

Pygame-3



Pygame est une librairie qui fonctionne avec des évènements.

Mais qu'est-ce qu'un évènement?

Un évènement est simplement une action que l'utilisateur effectuera, par exemple cela peut être « Appuyez sur la touche E » ou encore « Bouger la souris ». Pour cela, Pygame enregistre dans une liste tous les nouveaux évènements lorsqu'il y en a un et de manière asynchrone. Ces évènements sont alors enregistrés dans la « variable » : `pygame.event.get()`

Evènements liés au clavier

Tous les programmes utilisant pygame fonctionnent de la même manière, ils utilisent une boucle infinie du type : `while game_over`.

C'est-à-dire que le jeu va exécuter notre boucle à l'infini jusqu'à ce qu'une variable « continuer » ou « `game_over` » passe à 0. Ce sera notre condition d'arrêt de notre jeu.

Voici un morceau de code de « base » :

```
while 1: #boucle infinie
    # récupération des évènements
    for event in pygame.event.get():
        if event.type==pygame.QUIT: # sortie du programme
            sys.exit()
```

Analysons ce bout de code :

- Tout d'abord on crée la boucle infinie.
- Ensuite, on utilise une boucle `for`, pour parcourir tous les évènements reçus grâce à la fonction `get()` du module "event" de Pygame. Cette fonction retourne une liste d'objets `Event`, pour lesquels on peut connaître le type, la touche enfoncée si c'est au clavier, la position du curseur si c'est un clic, etc...
- La condition teste si l'évènement est de type `QUIT` (c'est à dire un `Alt+F4` ou un clic sur le bouton de fermeture)
- Si la condition est satisfaite, on demande à la boucle de s'arrêter.

Pygame gère de nombreux éléments, en voici une liste non exhaustive : (Touche du clavier)

<u>Lettres:</u> K_a ... K_z <u>Nombres:</u> K_0 ... K_9 <u>Controles:</u> K_TAB K_RETURN K_ESCAPE K_SCROLLOCK K_SYSREQ K_BREAK K_DELETE K_BACKSPACE K_CAPSLOCK K_CLEAR K_NUMLOCK <u>Parenthèses:</u> K_RIGHTBRACKET, K_LEFTBRACKET K_RIGHTPAREN, K_LEFTPAREN	<u>Ponctuation:</u> K_SPACE K_PERIOD K_COMMA K_QUESTION K_AMPERSAND K_ASTERISK K_AT K_CARET K_BACKQUOTE K_DOLLAR K_EQUALS K_EURO K_EXCLAIM K_SLASH, K_BACKSLASH K_COLON, K_SEMICOLON K_QUOTE, K_QUOTEDBL K_MINUS, K_PLUS K_GREATER, K_LESS <u>Touches F:</u> K_F1 ... K_F15 <u>Touches d'édition:</u> K_HELP K_HOME K_END K_INSERT K_PRINT K_PAGEUP, K_PAGEDOWN K_FIRST, K_LAST	<u>SHF,CTL,ALT etc:</u> K_LALT, K_RALT K_LCTRL, K_RCTRL K_LSUPER, K_RSUPER K_LSHIFT, K_RSHIFT K_RMETA, K_LMETA <u>Flèches:</u> K_LEFT K_UP K_RIGHT K_DOWN <u>Autres:</u> K_MENU K_MODE K_PAUSE K_POWER K_UNDERSCORE K_HASH <u>Clavier numérique:</u> K_KP0 ... K_KP9 K_KP_DIVIDE K_KP_ENTER K_KP_EQUALS K_KP_MINUS K_KP_MULTIPLY K_KP_PERIOD K_KP_PLUS
---	---	--

Voici comment utiliser ces constantes du clavier pour détecter l'appuie sur une touche:

```
# récupération des touches du clavier pour mouvement vaisseau
keys = pygame.key.get_pressed()
dx, dy = 0, 0
if keys[pygame.K_LEFT]:
    dx = -10
if keys[pygame.K_RIGHT]:
    dx = 10
if keys[pygame.K_UP]:
    dy = -10
if keys[pygame.K_DOWN]:
    dy = 10
```

Maintenant, passons aux évènements liés à la souris.

Evènements liés à la souris

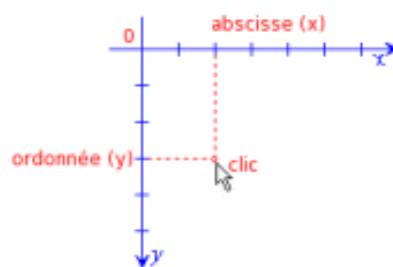
Le type d'événement créé lors d'un clic est `MOUSEBUTTONDOWN`, (ou `MOUSEBUTTONUP` au relâchement du bouton).

Un événement de souris possède deux attributs : le bouton (`event.button`) et la position du clic dans la fenêtre (`event.pos`).

`event.button` peut prendre les valeurs suivantes :

1	Bouton gauche
2	Bouton milieu (ou droite + gauche)
3	Bouton droit
4	Molette haut
5	Molette bas

`event.pos`, lui, renvoie un tuple contenant l'abscisse et l'ordonnée à partir de l'angle haut-gauche, c'est-à-dire le bout de la pointe de la flèche.



On peut donc imaginer de faire un script permettant d'afficher un message sur la console lorsque l'on clique droit sur une partie de l'écran. Ici nous prendrons la partie en haut à droite.

```
if event.type == MOUSEBUTTONDOWN and event.button == 3 and event.pos[1]<150 and event.pos[0]>100:  
    print("Zone dangereuse")
```

Le code précédent permet donc d'afficher « Zone dangereuse », lorsque l'on clique (`event.type == MOUSEBUTTONDOWN`) droit (`event.button == 3`) avec la souris, en haut à droite, entre `y<150` (`event.pos[1] < 150`) et `x>100` (`event.pos[0] > 100`) ;