



Une liste est une collection d'éléments dans un ordre bien défini.

Exemples:

- Liste des lettres de l'alphabet
- Liste des chiffres
- Liste des étudiants
- ...

```
lettres = ['a', 'b', 'c', 'd', 'e']
```

```
lettres
```

```
['a', 'b', 'c', 'd', 'e']
```

▼ Accéder aux éléments de la liste

Les listes sont des collections ordonnées. Les éléments peuvent être accédés en spécifiant leur position dans la liste (à partir de 0) entre crochets "[]".

```
lettres[2]
```

```
'c'
```

On peut utiliser des index négatifs pour accéder aux éléments depuis la fin de la liste

```
lettres[-1]
```

```
'e'
```

▼ Changer, Ajouter et Supprimer des éléments de la liste

Modifier les éléments d'une liste

La syntaxe pour la modification d'un élément est similaire à celle pour l'accès aux éléments de la liste. Pour changer la valeur d'un élément, il suffit d'utiliser le nom de la liste suivi de l'index de l'élément en question et fournir la nouvelle valeur.

```
cars = ['Audi', 'BMW', 'Mercedes', 'Honda', 'Toyota', 'Suzuki', 'Renault', 'Peugot']
```

```
cars[0] = 'Volkswagen'
```

```
cars
```

```
['Volkswagen',  
 'BMW',  
 'Mercedes',  
 'Honda',  
 'Toyota',  
 'Suzuki',  
 'Renault',  
 'Peugot']
```

Ajout d'éléments à la liste

La manière la plus simple pour ajouter des éléments à la liste est à travers la fonction **append**. L'élément est ajouté à la fin de la liste.

```
cars.append('Bugatti')
```

```
cars
```

```
['Volkswagen',  
 'BMW',  
 'Mercedes',  
 'Honda',  
 'Toyota',  
 'Suzuki',  
 'Renault',  
 'Peugot',  
 'Bugatti']
```

La fonction **append** permet de construire une liste de manière dynamique en partant d'une liste vide.

```
months = []  
months.append('Janvier')  
months.append('Février')  
months.append('Mars')  
months.append('Avril')  
months.append('Mai')
```

```
months.append('Juin')
months.append('Juillet')
months.append('Août')
months.append('Septembre')
months.append('Octobre')
months.append('Novembre')
months.append('Décembre')
```

months

```
['Janvier',
 'Février',
 'Mars',
 'Avril',
 'Mai',
 'Juin',
 'Juillet',
 'Août',
 'Septembre',
 'Octobre',
 'Novembre',
 'Décembre']
```

Insertion des éléments dans une liste On peut ajouter un nouveau élément à une liste dans n'importe quelle position en utilisant la fonction **insert** en spécifiant l'index de la position et la valeur du nouvel élément.

```
cars.insert(0, 'Audi')
```

cars

```
['Audi',
 'Volkswagen',
 'BMW',
 'Mercedes',
 'Honda',
 'Toyota',
 'Suzuki',
 'Renault',
 'Peugot',
 'Bugatti']
```

Enlever des éléments de la liste On a souvent besoin de retirer des éléments de la liste, on peut supprimer un élément à partir de sa position ou sa valeur.

Un élément peut être supprimé en utilisant l'instruction **del**. La valeur supprimée n'est plus accessible après l'exécution de l'instruction.

```
del cars[2]
```

cars

```
['Audi',  
 'Volkswagen',  
 'Mercedes',  
 'Honda',  
 'Toyota',  
 'Suzuki',  
 'Renault',  
 'Peugot',  
 'Bugatti']
```

La méthode **pop** peut être utilisée pour supprimer un élément de la liste. Elle permet aussi de retourner la valeur de l'élément supprimé

```
car = cars.pop()
```

```
car
```

```
'Bugatti'
```

```
print(cars)
```

```
['Audi', 'Volkswagen', 'Mercedes', 'Honda', 'Toyota', 'Suzuki', 'Renault', 'Peugot']
```

La méthode **pop** peut être utilisée pour supprimer les éléments de n'importe quelle position dans la liste en incluant l'index de l'élément en question.

```
car = cars.pop(3)
```

```
car
```

```
'Honda'
```

```
cars
```

```
['Audi', 'Volkswagen', 'Mercedes', 'Toyota', 'Suzuki', 'Renault', 'Peugot']
```

Supprimer les éléments par valeur Un élément peut être supprimé à travers sa valeur en utilisant la méthode **remove**.

```
cars.remove('Toyota')
```

```
cars
```

```
['Audi', 'Volkswagen', 'Mercedes', 'Suzuki', 'Renault', 'Peugot']
```

▾ Organisation d'une liste

Python fournit un nombre de méthodes permettant d'organiser les listes.

Trier une liste d'une manière permanente

La méthode **sort** permet de trier une liste.

```
cars.sort()
```

```
cars
```

```
['Audi', 'Mercedes', 'Peugot', 'Renault', 'Suzuki', 'Volkswagen']
```

La liste peut être aussi triée dans un ordre descendant en passant l'argument **reverse=True**

```
cars.sort(reverse=True)
```

```
cars
```

```
['Volkswagen', 'Suzuki', 'Renault', 'Peugot', 'Mercedes', 'Audi']
```

Trier une liste d'une manière temporaire

La fonction **sorted** permet de trier une liste d'une manière temporaire sans affecter l'ordre réel dans la liste.

```
cars = ['Audi', 'BMW', 'Mercedes', 'Honda', 'Toyota', 'Suzuki', 'Renault', 'Peugot']
```

```
sorted(cars)
```

```
['Audi', 'BMW', 'Honda', 'Mercedes', 'Peugot', 'Renault', 'Suzuki', 'Toyota']
```

```
cars
```

```
['Audi', 'BMW', 'Mercedes', 'Honda', 'Toyota', 'Suzuki', 'Renault', 'Peugot']
```

Afficher une liste dans l'ordre inverse d'une manière permanente

Afin d'inverser l'ordre d'une liste, on peut utiliser la fonction **reverse**.

```
cars.reverse()
```

```
cars
```

```
['Peugot', 'Renault', 'Suzuki', 'Toyota', 'Honda', 'Mercedes', 'BMW', 'Audi']
```

Afficher une liste dans l'ordre inverse d'une manière temporaire

On peut inverser l'ordre des éléments d'une liste d'une manière temporaire à l'aide de la fonction **reversed**.

```
list(reversed(cars))
```

```
['Audi', 'BMW', 'Mercedes', 'Honda', 'Toyota', 'Suzuki', 'Renault', 'Peugot']
```

Afficher la longueur d'une liste On peut trouver la longueur d'une liste en utilisant la fonction **len**.

```
len(cars)
```

```
8
```

Impossible d'établir une connexion avec le service reCAPTCHA. Veuillez vérifier votre connexion Internet, puis actualiser la page pour afficher une image reCAPTCHA.