Pygame : les transformations géométriques

Présentation

Le module **transform** de Pygame permet de créer de **nouvelles surfaces Pygame** en appliquant à une surface source une ou plusieurs transformations, isométriques ou non.

La fonction flip

```
pygame.transform.flip(Surface, xbool, ybool)
```

La fonction flip permet de faire une symétrie axiale, verticale ou horizontale ou les deux.

Elle prend en paramètre :

- la surface que l'on va transformer,
- un booléen indiquant que l'on procède à une symétrie selon l'axe vertical en indiquant True,
- un booléen indiquant que l'on procède à une symétrie selon l'axe horizontal en indiquant True.

Si les deux booléens sont à True, alors on procède aux deux symétries.

Exemple:

```
IMG_FRAISE = pygame.image.load("fraise.png")
IMG_FRAISE = pygame.transform.flip(IMG_FRAISE, True, True)
```

La fonction rotate

```
pygame.transform.rotate(Surface, angle)
```

La fonction **rotate** permet de faire faire une rotation à la surface :

 l'angle passé en paramètre est exprimé selon le sens antihoraire, en degrés, et non en radians comme c'est parfois le cas pour d'autres fonctions Pygame (le tracé d'arcs d'ellipse notamment).

Lien: https://stackoverflow.com/questions/4183208/how-do-i-rotate-an-image-around-its-center-using-pygame

La fonction scale

```
pygame.transform.scale(Surface, (width, height), DestSurface = None)
```

La fonction scale permet de redimensionner une surface. On passe en paramètre la hauteur et la largeur de la surface cible.

Le programme complet

```
import pygame
pygame.init()
HAUTEUR = 400
LARGEUR = 400
COULEUR_FOND = (255, 255, 255)
ECRAN = pygame.display.set_mode((LARGEUR, HAUTEUR))
ARRET = False
# image FRAISE
x_fraise = 50
y_fraise = 50
IMG_FRAISE = pygame.image.load("fraise.png").convert_alpha()
#IMG_FRAISE = pygame.transform.flip(IMG_FRAISE, True, True)
while not ARRET:
```

```
pygame.time.delay(100)
ECRAN.fill(COULEUR FOND)
for event in pygame.event.get():
  if event.type == pygame.QUIT:
    sys.exit()
  elif event.type == pygame.KEYDOWN:
    # symétrie axiale
   if event.key == pygame.K_RIGHT:
      IMG\_FRAISE = pygame.transform.flip(IMG\_FRAISE, \ \textbf{True}, \ \textbf{False})
    elif event.key == pygame.K_UP:
     IMG_FRAISE = pygame.transform.flip(IMG_FRAISE, False, True)
    # rotation
    elif event.key == pygame.K_LEFT:
     IMG_FRAISE = pygame.transform.rotate(IMG_FRAISE, 10)
    # redimensionner
    elif event.key == pygame.K_DOWN:
      IMG_FRAISE = pygame.transform.scale(IMG_FRAISE, (40,40))
  ECRAN.blit(IMG_FRAISE, (x_fraise, y_fraise))
  pygame.display.update()
```

Les activités ...

Je reviens à la liste des activités.

https://repl.it/@charlestecher/Transformations-geometriques <html>

</html>

Les activités ...

Je reviens à la liste des activités.

From:
/- Les cours du BTS SIO

Permanent link:
/doku.php/isn/pygame03

Last update: 2019/11/14 11:17

