

Les listes

Création express

- Initialiser une liste vide : le `_` indique qu'on ne souhaite pas utiliser la variable d'itération

```
l = [0 for _ in range(5)]  
print(l)  
# [0, 0, 0, 0, 0]
```

- Création d'une liste simple

```
l = [i for i in range(1, 10, 2)]  
print(l)  
# [1, 3, 5, 7, 9]
```

- Création d'une liste de listes

```
>>> l = [[i for i in range(1, 10, 2)] for _ in range(5)]  
>>> print(l)  
[[1, 3, 5, 7, 9], [1, 3, 5, 7, 9], [1, 3, 5, 7, 9], [1, 3, 5, 7, 9], [1, 3, 5, 7, 9]]
```

Fonctionnalités

Ajout

Pour ajouter un élément à la fin d'une liste, on utilise `append`.

```
l = [0, 1]  
l.append(3)  
print(l)  
# [0, 1, 3]
```

Suppression

Pour supprimer le dernier élément, on utilise `pop`.

```
l = [0, 1]  
print(l.pop())  
# 1  
print(l)  
# [0]
```

La fonction `range`

- `range(10)` crée la liste `[0,1,...,9]`
- `range(3,10)` crée la liste `[3,4,...,9]`
- `range(3,10,2)` crée la liste `[3,5,7,9]`
- `range(10,0,-1)` crée la liste `[10,9,...,2,1]`

Arguments:

```
range( indice_debut, indice_fin, pas)
```

La fonction len

Renvoie la taille de la liste

```
l = [0, 1, 2]
print(len(l))
# 3
```

La fonction slice

Envoyer une partie de liste (slice operation):

Syntaxe :

- list[index_debut:index_fin]
- index_debut inclus, index_fin exclus

Exemple :

```
l = [1, 2, 3, 4, 5]

print(l[0:2])
# [1, 2]

print(l[1:])
# [2, 3, 4, 5]

print(l[:3])
# [1, 2, 3]
```