

Exercice 1 : Problème de casse ?

En informatique, la casse désigne le fait de distinguer les lettres majuscules des lettres minuscules.

Le code ci-contre permet de créer 2 dictionnaires :

```
def casse() :
    minuscules = 'abcdefghijklmnopqrstuvwxyz'
    majuscules = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
    upper = {}
    lower = {}
    for i in range(26) :
        l = minuscules[i]
        L = majuscules[i]
        upper[l] = L
        lower[L] = l
    return upper , lower

# main -----
upper , lower = casse()
```

Ce code est exécuté. Pour chacune des commandes données dans le tableau ci-dessous, indiquer le résultat de l'exécution :

Commande	Résultat
<code>upper['a']</code>	
<code>upper['A']</code>	
<code>lower['A']</code>	
<code>len(upper)</code>	
<code>for c in upper : print(c , end = " ")</code>	
<code>for c in upper : print(c , upper[c] , end = " ")</code>	
<code>upper.clear() print(upper)</code>	

Exercice 2 : Changer les lettres d'un mot ?

On complète le code précédent :

- 1- Quelle valeur contient la variable *m* après exécution ?
- 2- Compléter le tableau ci-dessous donnant le contenu des variables au cours de l'exécution :

<i>mot</i>	<i>new</i>	<i>c</i>

```
def casse() :
    minuscules = 'abcdefghijklmnopqrstuvwxyz'
    majuscules = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
    upper = {}
    lower = {}
    for i in range(26) :
        l = minuscules[i]
        L = majuscules[i]
        upper[l] = L
        lower[L] = l
    return upper , lower

def majuscule(mot) :
    new = ''
    for c in mot :
        new = new + upper[c]
    return new

# main -----
upper , lower = casse()
m = majuscule("bonjour")
```

Exercice 3 : L'informatique ça peut servir !

- 1- Quel résultat à l'écran donne l'exécution de ce code ?
- 2- Le contenu des dictionnaire est accessible à l'intérieur de la fonction *traduction()*. Pourquoi ?

```
def anglais():
    uk = {}
    uk['je'] = 'i'
    uk['aime'] = 'love'
    uk['toi'] = 'you'
    return uk

def allemand():
    d = {'je':'ich' , 'aime' : 'liebe' , 'toi' : 'dich'}
    return d

def italien():
    d = {}
    d['je'] = "io"
    d['aime'] = 'amore'
    d['toi'] = 'voi'
    return d

def traduction(m1 , m2 , m3):
    print(uk[m1],uk[m2],uk[m3])
    print(g[m1],g[m2],g[m3])
    print(it[m1],it[m2],it[m3])

# main
uk = anglais()
g = allemand()
it = italien()
traduction("je","aime","toi")
```