

Projet – Un jeu avec Pygame

Cette séquence s'appuie sur :

- https://pixees.fr/informatiquelycee/n_site/nsi_term_projet_3.html

1 Rappel pygame

Un premier projet sur l'utilisation de pygame se trouve ici : https://ninoofr/LC/Term_NSI/seqX_projet_pygame/utilisation_pygame_et_projet.pdf

1.1 Affichage et rafraîchissement écran

L'affichage de l'image se fait en 2 étapes :

- création d'un objet de type image (`img` dans notre exemple) à l'aide de la méthode `load`. cette méthode `load` prend un seul paramètre : l'url de l'image
- utilisation de la méthode `blit` pour afficher l'image (la méthode `blit` doit être utilisée dans la "boucle de jeu"). La méthode `blit` prend 2 paramètres : l'objet image à afficher (dans notre cas `img`) et un tuple qui correspond aux coordonnées du coin haut-gauche de l'image ((200,200) dans notre exemple)

1.2 Utilisation souris

Il est possible de gérer les événements "clavier" et "souris".

Il existe un événement `pygame.MOUSEBUTTONDOWN` qui correspond à un clic de souris. La méthode `get_pressed` renvoie un tuple constitué de 3 éléments. En l'absence de clic de souris, ce tuple est (0,0,0). En cas de clic sur le bouton gauche de la souris le tuple est (1,0,0). En cas de clic sur le bouton central le tuple est (0,1,0). En cas de clic sur le bouton droit le tuple est (0,0,1)

A faire vous même 1.

Saisissez, analysez et testez ce code. Cliquez avec votre souris (clic gauche et clic droit) dans la fenêtre `pygame`, observez attentivement la console

```
import pygame
surf = pygame.display.set_mode((800,600))
run = True
while run :
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            run = False
        if event.type == pygame.MOUSEBUTTONDOWN :
            if pygame.mouse.get_pressed() == (1,0,0) :
                print ("clic bouton gauche")
            if pygame.mouse.get_pressed() == (0,0,1) :
                print ("clic bouton droit")
    surf.fill((0,0,0))
    pygame.display.flip()
pygame.quit()
```

Il est aussi possible de récupérer les coordonnées du pointeur de la souris au moment du clic à l'aide de la méthode `mouse.get_pos`. Cette méthode renvoie un tuple (x,y) => coordonnées du pointeur de la souris au moment du clic

1.3 Utilisation clavier

Il est aussi possible de gérer des événements de type "clavier" :

A faire vous même 2.

Saisissez, analysez et testez ce code, observez la console lorsque vous appuyez sur les touches "Entrée", "Espace" et "A" (vous pouvez aussi constater qu'il ne se passe rien quand)

```
import pygame

surf = pygame.display.set_mode((800,600))
run = True
while run :
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            run = False
        if event.type == pygame.KEYDOWN :
            if event.key == pygame.K_SPACE :
                print ("vous avez appuyé sur la touche espace")
            elif event.key == pygame.K_a :
                print ("vous avez appuyé sur la touche A")
            elif event.key == pygame.K_RETURN :
                print ("vous avez appuyé sur la touche Entrée")
            else :
                print ("vous avez appuyé sur une touche")
    surf.fill((0,0,0))
    pygame.display.flip()
pygame.quit()
```

Vous trouverez la liste complète des touches dans la [documentation officielle de pygame](#).

2 Pygame et programmation objet

Il est possible de gérer chaque objet animé grâce à la programmation objet.

A faire vous même 3.

Saisissez, analysez et testez ce code

```
import pygame
import wx

#img_fond = pygame.image.load("paysage.png")

def get_screen_size():
    app = wx.App(False) # the wx.App object must be created first.
    size = wx.GetDisplaySize()
    return size.GetWidth(), size.GetHeight()

class Objet:
    def __init__(self, l, h) -> None:
        self.mvt = 10
        self.img = pygame.image.load("logo_cordeliers_transparent.png")
        self.pos = [l, h]
    def calcul_future_pos(self, dx, dy):
        self.pos[0] = self.pos[0]+dx
        self.pos[1] = self.pos[1]+dy

SCREEN_L, SCREEN_H = get_screen_size()
SCREEN_L, SCREEN_H = round(0.75*SCREEN_L), round(0.75*SCREEN_H)

surf = pygame.display.set_mode((SCREEN_L, SCREEN_H))
run = True
clock=pygame.time.Clock()
les_objets = [] #Liste avec tous les objets à animer
les_objets.append(Objet(50,50))

while run :
    clock.tick(10)
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            run = False
        if event.type == pygame.KEYDOWN :
            if event.key == pygame.K_SPACE :
                print ("vous avez appuyé sur la touche espace")
            elif event.key == pygame.K_RIGHT :
                print ("vous avez appuyé sur la touche flèche droite")
            elif event.key == pygame.K_LEFT :
                print ("vous avez appuyé sur la touche flèche gauche")
            elif event.key == pygame.K_q :
                run = False
            else :
                print ("vous avez appuyé sur une touche")
    surf.fill((150,150,150))
    for obj in les_objets:
        surf.blit(obj.img,obj.pos)
        obj.calcul_future_pos(5,5)
    pygame.display.flip()
pygame.quit()
```

3 Le projet

Avec pygame, programmez un jeu pygame

Objectif :

1. Déterminez les rôles de chacun des élèves
2. Déterminez comment les fichiers seront échangés
3. Établissez un planning
4. Écrivez un 1^{er} document expliquant les grands principes du jeu, quelques schémas et aperçus du jeu
5. Reprenez le script ci-dessus et adaptez la classe Objet pour vos besoins
6. Développez votre jeu sur ces principes

Grille d'évaluation :

Investissement sur le projet / Implication en classe (2 pts)	Aucun investissement	Pas de présentation du projet, pas de demande d'aide	Présentation de l'avancement. Des questions posées	De nombreuses suggestions Investissement important	Tout le projet terminé en classe
Gestion de projet et document associé	Rien d'écrit	Des pratiques improvisés et des données partiels	Des pratiques correctes et des données avec quelques manques	De bonnes pratiques et des données complètes	De très bonnes pratiques et des données complètes et très bien présentées
Réalisation technique (2 pts)	Rien d'écrit	Ne fonctionne pas	Fonctionne imparfaitement	Fonctionne sans aide	Fonctionne de manière optimale
Oral / Explications / Réponse aux questions (4 pts)	Rien d'exprimé	Explications approximatives Nombreuses questions sans réponse	Explications approximatives Quelques questions sans réponse	Explications pertinentes Pas de question sans réponse	Explications pertinentes avec support écrit