

## TP : Les fichiers en PYTHON

---

### Exercice 1 Calcul de moyenne

1. Écrire une fonction PYTHON `remplir_fichier(nom_fichier, n)` qui permet de créer et de remplir un fichier texte contenant les noms et les notes d'un groupe d'étudiants. Chaque ligne du fichier contient deux colonnes séparées par " : ". La première colonne représente le nom de l'étudiant et la deuxième colonne représente la note.
2. Écrire une fonction PYTHON `calculer_moyenne(nom_fichier)` qui permet de lire un fichier texte contenant des valeurs sur deux colonnes séparées par le caractère " :" sous la forme `nom :note`. Cette fonction doit retourner la moyenne des notes des étudiants du fichier.

### Exercice 2 Municipalités

L'Instance Supérieure Indépendante pour les élections gère les informations concernant les différentes municipalités tunisiennes à travers un fichier texte "munip.txt". Ce fichier contient pour chaque municipalité son nom, son gouvernorat (un code de 1 à 24) et sa population. Ainsi chaque ligne du fichier est représentée sous cette forme : `#nom;gouvernorat;population`

Exemple du fichier 'munip.txt' :

```
#Ariana;2;114486
#Sidi Hassine;1;109690
#Mohamedia;3;106167
#El Mourouj;3;104586
#Gafsa;4;95242
#Raoued;2;94691
...
```

1. Écrire une fonction python `liste_munip(fichier)` qui à partir du fichier "munip.txt" crée et retourne une liste de chaînes de caractères où chaque chaîne correspond aux différentes informations d'une municipalité. Il est à noter que les chaînes de caractères ne contiennent pas les symboles '#' et '\n'.
2. Écrire une fonction python `tuple_munip(fichier)`, qui à partir du fichier "munip.txt", crée et retourne un tuple où chaque élément est aussi un tuple contenant le nom, le gouvernorat, et la population (avec nom et gouvernorat de type chaîne et population de type entier).

3. Écrire une fonction python `lc_munip(fichier)`, qui à partir du fichier "munip.txt" retourne la municipalité ayant le nom le plus long et la municipalité ayant le nom le plus court.
4. Écrire une fonction python, `seuil_munip(fichier, N)`, qui à partir du fichier "munip.txt" retourne la liste des municipalités ayant un nombre d'habitant inférieur à une valeur  $N$ . Cette liste comprendra des couples de la forme (nom, gouvernorat).
5. Écrire une fonction python `gouv_munip(fichier)`, qui à partir du fichier "munip.txt" retourne un dictionnaire contenant le nombre de municipalités et la population de chaque gouvernorat. Ainsi, le code du gouvernorat représentera la clé du dictionnaire et le tuple (nombre de municipalités, population par gouvernorat) représentera la valeur. Sur notre exemple, le dictionnaire sera : `{ '1' : (1, 109690), '2' : (2, 209177), '3' : (2, 210753), '4' : (1, 95242) }`
6. Écrire une fonction python `créer_fich(fichier1, fichier2)` qui à partir du fichier "munip.txt" permet de créer le fichier2 tel que fichier2 se présente comme suit :

```
CodeGouv (NbMuni,NbPop)
3 (2,210753)
1 (1,109690)
2 (2,209177)
4 (1,95242)
```