

---

# IngeSUP - TD 05 - Listes et chaînes 2

---

*"Nous ferons mieux de nous mettre plus haut sur notre propre liste de choses à faire."*

Michelle Obama

---

## Exercice 05.1

---

Pour chacune des questions suivantes, écrire un programme qui remplit une liste d'entiers demandés à l'utilisateur à l'aide d'une boucle et de la fonction input), puis effectuez les actions suivantes :

1. Complétez le programme pour afficher la liste en colonne. C'est-à-dire que chaque élément est du type i : liste[i] si i est l'indice et liste le nom de la liste.

Entrée [1]:



```
L=[]
taille=int(input("combien d'éléments dans la liste ?"))

for i in range(taille):
    nombre=int(input("entrez un nombre"))
    L.append(nombre)

print(L)

for i in L:
    print(i)
```

```
combien d'éléments dans la liste ?3
entrez un nombre1
entrez un nombre10
entrez un nombre100
[1, 10, 100]
1
10
100
```

1. Modifiez le programme pour qu'il compte le nombre de multiples de 3 dans la liste

Entrée [3]:



```
L=[]
taille=int(input("combien d'éléments dans la liste ?"))

for i in range(taille):
    nombre=int(input("entrez un nombre"))
    L.append(nombre)

print(L)

compteur=0
for i in L:
    if i%3==0:
        compteur += 1

print("le nombre de multiple de 3 dans la liste est :", compteur)
```

```
combien d'éléments dans la liste ?5
entrez un nombre1
entrez un nombre2
entrez un nombre3
entrez un nombre6
entrez un nombre45
[1, 2, 3, 6, 45]
le nombre de multiple de 3 dans la liste est : 3
```

2. Modifiez le programme pour qu'il calcule la somme de toutes les valeurs paires issues de la liste

Entrée [13]:



```
L=[]
continuer="oui"
#alternative ou on ne demande pas le nombre mais si l'utilisateur veut continuer a chaque j
while continuer=="oui":
    nb=int(input("entrez un nombre"))
    L.append(nb)
    continuer=input("voulez vous continuer ? (oui ou non)")

print(L) #affiche en ligne

somme=0
for i in L:
    if i%2==0:
        somme += i

print(somme)
```

```
entrez un nombre12
voulez vous continuer ? (oui ou non)oui
entrez un nombre78
voulez vous continuer ? (oui ou non)oui
entrez un nombre55
voulez vous continuer ? (oui ou non)oui
entrez un nombre3
voulez vous continuer ? (oui ou non)non
[12, 78, 55, 3]
90
```

3. Modifiez le programme pour qu'il indique si la valeur de la moyenne arithmétique des entiers entrés par l'utilisateur est supérieure à 10.

Entrée [16]:



```
L=[]
continuer="oui"

while continuer=="oui":
    nb=int(input("entrez un nombre"))
    L.append(nb)
    continuer=input("voulez vous continuer ? (oui ou non)")

print(L) #affiche en ligne

moyenne=sum(L)/len(L)
if moyenne>=10:
    print("vous avez la moyenne")
else:
    print("vous n'avez pas la moyenne")
```

```
entrez un nombre12
voulez vous continuer ? (oui ou non)oui
entrez un nombre18
voulez vous continuer ? (oui ou non)oui
entrez un nombre3
voulez vous continuer ? (oui ou non)non
[12, 18, 3]
vous avez la moyenne
```

4. Modifiez le programme précédent pour qu'il remplisse une nouvelle liste avec toutes les valeurs comprises dans l'intervalle [50,70].

Entrée [17]:



```
L=[]
continuer="oui"

while continuer=="oui":
    nb=int(input("entrez un nombre"))
    L.append(nb)
    continuer=input("voulez vous continuer ? (oui ou non)")

print(L) #affiche en ligne

L2=[]
for i in L:
    if 50<=i<=70:
        L2.append(i)

print(L2)
```

```
entrez un nombre34
voulez vous continuer ? (oui ou non)oui
entrez un nombre89
voulez vous continuer ? (oui ou non)oui
entrez un nombre67
voulez vous continuer ? (oui ou non)oui
entrez un nombre61
voulez vous continuer ? (oui ou non)oui
entrez un nombre8
voulez vous continuer ? (oui ou non)non
[34, 89, 67, 61, 8]
[67, 61]
```

5. Modifiez le programme précédent pour qu'il affiche la liste à l'envers, sans sauter la ligne et sans faire appel à une autre liste.

Entrée [31]:



```
L=[]
continuer="oui"

while continuer=="oui":
    nb=int(input("entrez un nombre"))
    L.append(nb)
    continuer=input("voulez vous continuer ? (oui ou non)")

print(L) #affiche en ligne
L.reverse()
print(L)

# version "a la main" sans utiliser reverse:
liste=[1,2,3,4,5,6]
liste2=[]
for i in range(len(liste)):
    liste2.append(liste[-i-1])
print(liste2)
```

```
entrez un nombre1
voulez vous continuer ? (oui ou non)oui
entrez un nombre2
voulez vous continuer ? (oui ou non)oui
entrez un nombre3
voulez vous continuer ? (oui ou non)oui
entrez un nombre4
voulez vous continuer ? (oui ou non)non
[1, 2, 3, 4]
[4, 3, 2, 1]
[6, 5, 4, 3, 2, 1]
```

---

## Exercice 05.2

---

1. Ecrire un programme qui calcule et stocke dans une liste `liste2` le double des éléments contenus dans une liste `liste` en utilisant la syntaxe des compréhensions de liste.

Entrée [6]:



```
liste=[2,4,3,1,5]
liste2=[]

for i in liste:
    liste2.append(i*2)

print(liste2)

#en compréhension cela donne :
liste = [2,4,3,1,5]
liste2 = [i*2 for i in liste]
print(liste2)
```

```
[4, 8, 6, 2, 10]
[4, 8, 6, 2, 10]
```

2. Ecrire un programme qui stocke dans une liste L2 les éléments impairs d'une liste L en partant du début de la liste en utilisant la syntaxe des compréhensions de liste.

Entrée [32]:



```
#en non compréhension cela donne :
liste = [2,4,3,1,5]
liste2=[]
for i in liste:
    if i%2==1:
        liste2.append(i)
print(liste2)

#en compréhension
liste = [2,4,3,1,5]
liste2 = [i for i in liste if i%2==1]
print(liste2)
```

```
[3, 1, 5]
[3, 1, 5]
```

3. Ecrire un programme qui remplisse une liste L2 avec des nombres pairs parmi les 5 premières valeurs d'une liste L en utilisant la syntaxe des compréhensions de liste. Afficher L2 .

Entrée [12]:



```
liste = [2,4,3,1,6,4,3,2,3,4,3,1]
liste2 = [i for i in liste[:5] if i%2==0]
print(liste2)
```

```
[2, 4, 6]
```

---

## Exercice 05.3

---

Ecrire un programme qui découpe, dans une liste `mots`, une chaîne de caractères en mot (c'est à dire une chaîne séparée par des espaces) et renvoie le nombre de caractères dans une seconde liste `nb_caracteres` dans le même ordre que la liste des mots `mots`.

Remarques :

1. On fera bon usage de la fonction `split()`
2. On pourra utiliser la syntaxe des compréhensions de liste
3. La liste `mots` est une liste qui peut servir d'intermédiaire

Entrée [34]:



```
texte="Bonjour comment ça va ?"
mots=texte.split(" ")
print(mots)
```

```
['Bonjour', 'comment', 'ça', 'va', '?']
```

Entrée [36]:



```
#version en compréhension
texte="Bonjour comment ça va ?"
liste2 = [mot for mot in texte.split(" ")]
print(liste2)
```

```
['Bonjour', 'comment', 'ça', 'va', '?']
```

---

## Exercice 05.4

---

Ecrire un programme qui génère les chaînes `tata`, `tete`, `titi`, `toto` et `tutu` à partir de la chaîne `t_t_` ou n'importe quelle chaîne composée de consonnes et du caractère `_` qui sera remplacé par toutes les voyelles `aeiou`. L'ensemble des variantes ainsi construite sera regroupé dans une liste.

Remarques :

1. On fera bon usage de la fonction ``replace``
2. On pourra utiliser la syntaxe des compréhensions de liste



Entrée [17]:



```
chaine="t_t_"
L=[]
for i in "aeiou":
    texte=chaine.replace("_",i)
    L.append(texte)

print(L)
```

['tata', 'tete', 'titi', 'toto', 'tutu']

Entrée [20]:



```
L=["t_t_".replace("_",i) for i in "aeiou"]
print(L)
```

['tata', 'tete', 'titi', 'toto', 'tutu']

---

## Exercice 05.6

---

Ecrire un programme qui affecte la valeur `True` à une variable *toute* si une liste *liste* possède des valeurs toutes supérieures à 1.

Entrée [30]:



```
L=[1,23,-4,3,6,5]

toute=True

for i in L:
    if i<1:
        toute=False
        break

print(toute)
```

False

---

## Exercice 05.7

---

1. Ecrire un programme qui effectue une permutation circulaire de la liste. Chaque élément prend la valeur de son élément de droite et la première valeur se retrouve en dernier : Par exemple si la liste est [8, 24, 48, 2, 16] après l'exécution du programme elle sera [24, 48, 2, 16, 8]

Entrée [22]:



```
L=[1,2,3,4,5]
L2=L[1:]
L2.append(L[0])
print(L2)

#autre manière de faire:
L=[1,2,3,4,5]
L2=[]
for i in range(len(L)):
    L2.append(L[-i])

print(L2)
```

[2, 3, 4, 5, 1]

2. Ecrire un programme qui modifie une liste par miroir, c'est-à-dire que tout élément situé à la place i en partant du début est échangé avec son correspondant à la place i en partant de la fin. Par exemple si la liste est [8, 24, 48, 2, 16] après l'exécution du programme elle sera [16, 2, 48, 24, 8].

Entrée [27]:



```
L=[1,2,3,4,5]
L.reverse()
print(L)

#autre manière de faire "a la main":
L=[1,2,3,4,5]
L2=[]
for i in range(len(L)):
    L2.append(L[-i-1])

print(L2)
```

[5, 4, 3, 2, 1]  
[5, 4, 3, 2, 1]

3. Dédurre de la question précédente un programme qui détermine si une liste est symétrique.

**Indication:** Une liste est symétrique si elle est égale à son miroir

Entrée [29]:



```
L=[1,2,3,2,1]
L2=L.copy()
L.reverse()
print(L==L2)
```

True

---

## Corrigé du TD 05

Vous pouvez retrouver le corrigé de ce TD [ici \(Corrig%C3%A9s/Corrig%C3%A9\\_TD%2004.ipynb\)](#).