

EXERCICES DE CODAGE

BOUCLES WHILE

Rappelez : on écrit une boucle while comme ceci : while condition: La condition peut être n'importe quoi qui s'évalue à Vrai ou Faux. Les boucles while utilisent souvent un compteur, comme ceci:

```
x = 0
while x<3:
    x = x+1
```

Vous pouvez terminer une boucle avec une instruction break.

Exercice 1 : Créez un compte à rebours simple qui affiche les nombres 10 à 0 en utilisant une boucle while.

Exercice 2 : Rendons le compte à rebours plus réel en ajoutant une pause d'une seconde entre l'affichage de chacun des nombres ! Interprétez ce qui suit avant la première ligne de votre code : import time

Puis trouvez l'endroit où vous voulez que l'exécution du programme attende une seconde et entrez le code suivant : time.sleep(1)

(1 est le nombre de secondes à attendre - vous pouvez expérimenter avec cette valeur.)

Exercice 3 : Créons des tables de multiplication : Demandez à l'utilisateur de choisir un nombre entre 1 et 9 et affichez la table de multiplication à l'aide d'une boucle. Exemple : Le nombre choisi est 3, donc affichez le résultat de 1x3, 2x3, 3x3, 4x3, 5x3, 6x3, 7x3, 8x3, 9x3, 10x3. L'idéal est d'afficher chaque ligne comme suit : 1x3 = 3

LISTS

Rappelez : vous définissez une liste comme ceci : myList = [1,2,3,4,5,6,7]

Vous pouvez modifier les éléments d'une liste comme ceci : myList[0] = 8 (Ceci change l'élément à la position 0 en 8)

Vous pouvez ajouter un nouvel élément à la fin comme ceci : myList.add(21)

Vous pouvez supprimer un élément comme ceci : myList.pop(6) ou : myList.remove(7) (les deux supprimeraient le 7 à la fin ; pop trouve la position indiquée, remove trouve la valeur)

Vous pouvez trouver la longueur d'une liste comme ceci : myList.len()

Exercice 4 : Allons faire du shopping ! Créez une liste vide pour contenir les prix. Créez une boucle dans laquelle vous laissez les utilisateurs saisir les prix des produits jusqu'à ce qu'ils vous fassent savoir qu'ils ont terminé. Enregistrez tous les prix dans une liste. Lorsque l'utilisateur a terminé, affichez la liste et le prix total.

Exercice 5 : Créez 2 listes de noms. Créez un programme qui prend chaque élément de la première liste et l'ajoute à la fin de la deuxième liste. Cependant, nous ne voulons pas avoir un nom deux fois, donc vérifiez si le nom existe avant de l'ajouter. Vous pouvez le faire avec l'instruction if suivante : if myName in myList: Remplacez myName

par la variable qui contient le nom que vous recherchez. Remplacez maListe par la liste que vous recherchez.

Exercice 6 : Créez une liste d'ingrédients de cuisine. Créez un programme qui permette à l'utilisateur de vérifier si un ingrédient est disponible ou non (en affichant « ingrédient en stock » ou « ingrédient non disponible »).