

Projet - Création d'un morpion avec pygame

1 Introduction à pygame

1.1 Initiation

Exécuter et analyser ce code :

```
1 import pygame
2
3 surf = pygame.display.set_mode((800,600))
4 pygame.quit()
```

Exécuter et analyser ce code :

```
1 import pygame
2
3 surf = pygame.display.set_mode((800,600))
4 run = True
5 while run :
6     for event in pygame.event.get():
7         if event.type == pygame.QUIT:
8             run = False
9 pygame.quit()
```

1.2 Dessiner avec pygame

Exécuter et analyser ce code :

```
1 import pygame
2
3 surf = pygame.display.set_mode((800,600))
4 run = True
5 while run :
6     for event in pygame.event.get():
7         if event.type == pygame.QUIT:
8             run = False
9     pygame.draw.line(surf,(255,255,255),(10,20),(150,200),2)
10    pygame.display.flip()
11 pygame.quit()
```

Exécuter et analyser ce code :

```
1 import pygame
2
3 surf = pygame.display.set_mode((800,600))
4 run = True
5 posX = 50
6 while run :
7     for event in pygame.event.get():
8         if event.type == pygame.QUIT:
```

```
9     run = False
10    pygame.draw.circle(surf, (255,0,0), (posX, 300), 30, 2)
11    posX = posX + 1
12    pygame.display.flip()
13    pygame.quit()
```

Exécuter et analyser ce code :

```
1  import pygame
2
3  surf = pygame.display.set_mode((800,600))
4  run = True
5  posX = 50
6  while run :
7      for event in pygame.event.get():
8          if event.type == pygame.QUIT:
9              run = False
10     surf.fill((0,0,0))
11     pygame.draw.circle(surf, (255,0,0), (posX, 300), 30, 2)
12     posX = posX + 1
13     pygame.display.flip()
14     pygame.quit()
```

1.3 Gestion des images

Exécuter et analyser ce code, après avoir placé l'image `pyg.png` dans le repertoire courant :

```
1  import pygame
2
3  surf = pygame.display.set_mode((800,600))
4  run = True
5  img = pygame.image.load("pyg.png")
6  while run :
7      for event in pygame.event.get():
8          if event.type == pygame.QUIT:
9              run = False
10     surf.fill((0,0,0))
11     surf.blit(img,(200,200))
12     pygame.display.flip()
13     pygame.quit()
```

ding42 Voir la documentation internet pour gérer la transparence des images.

1.4 Gestion d'événements de type Souris

```
1  import pygame
2
3  surf = pygame.display.set_mode((800,600))
4  run = True
5  while run :
6      for event in pygame.event.get():
7          if event.type == pygame.QUIT:
8              run = False
9          if event.type == pygame.MOUSEBUTTONDOWN :
10             if pygame.mouse.get_pressed() == (1,0,0) :
11                 print ("clik bouton gauche")
12             if pygame.mouse.get_pressed() == (0,0,1) :
13                 print ("clik bouton droit")
```

```
14 surf.fill((0,0,0))
15 pygame.display.flip()
16 pygame.quit()
```

```
1 import pygame
2
3 surf = pygame.display.set_mode((800,600))
4 run = True
5 while run :
6     for event in pygame.event.get():
7         if event.type == pygame.QUIT:
8             run = False
9         if event.type == pygame.MOUSEBUTTONDOWN :
10            if pygame.mouse.get_pressed() == (1,0,0) :
11                pos = pygame.mouse.get_pos()
12                print(pos)
13 surf.fill((0,0,0))
14 pygame.display.flip()
15 pygame.quit()
16
```

1.5 Gestion d'événements de type clavier

```
1 import pygame
2
3 surf = pygame.display.set_mode((800,600))
4 run = True
5 while run :
6     for event in pygame.event.get():
7         if event.type == pygame.QUIT:
8             run = False
9         if event.type == pygame.KEYDOWN :
10            if event.key == pygame.K_SPACE :
11                print ("vous avez appuyé sur la touche espace")
12            elif event.key == pygame.K_a :
13                print ("vous avez appuyé sur la touche A")
14            elif event.key == pygame.K_RETURN :
15                print ("vous avez appuyé sur la touche Entrée")
16            else :
17                print ("vous avez appuyé sur une touche")
18 surf.fill((0,0,0))
19 pygame.display.flip()
20 pygame.quit()
```

1.6 Aide

N'hésitez pas à consulter l'aide officielle de pygame : <https://www.pygame.org/docs/>

2 Le projet

2.1 Difficulté 1

cf. vidéo

Cahier des charges :

- Une fenêtre s'ouvre avec une grille de 9 cases. Vous gérez les couleurs comme bon vous semble.
- lorsque l'on clique sur une des 9 cases , une croix s'affiche dans cette case.
- si on recommence à cliquer, un rond s'affiche dans une autre case. Et ainsi de suite : à chaque click, on affiche alternativement une croix et un rond.
- ⚠ Il est impossible de cliquer de nouveau sur une case qui contient déjà un rond ou une croix (en tout cas, cela n'a pas d'effet)
- Les images 'rond' et 'croix' auront été réalisées par vos soins, en utilisant un logiciel de traitement d'images, comme Gimp (qui sait très bien gérer la transparence par exemple).
- Lorsque Croix ou rond gagne, le jeu se fige et tout clique n'a plus aucun effet
- En cas d'égalité, une fois les 9 cases remplies, le jeu se fige aussi.

Pour vous aider , voici une structure possible du programme :

```
import pygame

def nb(tour):
    ''' permet d'afficher une croix ou un rond en fonction
    du tour (tour et un entier entre 0 et 9)
    '''
    pass

def somme(t):
    ''' cette fonction prend un tableau en paramètre
    de la forme morpion = [[0,0,0],[0,0,0],[0,0,0]]
    au fur et à mesure de la partie, les 0 sont
    remplacées par de 1 (croix) ou des -1 (ronds)
    Cette fonction renvoie 1 si les croix gagnent, -1 si les ronds gagnent
    '''
    pass

tour = 0
morpion = [[0,0,0],[0,0,0],[0,0,0]]

continuer = True
jeu = True
while continuer:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            continuer = False
    while jeu :
        # dessin de la grille

        # gestion du click sur une case et affichage de l'image au bon
        endroit

        # gestion de la fin du jeu
```

2.2 Niveau de difficulté 2

cf. vidéo

- On ajoute ici un affichage dans la fenêtre : "Croix gagne", "Rond gagne" ou "égalité".
- Les croix, les ronds ne sont pas toujours identiques.

2.3 Niveau de difficulté 3

(pas de son)

- On peut ici jouer plusieurs parties.
- A la fin de chaque partie, on a un affichage qui donne le vainqueur ou affiche égalité. Un click sur la fenêtre permet de redémarrer un nouveau jeu.
- En cas de victoire (des ronds ou des croix), les cases qui ont permis l'alignement sont mises en reliefs.