

## Les listes en Python (suite)

### **Trouver un item (un élément) dans une liste**

Pour savoir si un élément est dans une liste, on utilise le mot clé «in» de cette manière:

```
liste = [1,2,3,5,10]
```

```
print(3 in liste)
```

```
True
```

```
Print(11 in liste)
```

```
False
```

1

## Les listes en Python (suite)

### ***Supprimer un élément avec sa position***

La méthode pop() permet de supprimer un élément identifié par sa position dans la liste (par défaut le dernier élément)

```
ma_liste = [1,2,3, 'A', 'B','C','D']
```

```
print ("suppression du dernier élément : ", ma_liste.pop())
```

```
print ("suppression du 2ème élément : ", ma_liste.pop(1))
```

```
print(ma_liste)
```

2

## Les listes en Python (suite)

*Boucler sur une liste (afficher les éléments d'une liste) :*

*La syntaxe d'une telle boucle est la suivante :*

```
liste = [10, "Table", 3.14, "Bonjour", "C'est la fin"]
```

*for element in liste:*

```
    print(element)
```

*On peut aussi récupérer l'index avec l'élément affiché, à l'aide de la fonction enumerate .*

*for lettre in enumerate(liste):*

```
    print lettre
```

*Résultat d'affichage :*

```
(0, 'a')
```

```
(1, 'd')
```

```
(2, 'm')
```

*Remarque : les valeurs retournées par la boucle sont des tuples.*

3

## Les listes en Python (suite)

### **Copier une liste**

Une erreur à éviter! : copier une liste avec la manière suivante :

```
x = [1,2,3]
```

```
y = x
```

Avec cette manière, on fait que créer deux noms différents à la même liste. Ainsi, si on change une valeur de la liste **y**, la liste **x** sera elle aussi affectée par cette modification :

```
y[0] = 4
```

```
print(x)
```

```
[4, 2, 3]
```

Alors, pour copier une liste dans une autre variable, on procède ainsi :

```
x = [1,2,3]
```

```
y = x[:]
```

```
y[0] = 9
```

```
print(y) # Résultat : [9, 2, 3], la liste y est modifiée
```

```
print(x) # Résultat : [1, 2, 3], la liste x reste intacte
```

4

## Les listes en Python (suite)

### Étendre une liste par une autre liste

Il est possible d'étendre une liste avec une autre liste à l'aide de la méthode extend

```
x = [1, 2, 3, 4]
```

```
y = [4, 5, 1, 0]
```

```
x.extend(y)
```

```
print(x)
```

```
[1, 2, 3, 4, 4, 5, 1, 0]
```

5

## Les listes en Python (suite)

### Exercice (Palindrome à l'aide des listes)

Réalisez une fonction qui accepte en paramètre une chaîne de caractères,

Puis transforme cette chaîne dans une variable locale de type liste

Utilisez une autre variable locale de type liste pour recopier la liste de la chaîne

Inversez cette deuxième liste

Comparez les deux listes pour conclure s'il s'agit d'un palindrome !

6