

Cours 3 - Les listes en python

Introduction

En python, une liste est une structure de données ordonnée et modifiable . Elle regroupe des éléments qui peuvent être de types différents. Les éléments d'une liste sont écrits entre crochets et sont identifiés par des indices.

1. Création d'une liste en Python

Exemple

```
#Création d'une liste
```

```
myList = ["Python", "Java", 34]
```

```
print ('La liste non vide myList est : ',myList)
```

```
#Création d'une liste vide
```

```
myList2 = [ ]
```

```
print ('La liste vide myList2 est: ', myList)
```

```
#Création d'une liste contenant une liste
```

```
myList3 = [ [12, 'hello', 3.5], "bonjour", 54 ]
```

```
print ('La liste myList3 contenant une liste est: ', myList3)
```

2. Accès aux éléments d'une liste

Pour accéder à un élément d'une liste, on spécifie son indice (numéro d'index). Par exemple, myList[i] désigne l'élément à l'indice i de la liste appelée myList.

2-1. Indexation positive

"Python"	"Java"	34
0	1	2

Exemple

Afficher les éléments de la liste myList:

```
print( myList[0] )
```

```
print( myList[1] )
```

```
print( myList[2] )
```

Afficher les éléments de la liste myList3

```
print( myList3[0] )
```

```
print( myList3[1] )
```

```
print( myList3[2] )
```

2-2. Indexation négative

"Python"	"Java"	34
-3	-2	-1

L'indexation négative signifie à partir de la fin, -1 se réfère au dernier élément, -2 se réfère à l'avant-dernier élément, etc.

L'exemple suivant affiche le dernier élément de la liste myList:

```
myList = ["Python", "Java", 34]
```

```
print(myList[-1])
```

```
print(myList[-2])
```

```
print(myList[-3])
```

2-3. Tranchage (slicing en Anglais)

Pour extraire une liste d'éléments, on donne une plage d'index en spécifiant par où commencer et où terminer la plage.

Syntaxe

```
myList[m:n ]
```

donne une contenant les éléments de myList qui sont entre les indices m et n-1

L'exemple suivant affiche les éléments entre les indices 2 et 3

```
myList = ["Blue", "Red", "Green", "Orange", "Black", "Yellow"]  
print(myList[2:4])
```

Si on omet l'indice de départ, la plage commencera au début de la liste

```
myList[ : n ]
```

donne une contenant les éléments de myList qui sont entre les indices 0 et n-1

L'exemple suivant affiche les éléments entre les indices 0 et 2

```
myList = ["Blue", "Red", "Green", "Orange", "Black", "Yellow"]  
print(myList[:3])
```

En ignorant la valeur de fin, la plage ira à la fin de la liste

```
myList[ m : ]
```

donne les éléments de myList qui sont entre l'indice m et la fin

L'exemple suivant affiche les éléments entre l'indice 2 et la fin

```
myList = ["Blue", "Red", "Green", "Orange", "Black", "Yellow"]  
print( myList[2 : ] )
```

2-4. Plage d'index négative

Spécifiez des index négatifs si vous souhaitez commencer la recherche à la fin de la liste. L'exemple suivant renvoie les éléments de l'index -3 (inclus) à l'index -1 (exclus):

```
myList = ["Blue", "Red", "Green", "Orange", "Black", "Yellow"]  
print( myList[-3 : -1] )
```

3. Changer la valeur d'un élément de la liste

Contrairement à une chaîne de caractères, on peut modifier les éléments d'une liste en python.

Exemple

```
myList = ["Python", "Java", 34]
print('la liste myList contient:')
print(myList)

# Changer l'élément à l'indice 0 de myList:
myList[0]=7.5

# Changer l'élément à l'indice 1 de myList:
myList[1]="Oracle"

# Changer l'élément à l'indice 2 de myList:
myList[2]="C"

print('la liste myList contient après modification:')
print(myList)
```

4. Parcourir les éléments d'une liste Python

Vous pouvez parcourir les éléments d'une liste en Python, en utilisant une boucle for:

Exemple

```
#Afficher tous les éléments de la liste, un par un:
for v in myList:
    print(v)
```

5. Quelques méthodes ou fonctions de manipulation d'une liste Python

Il existe des méthodes ou des fonctions qu'on peut utiliser pour faire des manipulations sur les listes.

5-1. Longueur d'une liste Python

La fonction len donne la longueur d'une liste donnée en paramètre

Exemple

```
#Afficher le nombre d'éléments de la liste:
myList = ["Python", "Java", 34]
print ("La longueur de ma liste est" , len (myList))
```

5-2. Ajout d'un élément à la fin d'une liste

la méthode append() ajoute à la fin d'une liste un élément donné en paramètre.

Exemple

```
#ajouter un élément à la liste avec la méthode append() :
myList = ["Python", "Java", 34]
myList.append ("Oracle")
print (myList)    #Affiche : ["Python", "Java", 34, "Oracle"]
```

5-3. Ajout d'un élément à un indice spécifié d'une liste

La méthode insert() ajoute un élément à une liste à l'indice spécifié en paramètre.

Exemple

```
#Insérer un élément en deuxième position:
myList = ["Python", "Java", 34]
myList.insert (1, "C++")
print (myList) # Affiche : ["Python", "C++" "Java", 34]
```

5-4. Suppression d'un élément d'une liste Python

Il existe plusieurs méthodes pour supprimer des éléments d'une liste:

- La méthode `remove()` supprime un élément donné en paramètre.
- La méthode `pop()` supprime un élément dont l'indice est donné en paramètre (ou le dernier élément si aucun indice n'est donné en paramètre)
- Le mot clé `del` supprime l'élément de la liste à l'index spécifié(supprime la liste si aucun indice n'est spécifié)
- La méthode `clear()` vide la liste en supprimant tous les éléments

Exemple

```
#suppression d'un élément spécifié avec la méthode remove()  
myList = ["Python", "Java", "PHP"]  
myList.remove("Java")  
print(myList)
```

Exemple

```
# Suppression d'un élément d'indice spécifié avec la méthode pop()  
myList = ["Python", "Java", "PHP"]  
myList.pop( 0 )  
print(myList)
```

Exemple

```
# Suppression d'élément à un indice spécifié avec le mot clé del  
myList = ["Python", "Java", "PHP"]  
del myList[0]  
print(myList)
```

Exemple

```
# Suppression d'une liste avec le mot clé del  
myList = ["Python", "Java", "PHP"]  
del myList  
print(myList) #cela causera une erreur car "myList" n'existe plus.
```

Exemple

```
# Suppression d'une liste avec la méthode clear  
myList = ["Python", "Java", "PHP"]  
myList.clear ()  
print (myList) #cela causera une erreur car "myList" n'existe plus.
```

Exercices sur les listes

Exercice 1

Remplir une liste de longueur 10 avec des éléments donnés par l'utilisateur.
Afficher la liste.

Exercice 2

Remplir une liste de longueur 3 dont les éléments sont des listes de 3 entiers donnés par l'utilisateur.

Par exemple [[1 , 2 , 3] , [4 , 5 , 6] , [7 , 8 , 9]]

Afficher la liste.

Remarque

Une telle liste est représentée une matrice carrée dont les lignes sont les éléments de la liste.

Exercice 3

Remplir deux matrices carrées a et b de dimensions (3,3) avec des entiers donnés par l'utilisateur.

Afficher les matrices a et b

Calculer et afficher la matrice s somme de a et b

Calculer et afficher la matrice p produit de a et b