

## Python: Les listes

Les listes sont l'une des structures de données les plus utilisées en Python. Elles permettent de stocker une collection ordonnée d'éléments, qu'ils soient de même type ou non. Dans cet article, nous explorerons en détail ce qu'est une liste en Python, comment les créer, les manipuler et les utiliser dans vos programmes, avec des exemples concrets à chaque étape.

5GMS

3TTR

 Exploration

## Qu'est-ce qu'une liste en Python ?

Une liste en Python est **une collection ordonnée d'éléments**, où chaque élément est **accessible par un index**. Les listes peuvent contenir des éléments de différents types : entiers, flottants, chaînes de caractères, voire d'autres listes ou objets plus complexes.

## Création d'une liste

Pour créer une liste en Python, vous pouvez utiliser des **crochets** `[]` et séparer les éléments par des **virgules**. Voici un exemple :

```
1 | ma_liste = [1, 2, 3, 4, 5]
```

Vous pouvez également y **ajouter des éléments** ultérieurement :

```
1 | liste_vide = []  
2 | liste_vide.append(1)  
3 | liste_vide.append(2)  
4 | liste_vide.append(3)
```

## Accéder aux éléments d'une liste

Les éléments d'une liste sont accessibles par leur index, en commençant par 0 pour le premier élément. Voici comment accéder aux éléments d'une liste :

```
1 ma_liste = [10, 20, 30, 40, 50]
2 print ma_liste[0] # Affiche le premier élément, soit 10
3 print ma_liste[2] # Affiche le troisième élément, soit 30
```

Vous pouvez également utiliser des indices négatifs pour accéder aux éléments en partant de la fin de la liste :

```
1 print ma_liste[-1] # Affiche le dernier élément, soit 50
2 print ma_liste[-2] # Affiche l'avant-dernier élément, soit 40
```

## Recherche dans une liste

La recherche dans une liste consiste à vérifier si un élément est présent dans une liste donnée. Cela se fait en utilisant l'opérateur `in`. Voici un exemple :

```
1 liste = [1, 2, 3, 4, 5]
2 print(3 in liste) # Affiche True
3 print(6 in liste) # Affiche False
```

## Modifier des éléments d'une liste

Les éléments d'une liste en Python sont **mutables**, ce qui signifie qu'ils **peuvent être modifiés** après leur création. Pour modifier un élément d'une liste, vous pouvez simplement accéder à cet élément par son index et lui assigner une nouvelle valeur :

```
1 ma_liste = [1, 2, 3, 4, 5]
2 ma_liste[0] = 10
3 print ma_liste # Affiche [10, 2, 3, 4, 5]
```

## Supprimer des éléments d'une liste

Il existe plusieurs façons de supprimer des éléments d'une liste en Python. Vous pouvez utiliser la méthode `remove()` pour supprimer un élément spécifique sur base de sa **valeur**:

```
1 ma_liste = [1, 2, 3, 4, 5]
2 ma_liste.remove(3)
3 print ma_liste # Affiche [1, 2, 4, 5]
```

Vous pouvez également utiliser le mot-clé `del` suivi de l'**index** de l'élément à supprimer :

```
1 | del ma_liste[2]
2 | print(ma_liste) # Affiche [1, 2, 5]
```

## Ajouter des éléments à une liste

Pour ajouter un élément à une liste, vous pouvez utiliser la méthode `append()` pour ajouter l'élément à la fin de la liste :

```
1 | ma_liste = [1, 2, 3]
2 | ma_liste.append(4)
3 | print(ma_liste) # Affiche [1, 2, 3, 4]
```

## Exemples avec des chaînes de caractères

Les listes peuvent également contenir des chaînes de caractères. Voici un exemple :

```
1 | liste_mots = ["Bonjour", "à", "tous"]
2 | print(liste_mots[0]) # Affiche "Bonjour"
3 | print(liste_mots[-1]) # Affiche "tous"
```

Vous pouvez également concaténer des listes de chaînes de caractères à l'aide de l'opérateur `+=` :

```
1 | liste_mots = ["Bonjour", "à"]
2 | liste_mots += ["tous"]
3 |
4 | print(liste_mots) # Affiche ["Bonjour", "à", "tous"]
```

## Conclusion

Les listes en Python sont des structures de données polyvalentes et puissantes qui permettent de stocker et de manipuler des collections d'éléments de manière efficace. Dans cet article, nous avons couvert les bases de la création, de l'accès, de la modification et de la suppression des éléments d'une liste, ainsi que l'ajout de nouveaux éléments. En comprenant ces concepts fondamentaux, vous serez en mesure d'utiliser les listes de manière efficace dans vos programmes Python.

Vidéo à la une à regarder sur Youtube:  [http://youtu.be/Vqtmbon5\\_Nk](http://youtu.be/Vqtmbon5_Nk)