

Problématique :

<pre> 1 l = [] 2 s = "" 3 while s != "STOP": 4 s = input() 5 l.append(s) 6 print(l)</pre>	<p>Que fait ce programme ?</p> <p>-----</p> <p>-----</p> <p>-----</p> <p>-----</p> <p>Qu'affiche ce programme si l'utilisateur tape au clavier les valeurs suivantes :</p> <p>a</p> <p>b</p> <p>c</p> <p>STOP</p> <p>-----</p>
---	--

Qu'affiche ces programmes ?

<pre> 9 l = ["tata", "toto"] 10 l.append("tutu") 11 print(l)</pre>	
<pre> 13 l = ["tata"]*3 14 l.insert(1, "tutu") 15 print(l)</pre>	
<pre> 17 l = ["tata", "toto", "titi"] 18 l.pop() 19 print(l)</pre>	
<pre> 21 l = ["tata", "toto", "titi"] 22 l.pop(1) 23 print(l)</pre>	

```

25 l = ["tata", "toto"]
26 l.append("tutu")
27 l[2] = "titi"
28 print(l)

30 l = ["tata", "toto", "tutu", "titi"]
31 print(l[3])
32 print(len(l))

```

Méthodes et fonctions pour manipuler des listes :

Création	
Lecture	
Ajout	
Modification	
Suppression	
Taille	
Afficher le contenu	

Compléter les programmes suivants.

```

35 l = ["N", "A", "S", "A"]
36
37
38
39
40 print(l) # Doit afficher ["N", "S", "I"]
41
42 t = ["2", "n", "d"]
43
44
45
46
47 print(t) # Doit afficher ["1", "2", "3"]

```

A programmer : Coder la fonction demandée et à chaque fois, écrire un ou plusieurs cas de test pour valider votre fonction.

Exercice 1 : Écrire une fonction **premier** qui prend en paramètre une *liste* et qui renvoie la valeur de son premier élément.

```
def premier(liste):  
    # à compléter  
  
#Tests  
a = [1,2,3,4]  
print(premier(a)) # affiche 1  
b = [99,98,97]  
print(premier(b)) # affiche 99
```

Exercice 2 : Écrire une fonction **estVide** qui prend en paramètre une *liste* et renvoie vrai si la taille du tableau est égale à zéro, et renvoie faux sinon.

Exercice 3 : Écrire une fonction **dernier** qui prend en paramètre une *liste* et qui renvoie la valeur de son dernier élément.

Exercice 4 : Écrire une fonction **initialisationParComprehension** qui prend en paramètre une valeur *debut*, une valeur *fin* et une valeur *pas* et qui renvoie une nouvelle liste contenant les valeurs de *debut* à *fin* allant de *pas* en *pas*.

Exercice 5 : Écrire une fonction **suppressionIntervale** qui prend en paramètre une *liste*, une valeur *debut*, une valeur *fin* et qui supprime de la liste les valeurs aux indices appartenant à [*debut* ; *fin*].

Exercice 6 : Écrire une fonction **echange** qui prend en paramètre une *liste*, un *indice1*, un *indice2* et qui échange dans la liste les valeurs contenues aux indices *indice1* et *indice2*.