

### Sequences (Tuples/Liste/Ensemble)

```
a,b = (2,1) print(a+b) °°°°a,b = {2,3} print(a+b) °°°° a,b,_,c = [4,5,9,10] print(a+b+c)   matrice = [[2,3], [4,5]] for c1,c2 in matrice : print(c1+c2)
3°°°°° 5°°°°° 19   5 9 suma de los valores de las matrices
```

### Parcours de listes

enumerate et range Il est indispensable de comprendre les parcours de séquences avec range, enumerate et zip : Enumerate retourne un générateur qui renvoie des tuples (indice, valeur). Pour s'en convaincre, on peut récupérer l'ensemble de ces tuples avec list()

```
l = [1,2,3]°°°° list(enumerate(l))
```

```
[(0, 1), (1, 2), (2, 3)]
```

```
l = [1,2,3] °° for i,v in enumerate([1,2,3]) : °° print(i,v)
```

```
0 1 1 2 2 3
```

Range est aussi un générateur qui lui renvoie les nombres compris entre a inclus et b exclus avec un pas de n : range(a,b,n) Notez que le pas et la valeur initiale sont optionnelles (1 et 0 par défaut respectivement).

```
range(4) == range(0,4,1) Rta:True
```

```
range(1,4) == range(4) Rta:False
```

```
list(range(1,100, 10)) Rta: [1, 11, 21, 31, 41, 51, 61, 71, 81, 91]
```

L'utilisation de range pour parcourir les listes peut engendrer des erreurs.

```
%%script python --no-raise-error °° l1 = [7,8,9,90] °°l2 = [4,5,6,90,3992] °° for i in range(0, len(l2)): °° print(l1[i], l2[i])
```

```
7 4 8 5 9 6 90 90 °°Traceback (most recent call last): File "<stdin>", line 6, in <module> IndexError: list index out of range
```

C'est notamment le cas quand on veut parcourir deux listes de taille différente en même temps. Dans ce cas soit on s'arrête au minimum des deux listes :

```
l1 = [7,8,9,90]°° l2 = [4,5,6, 90, 3992] °°°for i in range(min(len(l1), len(l2))): °° print(l1[i], l2[i])
```

```
7 4°° 8 5°° 9 6°° 90 90 el resultado es como dos listas
```

Soit on utilise la fonction zip :

```
l1 = [7,8,9,90] l2 = [4,5,6, 90, 3992] for e1,e2 in zip(l1,l2): print(e1,e2)
```

```
7 4°° 8 5 °°°9 6 °°°90 90
```

```
list(zip(l1,l2))
```

```
[(7, 4), (8, 5), (9, 6), (90, 90)]
```

### Rappel sur les chaînes

```
t1 = len(l1) °° t2 = len(l2)°° print(-   On peut même écrire du HTML :
f"La plus petite taille est {min(t1,t-2)}")
```

```
La plus petite taille est 4   from IPython.core.display import HTML °°a = f"" Ceci est un mot en <strong>gras</strong> <hr/>°°
<div style="color : brown; background : green"> Ceci est une boîte </div> "" display(HTML(a))
```

### Exemples "simples" avec des entiers

```
a = 2 °°b = a°° b = 3°°   L'opération + sur les entiers génère un nouvel entier ce qui fait que b pointe sur une nouvelle zone (2+1) après
print(a) Rta: 2   l'opération +=). Ainsi a n'est pas modifié
```



### Exemples "simples" avec des entiers (cont)

Ici on peut vérifier que a et b ne sont pas identiques (pointent sur la même zone mémoire) avec is

```
a is b Rta: False
```

```
a = 2 b = a b += 1 print(a)
```

```
a = 2 b = a print(id(a)) print(id(b)) a is b Rta 1909773723984
1909773723984 True
```



By **alejcia**

[cheatography.com/alejcia/](https://cheatography.com/alejcia/)

Not published yet.

Last updated 27th October, 2022.

Page 3 of 3.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish

Yours!

<https://apollopad.com>

