
```

```
pygame.display.set_caption("Ma belle fenêtre")
```

```
color = (255,255,255)   #blanc
```

```
vert=(0,255,0)          #vert
```

```
color_light = (170,170,170) # pâle
```

```
rouge = (255,0,0)       # rouge
```

```
color_dark = (100,100,100) # foncé
```

```
width = screen.get_width()    # largeur de l'écran
height = screen.get_height()  # hauteur de l'écran
```

```
smallfont = pygame.font.SysFont('Corbel',35) # fonte
text = smallfont.render('Fin ', True , color)
```

```
# carré rouge mobile
rect =pygame.draw.rect(screen,rouge, [10,10,10,10])
```

```
hh=1
```

```
vv=1
```

```
cesse=0 # ajout
```

```
while True:
```

```
    for ev in pygame.event.get():
        if ev.type == pygame.QUIT:
            pygame.quit()
        # click souris
        if ev.type == pygame.MOUSEBUTTONDOWN:
            if width/2 <= mouse[0] <= width/2+160 and height/2 <= mouse[1] <=
                                                    height/2+40:
                cesse=1 # ajout
```

```
time.sleep(0.010)
```

```
screen.fill((60,25,60))
```

```
if cesse == 0 : # ajout
```

```
    if vv>= 590 :
```

```
        vv=590
```

```
        hh+=1
```

```
    if hh >= 1060 :
```

```
        hh=1060
```

```
        vv-=1
```

```
    if vv <= 0 :
```

```
        vv=0
```

```
        hh-=1
```

```
    if hh<=0 :
```

```
        hh=0
```

```
        vv+=1
```

```
    if hh !=1060 and vv !=590 and vv != 0 and hh !=0:
```

```
        hh+=1
```

```
        vv+=1
```

```
rect=pygame.draw.rect(screen,rouge, [hh,vv,10,10])
pygame.draw.rect(screen,vert, [1070,10,10,580])
```

```
mouse = pygame.mouse.get_pos() # coord souris
# quand souris passe sur le bouton
if width/2 <= mouse[0] <= width/2+140 and height/2 <= mouse[1] <=
                                                                    height/2 + 40 :
    pygame.draw.rect(screen,color_light,[width/2,height/2,160,40])
else:
    pygame.draw.rect(screen,color_dark,[width/2,height/2,160,40])
```

```
# affichage du texte sur le bouton
screen.blit(text , (width/2+50,height/2))
```

```
# update
pygame.display.update()
```