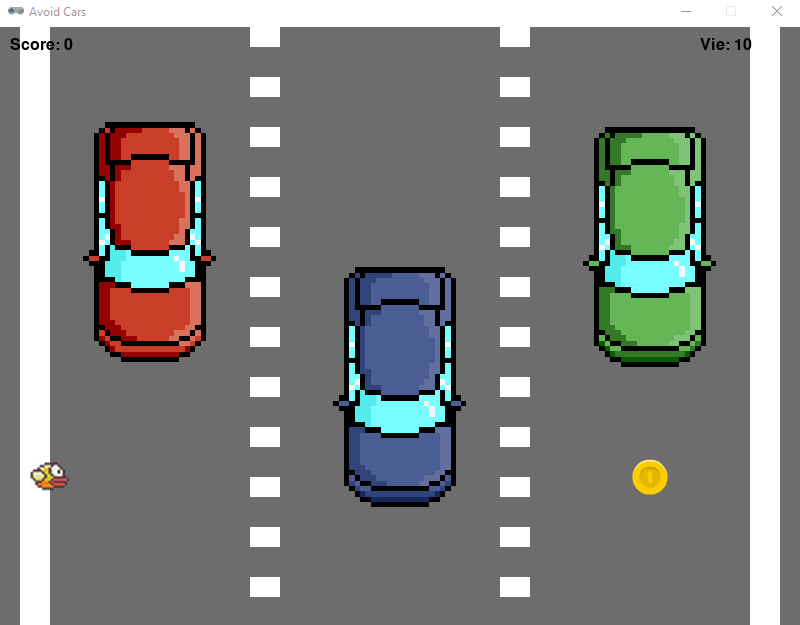
# Projet développement : jeu en python avec Pygame Zero

Je vous propose de créer un petit jeu en python avec l’aide de Pygame Zero. La librairie pgzhelper pourra également être utilisée pour compléter.



## Etape 1 : formation à pygame zero

Pour cela, suivez :

* l’introduction,
* l’installation
* la mise en place de la structure du projet (Setting up the folders)
* la partie « Gem Catcher » (sans les challenges)

sur le site <https://quirkycort.github.io/tutorials/20-Pygame-Zero-Basics/10-Intro/10-intro.html>

Ajoutez le fichier pgzhelper.py à votre projet, fichier disponible ici : <https://github.com/QuirkyCort/pgzhelper>

Ajoutez cette instruction en haut de votre fichier :

from pgzhelper import \*

## Etape 2 : codage et tests des fonctionnalités

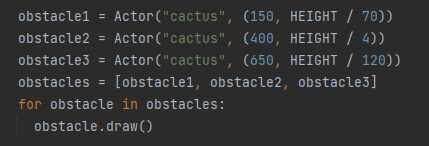
Une fois ce tutoriel effectué et compris, adaptez-le pour remplir les fonctionnalités suivantes :

* une route de 3 voies est dessinée en fond (fond gris avec un ensemble de rectangles blanc en réalité)
* des voitures y circulent, aléatoirement sur l’une ou l’autre des voies : d’abord 3 voitures puis d’autres apparaîtront selon le score obtenu
* le jeu commence avec 10 points de vie et un score de 0 pour l’utilisateur / personnage
* un personnage de votre choix apparaîtra au début du jeu, pourra entrer en collision avec une des voitures lors de ses déplacements et perdra alors un point de vie à chaque collision
* un personnage partira à la recherche d’un objet à ramasser qui lui permettront d’augmenter son score.
* le personnage pourra se déplacer de gauche à droite lors de l’appui sur une touche du clavier et il se tournera dans la direction dans laquelle il se dirigera.
* un son et / ou de la musique seront utilisés (lors du déroulement du jeu ou lors d’une collision par exemple)
* lorsque le jeu se termine (point de vie = 0) ou lorsque le score atteint un maximum à définir, le jeu s’arrête : un message l’indique à l’utilisateur. Il est alors possible de rejouer en appuyant sur une touche à définir.
* les points de vie et le score de l’utilisateur sont affichés à l’utilisateur tout au long du jeu.

Techniquement :

* Vous travaillez avec l’IDE PyCharm, comme nous l’avons déjà vu.
* Une liste de voitures est utilisée

Exemple de remplissage d’une liste et parcours d’une liste :



* Pour savoir si une touche est appuyée, le développeur indique une condition sur la touche appuyée :

Exemple pour tester si la flèche du haut est cliquée (code ajoutée dans la fonction update()):



* Votre code doit être propre et soignée. Vous séparerez votre code à l’aide de fonctions bien choisies dont chacune remplira un objectif (une fonction en fait).
* Une documentation complète de pygame zero se trouve ici :

<https://pygame-zero.readthedocs.io/en/stable/introduction.html>

* Tutorial au format pdf :

<https://readthedocs.org/projects/pygame-zero/downloads/pdf/latest/>

* D’autres exemples se trouvent ici :

<https://quirkycort.github.io/tutorials/20-Pygame-Zero-Basics/10-Intro/10-intro.html>

* Pour trouver des ressources (images, sons et musiques)

<https://kenney.nl/assets>

https://opengameart.org/art-search-advanced?keys=&field\_art\_type\_tid%5B0%5D=13&sort\_by=count&sort\_order=DESC&page=4

## Conditions

Ce projet sera fait de manière individuelle, il est possible de vous entraider pour vous débloquer mais en aucun cas, vous ne pourrez copier-coller ou reprendre le code de vos camarades tel quel.

Vous êtes cependant autorisés à vous inspirer du code de vos camarades et à l’adapter selon votre cas. Un commentaire viendra compléter ce code pour indiquer la source.

L’utilisation d’IA telles que ChatGPT est autorisée comme aide uniquement et devra être mentionnée sous la forme d’un commentaire.

Chaque ligne de votre code doit pouvoir être expliquée par oral à vos camarades et enseignants.