

## Principales méthodes (modifient la liste)

Méthode	Effet
<code>append(x)</code>	ajoute <code>x</code> à la fin
<code>insert(i,x)</code>	insère <code>x</code> à l'indice <code>i</code>
<code>remove(x)</code>	supprime la 1 <sup>ère</sup> occurrence de <code>x</code>
<code>pop()</code>	supprime et renvoie le dernier élément de la liste
<code>pop(i)</code>	supprime et renvoie l'élément d'indice <code>i</code>
<code>clear()</code>	vide complètement la liste
<code>reverse()</code>	inverse l'ordre
<code>sort()</code>	trie par ordre croissant

### Exercice 1

- Créer une liste `fruits` qui ne contient que les éléments pomme et poire.
- La liste `fruits` doit contenir maintenant dans l'ordre : pomme, banane, poire, orange, cerise.

```
1 fruits = ["pomme", "banane", "poire", "orange", "cerise"]
```

- Que vaut `len(fruits)` après la question précédente ?
- Écrire l'instruction qui affiche le 2<sup>e</sup> élément de la liste `fruits`.
- On tape :

```
1 animaux = ["chat", "chien", "lapin"]
2 animaux[1] = "tigre"
3 print(animaux)
```

Que voit-on à dans le shell de Python ?

- On veut ajouter « poisson » à la fin de la liste `animaux`. Écrire l'instruction.
- Que se passe-t-il si on écrit les instructions suivantes ?

```
1 animaux[4] = "lapin"
2 print(animaux)
```

### Exercice 2

On a la liste : `nombre = [5, 12, 8, 19, 3]`

Écrire une boucle qui affiche tous les éléments de la liste `nombre`.

### Exercice 3

Voici les notes d'une classe.

```
1 notes = [4, 3.5, 5.5, 4.5, 3.5, 5, 5, 4, 5, 3, 4.5]
```

Écrire les instructions :

- qui affichent le nombre d'éléments de la liste `notes` ;
- qui calculent la somme des notes ;
- qui calculent la moyenne de la liste ;

- d) qui affichent le nombre d'élèves qui ont 4 ou plus ;
- e) qui affichent la note maximale, puis la note minimale.

#### Exercice 4

On exécute dans l'ordre :

```

1 legumes = ["carotte", "salade", "tomate", "poivron"]
2 print(len(legumes))
3 legumes.append("concombre")
4 legumes.insert(1, "oignon")
5 legumes.remove("tomate")
6 print(legumes)
7 print(len(legumes))
8 legumes = []
9 print(legumes)

```

Que peut-on lire dans le shell de Python ?

#### Exercice 5

Voici une liste grille représentant un morpion :

```

1 grille = [["X", "0", " "],[" ", "X", " "],["0", " ", "X"]]

```

- a) Que contient `grille[0][2]` ?
- b) Que contient `grille[1][2]` ?
- c) Que contient `grille[2][1]` ?
- d) Écrire l'instruction pour mettre un « 0 » dans la case centrale.
- e) Dessiner à quoi ressemble la grille après cette modification.


- f) Écrire les instructions pour afficher la deuxième ligne du morpion.

#### Exercice 6

On donne la liste suivante.

```

1 nombres = [12, 8, 15, 7, 9, 18, 6]

```

- a) Créer une nouvelle liste `carres` contenant les carrés des éléments de `nombres`.
- b) Afficher les indices des éléments pairs de la liste `nombres`.

#### Exercice 7

Écrire un programme avec l'instruction `while` qui demande à l'utilisateur de saisir des nombres jusqu'à ce qu'il tape `-1`.

Tous les nombres saisis (sauf `-1`) sont stockés dans une liste, puis on affiche la liste et sa moyenne.