

Groupe "Python 2" - Activité 01

Mentor : MILEGNE Dieu donné K. Jules

L'idée est de vous permettre d'utiliser vos connaissances en classe dans de vrais et réels projets.

NB : Utilisez les fonctions dans tous les exercices

Comment rendre les divers projets

- J'aurai préféré que chacun chronomètre mon travail, par exercice. Il faut que ce soit un challenge positif.
- Contenu du rendu :
 - Temps par exercice ;
 - Difficultés rencontrées ;
 - Captures du code et du résultat ou carrément envoyer vos codes sources portant votre nom

Exercice 1

Écrire un programme qui, à partir de la saisie de deux réels et un opérateur affiche le résultat après exécution de l'opération choisie.

Exemple :

- Si les entiers sont 14 et 2 et l'opérateur est "+" alors le résultat affiché est $14.00 + 2.00 = 16.00$
- Pour les mêmes entiers et si l'opérateur est "/" alors le résultat affiché est $14.00 / 2.00 = 7.00$

Exercice 2

Ecrire un programme en Python qui permet de calculer et d'afficher la distance entre deux points dont les coordonnées sont données.

Soient les points A (Xa, Ya) et B (Xb, Yb), la distance entre eux est donnée par la formule suivante :

$$d(A, B) = \sqrt{(Xa - Xb)^2 + (Ya - Yb)^2}$$

Mentor : MILEGNE Dieu donné K. Jules

Exercice 3 - Première partie du projet "Jeu de devinette"

But : Le but du jeu est d'inviter l'utilisateur à deviner un nombre que l'ordinateur aurait "deviné" (en réalité, il s'agit d'un nombre aléatoire).

Dans un premier temps, le nombre est choisi aléatoirement (il serait intéressant de lire en entrée la borne maximale du nombre aléatoire).

Ensuite, l'utilisateur est amené à le deviner ; il sera aidé au cours de sa quête par des phrases du genre 'Plus petit' ou 'Plus grand' suivant que le nombre à deviner est respectivement soit plus petit, soit plus grand que le nombre entré ...

Implémentez cette partie solution

Exercice 4 - Deuxième partie du projet "Jeu de devinette" : Gestion du niveau de difficulté

Pour rendre le jeu plus geek, on permet à l'utilisateur de choisir d'abord son niveau entre :

1. Facile (0 -100) ;
2. Moyen (0 – 500) ;
3. Difficile (0 -1000).

Ainsi, la limite de la génération du nombre aléatoire est respectivement 100, 500 et 1000.

Implémentez cette deuxième partie.

Exercice 5 - Dernière partie du projet "Jeu de devinette" : Introduction de la notion de score

Pour que le joueur ne joue pas indéfiniment sur la même partie, on précise qu'il dispose d'un nombre fixe de tentatives (Vous pouvez choisir mais on va prendre 15 pour harmoniser). Donc chaque utilisateur dispose seulement de 15 essais. Que ce soit à l'issue de ce nombre de tentatives ou que l'utilisateur devine le nombre caché : un message de félicitations est alors affiché.

Après que l'utilisateur ait deviné le nombre caché ou que ces tentatives aient atteint le nombre maximum de tentatives, il est invité à taper n ou N s'il désire arrêter le jeu, ou alors une autre lettre pour reprendre le jeu.

Message de félicitations :

- [1 – 3] tentatives => Super Voyant
- [4 – 6] tentatives => Sage Voyant

Mentor : MILEGNE Dieu donné K. Jules

[7 – 9] tentatives => Apprenti Voyant

[10 et plus[tentatives => Pusillanime

NB : Le programme doit mémoriser les tentatives de tout joueur de manière à afficher à la fin d'un jeu :

- L'historique des tentatives,
- Leur écart-type,
- Leur moyenne olympique.

Conseils :

- Utilisation des tableaux ou des fichiers
- Tableaux dans ce cas serait plus facile je trouve, mais faites ce qui vous semble alaise
- Vous pouvez faire les deux même sans souci
- Ou alors utiliser une base de données carrément 🤖. L'essentiel est de trouver le résultat

Henri Ford a dit un jour : *“L'échec n'est qu'une excellente occasion de recommencer de manière plus intelligente.”*

Rick WARREN a dit : *“Si vous pratiquez régulièrement quelque chose, vous finirez par devenir très bon. L'entraînement mène à l'excellence.”*

A vos claviers, let's code!

Mentor : MILEGNE Dieu donné K. Jules