Listes de Listes , Tableaux

INTRODUCTION AUX LISTES IMBRIQUES :

On part d'une liste contenant des listes imbriquées :

```
L=[[1,2,3],[4,5,6],[7,8,9]]
```

Attention , si on veut obtenir un index de la liste, on obtiendra une sous liste et non un item de la sous liste :

```
>>> L1=L[0]

>>> L1
[1, 2, 3]
```

Si on veut un item de la sous liste, il faut rajouter un indice :

```
>>> L2=L[1][2]

>>> L2
6
```

TABLEAU A DEUX DIMENSIONS :

On a ici une fonction qui lance un tableau en deux dimensions , ou un array. Pour cela , on a besoin de la liste créée ci-dessus(On peut également mettre la liste en argument de la fonction):

```
L=[[1,2,3],[4,5,6],[7,8,9]]
```

```
def Array():
    for SousListe in range(len(L)):
        for ItemSousListe in range (len(L[SousListe])):
            ItemSousListe=L[SousListe][ItemSousListe]
            print(ItemSousListe, end=" ")
        print()
```

Explication du code:

→ On commence par faire une boucle pour chaque Sous Liste dans la liste dans un range du nombre de sous listes. Ici , on retrouve range = 3

```
for SousListe in range(len(L)):
```

→ Puis, on fait une boucle dans la boucle pour chaque Sous Item de chaque Sous Liste. Pour cela , le range sera égal à la longueur de la sous liste. Comme on a un tableau de 3 éléments a chaque fois, le nombre de sous item sera de 3 encore une fois :

```
for ItemSousListe in range (len(L[SousListe])):
```

→Enfin, on indente dans la deuxième boucle la valeur de chaque Sous Item, que l'on va print ensuite. La fonction print possède une méthode nommée 'end'. Elle nous permet d'afficher quelque chose après le print de l'élément. Si on avait pas la méthode 'end', alors les numéros seraient print l'un après l'autre sur une ligne différente. Ici, le 'end' contiendra une valeur d'un espace pour avoir de la visibilité dans le tableau :

```
ItemSousListe=L[SousListe][ItemSousListe]
print(ItemSousListe, end=" ")
```

→Pour finir, on indente un print vide à la fin de la première boucle pour que ça fasse un retour à la ligne.

Si on met tout ça ensemble, le code ressemble à ça :

```
>>> Array()
1 2 3
4 5 6
7 8 9
```

TROUVER UNE LIGNE ET UNE COLONNE DU TABLEAU:

```
def Ligne():
    for SousItem in range(len(L)):
        ElementLigne=L[0][SousItem]
        print(ElementLigne, end=" ")
```

Pour chaque sous item de sous liste:

On prend tout les items de la sous liste

On print ensuite les items suivis d'un espace entre eux.

```
def Colonne():
    for SousListe in range(len(L)):
        ElementColonne= L[SousListe][2]
        print(ElementColonne)
    print()
```

Pour chaque sous liste:

On prend le troisième élément de chaque sous liste

On affiche l'item suivis d'un retour à la ligne automatique

Optionnel

Affichages:

→

```
>>> Ligne()
1 2 3
```

→

```
>>> Colonne()
3
6
9
```