

Pygame

Pygame est une bibliothèque qui peut être rajoutée à Python pour afficher des images, gérer les sons et interagir avec l'écran. Pygame est donc particulièrement adapté pour développer des jeux programmés en Python.

Installation :

Ce n'est pas un module intégré dans l'environnement Python de base du type "IDLE". Il faut donc le télécharger. C'est un logiciel libre.

Attention !!! Pygame ne fonctionne qu'avec certaines versions de Python et uniquement en **32 bits**.

Pour Windows, utiliser Python 3.2 en 32 bits.

Pour Mac, utiliser Python 2.7 en 32 bits.

Utilisation :

Les images et les sons utilisés doivent être enregistrés dans le même fichier que le programme. Sinon le chemin doit être spécifié dans le programme.

"**Faire bouger**" un personnage sur pygame consiste à superposer rapidement plusieurs images du personnage en décalant sa position et à intercaler entre ces image un fond qui cache la position précédente. Pour "**coller**" une image sur le fond, on utilise "blit". C'est une copie pixel par pixel.

Après chaque copie, il faut **rafraichir l'écran** pour afficher la nouvelle image. On utilise display.flip()

Quelques commandes : (les mots en italique sont les variables)

	comandes
Importer pygame :	import pygame
et initialiser les modules	pygame.init()
Création d'une fenetre Pygame de 450x450 pixels stockée dans la variable f.	<i>f</i> = pygame.display.set_mode((450, 450))

Chargement du fond stocké dans la var fond	fond = pygame.image.load("fond.jpg").convert()
Collage du fond, le coin en haut à gauche de notre fenêtre, aux coordonnées (0;0)	f.blit(fond, (0,0))
Rafraîchissement de l'écran	pygame.display.flip()
Création de l'objet son	<i>son</i> = pygame.mixer.Sound("goutte2.WAV")
Dessiner un cercle dans notre fenetre f, couleur (r,v,b) coordonnées du centre (x,y), rayon : r ;vide (0) ou plein (1)	pygame.draw.circle (f,(r,v,b),(x,y),r,0)
Dessiner un segment qui relie (x1,y1) à (x2,y2)	pygame.draw.line (f, (r,v,b),(x1,y1),(x2,y2),2)
recupère l'image "minion" et la stocke dans le var m. (convert_alpha pour la transparence)	m = pygame.image.load ("minion.bmp").convert_alpha()
crée une variable p qui contient l'image "minion" sous forme rectangulaire	p = m.get_rect()
colle l'image m en (0,0)	f.blit(m,(0,0))
si une commande du type touche est activée :	if event.type == pygame.KEYDOWN:
la touche (flèche bas) est utilisée (haut gauche droite)	if event.key == pygame.K_DOWN: (ou K_UP, K_LEFT; K_RIGHT)
la position rajoute 3 en abscisse.	p= p.move(0,3)

La boucle infinie qui permet de clore le programme :

```
continuer = 1
while continuer==1:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            continuer=0
            pygame.quit()
```