

# Interface graphique Pygame pour Python

---

## 1) Présentation

La bibliothèque **Pygame** est assez intuitive et constitue un très bon moyen de se lancer dans la programmation graphique avec Python.

Pygame ne fait pas partie des bibliothèques standards de Python, elle est donc développée indépendamment et il est nécessaire de la télécharger et de l'installer en plus de Python pour pouvoir l'utiliser. Bonne nouvelle : Pygame est installé avec **Edupython**.

## 2) Premières fenêtres

La **documentation** officielle Pygame est disponible sur <http://pygame.org/docs> (en anglais)

Ici en PDF : <https://readthedocs.org/projects/pygame/downloads/pdf/latest/> (en anglais)

Et aussi en français : <https://fr.wikibooks.org/wiki/Pygame>

Donc, inutile de refaire un cours, il suffit de lire la doc.

Pygame est en fait composé de plusieurs modules, dont parmi des plus importants :

- display
- mixer
- draw
- event
- image
- mouse
- time

On peut importer les modules un par un. Par exemple, pour jouer un son, nous avons seulement besoin du module « mixer ». Pour importer le module mixer il faut écrire :

```
import pygame.mixer
```

On peut aussi importer tous les modules présents dans la bibliothèque Pygame :

```
import pygame
```

Il faut aussi importer les constantes de Pygame (vous verrez plus tard à quoi ça sert) :

```
from pygame.locals import *
```

Voilà, c'est tout, vous pouvez utiliser Pygame, mais avant de faire quoi que ce soit il faut initialiser :

```
pygame.init()
```

Une fois Pygame initialisé, on peut créer une fenêtre ( qui sera vide pour commencer ) :

```
fenetre = pygame.display.set_mode((300, 400))
```

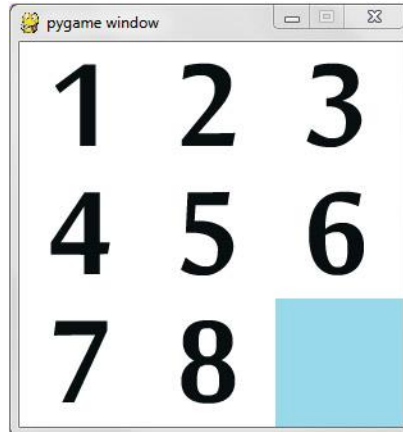
La fenêtre apparaît et disparaît aussitôt ! En effet, Pygame affiche la fenêtre, et une fois qu'il a terminé, il considère que le travail est fait et continue le script. Pour rester ouverte, la fenêtre a donc besoin d'une boucle infinie.

Pour en savoir plus sur les fenêtre : <https://openclassrooms.com/fr/courses/1399541-interface-graphique-pygame-pour-python/1399813-premieres-fenetres>

### 3) Exemple : programme puzzle

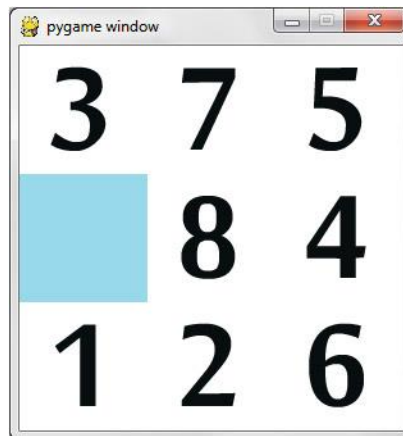
Le programme puzzle téléchargeable ici : <http://www.info-nsi.fr/nsi-python/puzzle.zip>

Ce programme crée un plateau avec 8 pièces de puzzle correctement placées :



Les pièces peuvent être déplacées en cliquant dessus. Mais ça ne sert à rien vu que le puzzle est déjà fini.

**Travail à faire** : modifier le programme en ajoutant, dans le programme, un mélange aléatoire des pièces.



**Principe** : on choisit «  $n$  fois » un nombre aléatoire de 1 à 9. A chaque fois, si la pièce choisie aléatoirement peut être déplacée alors elle est déplacée. Si  $n$  très grand alors le puzzle sera mélangé.

Une fois le puzzle mélangé le joueur pourra jouer.

**Conseil** : passer du temps sur programme puzzle qui contient de nombreux commentaires.