**camera.py**

**Provenance :**

Ce programme est en partie tiré d’un tutoriel du vidéaste [Clear Code](https://www.youtube.com/@ClearCode), plusieurs fonctionnalités spécifiques nécessaires à notre jeu ont été rajoutées.

**Utilités :**

* Contrôler le fonctionnement de la caméra
* Afficher correctement tous les éléments de l’écran

**Fonctionnement :**

Ce programme consiste en une classe qui contrôle toutes les caractéristiques de la caméra. Lors de la création de la caméra, on crée la surface sur laquelle sera affiché l’ensemble des images. Puis on initialise les caractéristique du décalage de l’écran (expliqué plus bas) et du zoom.

La méthode « centre\_target\_camera » permet de cibler un sprite en définissant ses coordonnées comme le centre de l’écran (plus précisément, on calcule un vecteur décalage à appliquer aux coordonnées de toutes les images lors de leur affichage).

Une image contenant texte, capture d’écran, diagramme, ligne

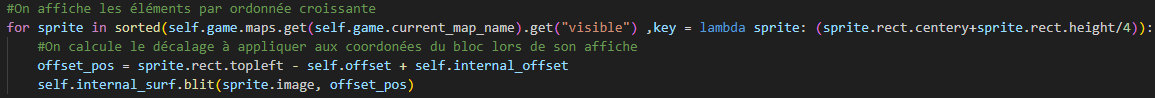
Description générée automatiquement

La méthode « custom\_draw » permet d’afficher tous les image de l’écran. D’abord, on centre l’écran sur le jouer. Ensuite, nous avons rajouté un système permettant d’optimiser les performances du jeu en récupérer uniquement les blocs qui peuvent apparaître à l’écran. Ces derniers sont les seuls à être affichés, tous les autres blocs sont ignorés et ne demandent aucune ressource de calcul. Cette sélection se base sur les coordonnées du joueur.

Une image contenant texte, capture d’écran, diagramme, ligne

Description générée automatiquement

Ensuite on affiche les blocs de la couche inférieur (le sol). Puis, on affiche toutes les autres images par ordonnée croissante.



Si le joueur est proche d’un bloc d’interaction, on affiche la touche E pour montrer qu’une interaction est possible.



Enfin, on agrandit la surface à afficher si le zoom est activé. (Pour optimiser les performances du jeu, nous créons une surface cible comportant les bonnes dimensions pour changer la taille de la surface affichée de façon plus rapide.)

Le programme se finit en affichant l’écran.