**EXPLICATIONS DU PROGRAMME**

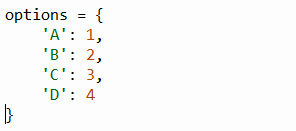
Tout d’abord, il faut importer les modules nécessaires pour l’application avec le mot clé « import ». Dans ce projet, on a besoin de deux modules : Le module CSV qui permet de lire et d'écrire des fichiers csv très facilement et le module random qui permet de générer de nombres aléatoires. Ce module nous aidera à choisir une question au hasard.



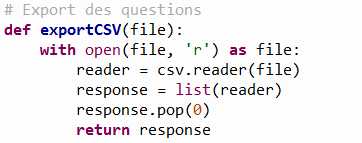
Puis, je déclare deux variables globaux à utiliser dans le code à venir.



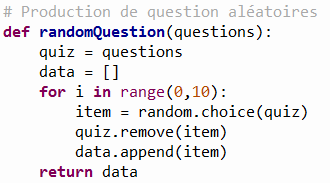
Ensuite, je déclare un dictionnaire qui stocke quatre clés correspondant à quatre valeurs pour quatre questions à choix multiples.



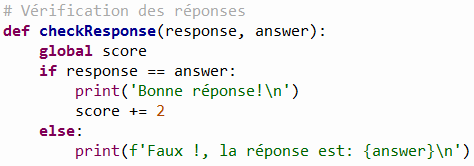
Pour récupérer toutes les questions et réponses dans le fichier CSV, j’ai créé une fonction « exportCSV ». Dans cette fonction, j’ouvre le fichier CSV en mode lecture et déclare une variable de lecteur CSV. J’utilise ici la fonction python pop avec comme paramètre 0 (index 0) pour ignorer la première ligne du fichier CSV parce que la première ligne représente les noms de colonne, et je n'en ai pas besoin dans ce programme.



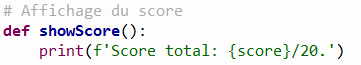
Pour avoir des questions aléatoires, j’ai créé la fonction « randomQuestion ». Cette fonction permet de générer dix questions aléatoires à partir de la liste obtenue par la fonction « exportCSV ». Une fois la question générée dans une nouvelle liste, cette dernière sera supprimée de la liste de questions afin qu’elle ne soit pas proposée une seconde fois.



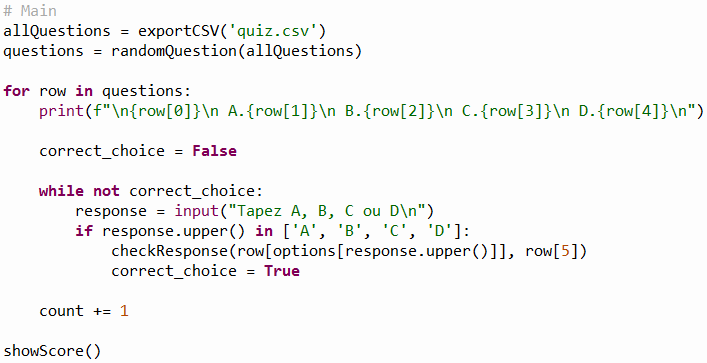
La tâche suivante consiste à vérifier si l'option est correcte ou non, choisie par l’utilisateur. La fonction « checkResponse » vérifiera si l’utilisateur choisit la bonne réponse ou non. Si la réponse est correcte, il affiche un message et met à jour la variable score en y ajoutant 2 points. Si la réponse est fausse, un autre message s'affichera avec la bonne réponse.



Le nom de la fonction showScore suggère d'afficher le score. Il affiche le score total sur vingt.



Enfin, Je déclare une boucle for pour lire les données de cette variable questions. Puis, j’affiche la question et les choix de réponses. Je déclare ensuite une variable nommée « **correct\_choice** », j’affecte de la valeur ' **False** '. Ici, je vérifierai si l’utilisateur a entré la bonne option ou non parmi « A », « B », « C » ou « D ». Si l'option n'est pas correcte, la boucle sera encerclée jusqu'à ce que l’utilisateur entre dans la bonne.



**PROGRAMME COMPLET**

La figure ci-dessous représente l'ensemble du code :

