

Apprentissage axé sur un projet :

Aidez MacGyver à s'échapper !

Vue d'ensemble du projet :

Dans cet exercice, l'étudiant est amené à apprendre les bases du langage Python ; c'est-à-dire : son écriture, sa logique mais aussi ses facilités ainsi que ses difficultés.

Le devoir est présenté sous la forme d'un labyrinthe où MacGyver (notre héros) doit récupérer certains objets avant de se présenter devant le Gardien. Tout l'aspect visuel est créé avec le module Pygame ; l'étudiant devra en apprendre le b.a-ba.

Normes :

- Le code sera visionné en utilisant Git et sera publié sur Github pour que le mentor puisse le commenter.
- Je respecterai les bonnes pratiques de la PEP 8 et je développerai dans un environnement virtuel utilisant Python 3.
- Mon code devra être écrit en anglais : nom des variables, commentaires, fonctions....

Objectifs :

- Bien comprendre les fonctionnalités du langage et savoir les exploiter.
- Apprendre les bonnes coutumes d'un bon développeur. (Rédiger proprement son code, ...)
- Savoir bien rechercher la documentation et l'utiliser.

Spécifications/Tâche(s) :

Tâche 1 – Créer le cadre de départ :

Initialiser un repo Git et l'envoyer sur Github.

Commencer par créer le labyrinthe sans interface graphique. Quand la logique du labyrinthe est faite, utiliser le module PyGame pour dessiner l'interface graphique.

Puis s'intéresser aux trois éléments principaux du jeu : le gardien, MacGyver et les objets.

Comment les représenter dans le programme ? Où sont-ils placés au commencement du jeu ?

Tâche 2 – Animer le personnage :

Le seul élément mouvant est MacGyver. Créer les méthodes de classe qui permettent de l'animer et de trouver la sortie. Pour l'instant, faire une version simplifiée du jeu dans laquelle MacGyver gagne en arrivant face au gardien.

Tâche 3 – Récupérer les objets :

Ajouter la gestion des objets. Comment MacGyver les ramasse-t-il?
Ajouter également un compteur qui les listera.

Tâche 4 – Gagner ! :

Enfin, changer la fin du jeu : MacGyver gagne s'il a bien ramassé tous les objets et endormi le garde. Sinon, il perd.

Notes/résultats de recherches :

J'ai beaucoup utilisé la documentation de Python sur internet (<https://docs.python.org/fr/3/>) ; cela m'a permis de mieux comprendre ce que je faisais.

De plus un youtubeur m'a bien aidé à clarifier certaines choses.

(<https://www.youtube.com/watch?v=8J8wWxbAdFg&list=PLMS9Cy4Enq5KsM7GJ4LHn1BQKTQBV8kaR>)

Enfin pour customiser l'interface graphique de mon labyrinthe, j'ai utilisé des images libres de droit sur ce site : <https://www.flaticon.com>

Les étapes et les plans de mon projet :

Tout d'abord, lister les différentes « Classes » que je vais rencontrer dans le programme pour les isoler individuellement dans un fichier. Cela permet d'avoir une bonne organisation de travail.

Ensuite, réfléchir au cheminement de création de la Classe ; c'est-à-dire : ses « Objets » ainsi que ses « Attributs ».

Enfin, créer des « Méthodes » pour exploiter le travail effectué en amont.

Et pour terminer, tester et corriger la totalité du programme.

Résumé de ce que j'ai appris :

J'ai eu beaucoup de mal à comprendre la logique pour coder du Python, à tel point que j'ai arrêté le Projet 3 plus d'une fois. Malheureusement quand je revenais dessus, j'avais oublié se que j'avais appris et ça me démotivait de continuer.

A l'heure actuelle, je comprends d'avantage la façon de penser et d'écrire en Python mais il y a quand même une bonne marche avant que tout soit fluide.

De plus, j'ai pu découvrir le module PyGame et ce ne m'a pas déplu car ça changeait de l'ordinaire. Pour moi c'était une parenthèse amusante.

Un ensemble positif malgré tous les obstacles que j'ai vaincu avec mes mentors. (Surtout avec **Ranga Gonnage**)

Le lien vers mon projet :

https://github.com/R4mTex/P3_POUTOT_Simon.git