

# Listes : manipulation et méthodes

## Algorithmique et structuration de données

Léna Gaubert

[lana.gaubert@proton.me](mailto:lana.gaubert@proton.me)

L3 TIM/TAL @ INALCO

# Listes

## Définition et manipulation

Une **liste** (`list`) est une structure de données qui contient une série d'éléments (les éléments peuvent être de *types différents*). On définit une liste entre **crochet** : ses éléments sont séparés par une **virgule**.

```
students = ["Ava", "Max", "Chaker"]  
grades = [3.9, 3.5, 4]  
mixte = [5, "Alan", ["maths", "python"]]
```

Une liste est **ordonnée**. On accède aux éléments d'une liste grâce à leurs indices correspondants (*indexation*). On accède au *nombre* d'éléments de la liste avec la fonction `len()`.

```
names = ["Eve", "Julie", "Eli"]  
print(len(names))  
print(names[0])  
print(names[-1])
```

À l'instar des chaînes de caractères, on peut extraire une tranche d'une liste (*slicing*).

```
names = ["Eve", "Julie", "Eli", "Hope"]  
print(names[:2])
```

Tester le code ci-dessous. Que se passe-t-il ? Quelles sont les opérations valides sur les listes ?

```
l = [1, 2, 3]
m = [4, 5, 6]
n = l + m
print(n)
print(l * 3)
print(l + 2)
```

# Listes

## Méthodes et itérations

# Taille dynamique

Contrairement à d'autres langages de programmation, en Python, les listes ont une *taille dynamique* : elles peuvent grandir ou rétrécir dynamiquement.

```
l = ["Fortran", "C", "PHP"]  
l.remove("C")  
l.append("Python")  
print(l)
```

- `append()` pour *ajouter* un élément à la fin d'une liste ;
- `remove()` pour *enlever* un élément de la liste.

Les listes sont des objets *itérables*, qu'on peut manipuler facilement à travers une boucle.

## Méthode 1 : itération sur les éléments d'une liste

```
students = [  
    "Cynthia",  
    "Bintou", "Djamila"  
]  
  
for stud in students:  
    print(f"Hello {stud}!")
```

## Méthode 2 : itération sur les indices d'une liste

```
students = [  
    "Cynthia",  
    "Bintou", "Djamila"  
]  
  
n = len(students)  
for i in range(n):  
    print(f"Hello {students[i]}!")
```

Il existe davantage de **méthodes** pour manipuler des listes :

- `.insert(index, element)` : insère `element` à l'indice `index`.
- `.sort()` : trie les éléments d'une liste
- `.reverse()` : inverse une liste
- `.count(item)` : compte le nombre d'éléments égaux à `item` dans la liste.

Et bien plus encore !

**time to practice! ( $\cdot\omega<$ )☆**