

Chapitre 7 : Tableaux

On utilise les tableaux pour traiter une grande quantité de données, de même type ou pas.

Syntaxe

- Pour déclarer un tableau :

```
In [ ]: tab = ["Sophie", 30.2, False]
```

- On accède à l'élément `t[i]` par son *indice* `i`, qui va de 0 à `len(t)-1` où `len(t)` est la taille du tableau.
- `t[i] = val` modifie l'élément d'indice `i`.
- `t.append(val)` ajoute l'élément `val` à la fin du tableau `t` déjà créé.
- `t2 = t` crée un *alias* de `t`, c'est-à-dire que `t2` et `t` désignent le même objet jusqu'à ce que l'un d'eux soit réaffecté.

Parcours d'un tableau

On parcourt un tableau avec une boucle pour le créer ou agir sur les éléments.

Exemple :

```
In [ ]: nombres = []
        i = 0
        while i < 10:
            nombres.append(i)
            i+=1
        print(nombres)
```

Les chaînes de caractères ressemblent aux tableaux. On peut accéder en lecture aux caractères d'une chaîne `s` par leur indice (`s[i]`), mais on ne peut pas les modifier.

Mode	Description	Remarques
r	Read - Valeur par défaut. Ouvre le fichier en lecture et renvoie une erreur si le fichier n'existe pas.	
w	Write - Ouvre le fichier en écriture et crée le fichier si il ne le trouve pas.	Attention Si le fichier préexiste, le contenu initial du fichier est écrasé par le nouveau contenu.
a	Append - Ouvre le fichier en écriture à la fin et crée le fichier si il ne le trouve pas.	Attention Si le fichier préexiste, le contenu initial du fichier est préservé, le nouveau contenu est écrit à la suite du contenu préexistant.
x	Create - Crée le fichier ou retourne une erreur si il existe déjà.	

fonction de parcours	Utilisation
f.open(path, mode)	retourne un flux sur le fichier path
f.readline()	retourne la ligne suivante du fichier
f.write(chaine)	écrit dans le fichier la chaine
f.close()	ferme le fichier

fonction utiles lors du parsing	Utilisation
chaine.strip()	suprime les caractères espace en début et fin de ligne (aussi le caractère '\n')
chaine.split(separateur)	sépare la chaine là où il y a un caractère séparateur et retourne le tableau des chaines séparées.

Pour parcourir un fichier jusqu'à sa dernière ligne :

```
In [ ]: f = open( path, mode )
        li = f.readline()
        while( '' != li ) :
            operations ...
            li = f.readline()
        f.close()
```