

Gestion des événements

Lorsqu'un programme `pygame` est lancé, la variable interne `pygame.event.get()` reçoit en continu les événements des périphériques gérés par le système d'exploitation.

Dans les exemples précédents, on a uniquement récupéré l'événement de type `QUIT` (clic sur la croix de fermeture de la fenêtre) pour stopper la boucle et terminer le programme:

```
for evenement in pygame.event.get():
    if evenement.type == QUIT:
        continuer = False
```

Nous allons maintenant nous intéresser aux événements de type `KEYDOWN` (touche de clavier appuyée) ou de type `MOUSEBUTTONDOWN` (boutons de souris appuyés).

1. Événements clavier

!!! code “Structure de code” On contrôle si l'événement est de type `KEYDOWN`, puis on détermine l'instruction à exécuter en fonction de l'attribut `key` de l'événement (c'est-à-dire le code de la touche):

```
```python linenums='1'
for evenement in pygame.event.get():
 if evenement.type == KEYDOWN:
 if evenement.key == K_RIGHT:
 print("flèche droite appuyée")
...
```
```

Remarques:

- la liste des codes des touches se retrouve ici : <https://www.pygame.org/docs/ref/key.html>{:target="__blank"}
- rémanence des touches : si on veut que l'action associée à l'appui d'une touche se répète (par ex. toutes les 50 millisecondes) tant que la touche reste enfoncée, on utilise:

```
pygame.key.set_repeat(50)
```

!!! exemple “À vous de jouer” Écrire un programme qui déplace un personnage dans la fenêtre, piloté pour les quatre touches directionnelles du clavier.

2. Événements souris

!!! code “Structure de code” On contrôle cette fois que l'événement est de type `MOUSEBUTTONDOWN`, puis on détermine l'instruction à exécuter en fonction de l'attribut `button` de l'événement (1 pour bouton gauche, 2 pour bouton droit, 3 à 5 pour la molette).

```
```python linenums='1'
for evenement in pygame.event.get():
 if evenement.type == MOUSEBUTTONDOWN:
 if evenement.button == 1:
 print("clic gauche détecté")
...`
```

**Remarque:** on récupère le tuple des coordonnées de la souris par l'instruction `pygame.event.get_pos()`.