➢ Indiçage

len() : Permet de connaître la longueur d'une liste ou d'un tulpe.
Exemple : *len*(("a", "b", "c")) et *len*(["a", "b", "c"]) renvoient la valeur 3
...[n] : Permet de récupérer directement le $(n + 1)^{ème}$ terme.
Exemple : ("a", "b", "c")[2] et ["a", "b", "c"][2] renvoient "c"
...[n][m] : Permet de récupérer directement le $(m + 1)^{ème}$ terme de la $(n + 1)^{ème}$ liste.
Exemple : [[1, 2, 3], [4, 5, 6], [7, 8, 9]] [1] [2] renvoie 6

Attention : Le premier terme a pour rang 0.
Remarque : Le dernier terme a pour rang $n - 1$ ou plus simplement -1.

➢ Opérations

concaténation : Le symbole + permet de mettre bout à bout deux listes ou tuples.
Exemple : ["a", "b", "c"] + ["d", "e"] renvoie la liste ["a", "b", "c", "d", "e"]
répétition : Le symbole * permet de répéter plusieurs fois une même liste ou un même tuple.
Exemple : (0, 1)*3 renvoie le tuple (0, 1, 0, 1, 0, 1)
appartenance : L'instruction *in* permet de tester si un caractère ou une chaîne de caractère sont présents dans une liste ou un tuple.
parcours : *in* permet aussi de parcourir une liste ou un tuple lorsqu'on l'associe à un *for*.
Exemple : *for* c *in* ["a", "b", "c"] :
suppression : *del* permet de supprimer un élément d'une liste à partir de son indice.
Exemple : l1 = ["a", "b", "c", "d", "e"]
del l1 [2] modifie la liste l1 en ["a", "b", "d", "e"]

Remarque : Les tuples, contrairement aux listes, ne sont pas modifiables.

➢ Outils et méthodes

Les listes sont des objets, c'est-à-dire que l'on peut leur appliquer une méthode :

liste.*count*(mot) Permet de compter combien de fois apparaît un mot dans une liste.
liste.*index*(mot) Permet de trouver le premier indice où apparaît un mot dans une liste.

liste.*reverse*() Permet d'inverser l'ordre d'une liste.
liste.*sort*() Permet de trier une liste.
liste.*append*(mot) Permet d'ajouter un mot en fin de liste.
liste.*remove*(mot) Permet de supprimer la première occurrence d'un mot dans une liste.

Attention : Ces méthodes modifient l'objet sur lequel elles agissent ! Les tuples n'étant pas modifiables, on ne peut pas leur appliquer ces méthodes

➢ Conversions

chaine.*split*(" ") Permet de convertir une chaîne de caractères en une liste en coupant à espace.
" ".*join*(liste) Permet de convertir une liste en un chaîne de caractère en intercalant un espace entre chaque mot de la liste.

Remarque : On peut bien entendu remplacer " " par n'importe quel mot.