

## TP N°2 Structure de données

### Exercice N°1 : Manipulation des listes

```
>>> list=['lundi', 2, 'janvier']
>>> print(list)
>>> list[0] # Mettre print(list[0]) si vous n'êtes pas en ligne de commandes
>>> list[-1]
>>> print(list[2])
>>> len(list)
>>> list.append(2010)
>>> list
>>> list[3]=list[3]+1
>>> list[3]
>>> del list[0]
>>> list
>>> list.insert(0,'mardi')
>>> list
>>> 'mardi' in list
>>> 'lundi' in list
>>> list.index("mardi")
>>> list2=list[1:3]
>>> list2
>>> list3=list[:2]
>>> list3
>>> list4=list[1:]
>>> list4
>>> list5=list[-3:-1]
>>> list5
>>> list6 = list[: -1]
>>> list6
>>> list3=list3 + [2011]
>>> list3
>>> list7=3*list
>>> list7
>>> list.extend([3,4])
>>> list
>>> list=list.pop(0)
>>> list
>>> list=[1,2,3]
>>> list2=list # Attention list et list2 correspondent a la meme liste!!
>>> list
>>> list.pop(1)
>>> list
>>> list2
>>> list=[1,2,3]
>>> list2=list.copy() # list2 est une copie de list
>>> list.pop(1)
>>> list
>>> list2
```

```
>>> list=[1,"ab",[1,True]] # liste imbriquee
>>> list[2]
>>> print(list(range(10)))
>>> print(list(range(1,10)))
>>> print(list(range(1,10,3)))
>>> help(list) # pour l'aide sur les listes
```

### Exercice N°2 : Saisies d'une liste au clavier

```
>>> list = [input(),input(),input()]
>>> list
>>> list=[x for x in input("Saisir les elements de la liste separees
par une virgule (ex. 1,2,\"abc\") :").split(',')]] # Taper <Enter>
>>> list
>>> list=[int(x) for x in input("Saisir des entiers separees
par une virgule (ex. 1,2,\"abc\") :").split(',')]] # Taper <Enter>
>>> list
```

### Exercice N°3 : Dictionnaire

Tapez le code suivant et observez le résultat.

```
>>> dico = {} # dictionnaire vide
>>> dico['computer'] = 'ordinateur'
>>> dico['mouse'] = 'souris'
>>> dico['keyboard'] = 'clavier'
>>> print(dico)
>>> print(dico.keys())
>>> print(dico.values())
>>> del dico['mouse']
>>> print(dico)
>>> print(len(dico))
>>> dico.__contains__('computer')
>>> print(dico.items())
>>> for clef in dico: # Ne pas oublier les 2 points
...     print(clef) # Attention ne pas oublier la tabulation!!
>>> for clef in dico: # Ne pas oublier les 2 points
...     print(clef, dico[clef]) # Attention ne pas oublier la tabulation!!
>>> for clef, value in dico.items(): # Ne pas oublier les 2 points
...     print(clef, value) # Attention ne pas oublier la tabulation!!
>>> dico2={'ordinateur': 'computer', 'souris' : 'mouse'}
>>> print(dico2)
>>> dico2=dico # Attention dico et dico2 correspondent au même dictionnaire!
>>> print(dico)
>>> del dico2['computer']
>>> print(dico)
```

## Licence Ingénierie Logicielle et Système d'Information

```
>>> print(dico2)
>>> dico3=dico.copy() # dico3 est une copie du dictionnaire reference par dico!
>>> del dico3['keyboard']
>>> print(dico)
>>> print(dico3)
>>> help(dict) # pour l'aide sur les dictionnaires
```

### Exercice N°4 : Définir une fonction sans paramètre

Tapez le code suivant et observez le résultat.

```
>>> def function(): # Definition de fonction sans parametre - Ne pas oublier les :
...     n=10 # Faire une tabulation en debut de ligne
...     while n>0: # Faire une tab. en debut de ligne et ne pas oublier les :
...         print(n/2, n%2) # Faire 2 tabulations en debut de ligne
...         n=n-1 # Faire 2 tabulations en debut de ligne
# Tapez encore une fois <Enter> si vous etes en ligne de commande
>>>function() # Appel de la fonction
```

### Exercice N°5: Définir une fonction sans paramètre qui appelle une autre fonction

Tapez le code suivant et observez le résultat.

```
>>> def function2(): # Ne pas oublier les : et une tab. sur la ligne suivante
...     print('Affichage du resultat et du reste de la division des entiers de 10 \
...     function() # Faire une tabulation en debut de ligne
# Tapez encore une fois <Enter> si vous etes en ligne de commande
>>> function2()
```

### Exercice N°6: Définir une fonction à 1 paramètre

Tapez le code suivant et observez le résultat.

```
def function(n): # Definition d'une fonction a un parametre
...     while n>0: # Faire une tab. en debut de ligne et ne pas oublier les :
...         print(n/2, n%2) # Faire 2 tabulations en debut de ligne
...         n=n-1 # Faire 2 tabulations en debut de ligne
# Tapez encore une fois <Enter> si vous etes en ligne de commande
>>> function(3) # Appel de la fonction avec le parametre 3
# Tapez encore une fois <Enter> si vous etes en ligne de commande
>>> function(5) # Appel de la fonction avec le parametre 5
```

### Exercice N°7:

Tapez le code suivant et observez le résultat.

```
>>> def afficher3fois(arg):
...     print(arg, arg, arg)
# Tapez encore une fois <Enter> si vous Ãªtes en ligne de commande
>>> afficher3fois(3)
>>> afficher3fois('exemple')
>>> afficher3fois([3,4])
>>> afficher3fois(3*4)
```

### Exercice N°8 : Valeur par défaut des paramètres

Tapez le code suivant et observez le résultat.

```
>>> def function(entree,diviseur=2): # fct. avec valeur par default pour diviseur
...     while(entree>0):
...         print(entree/diviseur, entree%diviseur)
...         entree=entree-1
# Tapez encore une fois <Enter> si vous Ãªtes en ligne de commande
>>> function(10)
```

### Exercice N°9 :

Tapez le code suivant et observez le résultat.

```
>>> def function(entree=10,diviseur=2):
...     while(entree>0):
...         print(entree/diviseur, entree%diviseur)
...         entree=entree-1
# Tapez encore une fois <Enter> si vous Ãªtes en ligne de commande
>>> function()
>>> function(4,2)
>>> function(diviseur=4,entree=12)
```

### Exercice N°10 :

Tapez le code suivant et observez le résultat.

```
>>>def multiplication(n,p):
...     return n*p # pour retourner une valeur
# Tapez encore une fois <Enter> si vous Ãªtes en ligne de commande
>>> a=multiplication(3,6)
>>> a
```