

Lien GitHub : <https://github.com/Laduff/MacGyver>

## Introduction

Ce document résume l'ensemble de mon projet numéro 3 réalisé pour ma formation de développeur d'application Python sur Openclassrooms.

Pour réaliser ce projet, mon mentor m'a conseillé de fonctionner de cette façon :

- Créer un fichier .txt qui sera l'architecture du labyrinthe.
- Créer des fichiers .py correspondants aux classes définies.
- Créer un fichier .py centralisant tous les fichiers et contenant la boucle du jeu.

## Mon programme

Le fichier maze.txt me permet d'architecturer le labyrinthe en symbolisant les murs par des « X », le chemin par des « O », MacGyver par « D » et le méchant par « F ».

Les fichiers « maze.py », « macgyver.py » et « py\_game.py » comprennent leurs classes respectives. Cela me permet de les « appeler » dans le fichier principal (GAME.py).

Le fichier GAME.py centralise tout, il permet d'initialiser tout le contenu du jeu en statique et grâce à la boucle while permet de faire « vivre » le labyrinthe.

## Problèmes rencontrés

N'ayant jamais programmé, le plus dur au début a été d'imaginer l'architecture de programmation du jeu.

Je me suis surtout appliqué à apprendre un maximum ce langage de programmation, je me suis formé sur plein de sites différents : Codingame, codecademy, Udemy... Openclassroom aussi bien sûr !

Ensuite un collègue d'openclassroom m'a conseillé ce cours :

<https://openclassrooms.com/courses/interface-graphique-pygame-pour-python>

Et là bingo ! Tout était très bien expliqué, clair comme de l'eau de roche et de plus l'activité était de programmer un labyrinthe avec Donkey Kong !

Après avoir fini ce cours j'ai retranscrit ce que j'avais appris sur mon projet. Grâce à cela j'ai pu construire mon labyrinthe et en voir les couleurs.

Mes plus gros problèmes étaient surtout liés à des petites erreurs d'indentation, de rafraîchissement d'images et etc... Il y avait aussi le choix des algorithmes qui était un vrai casse-tête pour un débutant comme moi.

Cela m'a malheureusement fait perdre beaucoup de temps...

## Les bénéfices de ce projet

Ce projet m'a permis de comprendre :

- L'architecture d'un programme.
- Le choix des algorithmes.
- L'utilisation des « class ».
- L'interface Pygame.

M'a permis également d'améliorer mes compétences en langage python grâce aux multiples échecs/corrections de ma part.

## Mes axes d'amélioration

Pour les prochaines fois il faudra que je pense à commit plus régulièrement l'avancée de mon projet. J'étais trop focalisé sur la construction du projet.

Il faudra également que j'améliore mes connaissances en python et mon anglais pour mieux comprendre les exercices et les cours sur le net.