

**OBJECTIFS** : L'objectif de ce TP est d'appliquer ce qui a été vu en cours sur le chapitre des fichiers.

1. Lecture sur un fichier plus volumineux :

⇒ Télécharger le fichier *tp10.zip* à partir de *nsibrantly.fr* et décompresser le fichier *world\_nsi.txt* dans votre répertoire de travail.

⇒ Ce fichier *world\_nsi.txt* contient 150 000 lignes correspondant à tous les élèves de **Nsi** de notre planète. Jeter un coup d'œil dans ce fichier pour vérifier que les données y sont écrites dans le même format que dans le fichier *exercice\_1.txt*. Utiliser votre code pour déterminer la moyenne de ces notes et les nom et prénom de l'élève dont la note est la plus proche de la moyenne.

Le résultat de l'exécution sera normalement :

```
>>> (executing file "exercice_3.py")
La moyenne est de : 10.04
EZZAKI winnie a comme note : 10.04
```

```
# Fonctions
def lecture(nom) :
    """
    Lit le fichier dont le nom est en argument.
    Retourne une liste double
    """
    f = open(nom, "r", encoding='utf8')
    lst = []
    c = f.readline()
    while c != "" :
        c = c[:-1]
        l = c.split(",")
        lst.append(l)
        c = f.readline()
    f.close()
    return lst

def moyenne(liste) :
    """
    Renvoie la moyenne des notes des élèves
    enregistrés dans la liste qui est en
    argument.
    """
    tot = 0
    for i in range(1, len(liste)) :
        tot = tot + float(liste[i][2])
    moy = tot / (len(liste)-1)
    return moy
```

*Corrigé possible*

```
def eleve_moyen(liste,moy) :
    """
    Renvoie l'indice de l'élève de la liste
    mise en argument pour lequel la note est la
    plus proche de la valeur moyenne des notes
    mise en argument.
    """
    min = float(liste[1][2]) - moy
    if min < 0 : min = -min
    indice = 1
    for i in range(1,len(liste)) :
        ecart = float(liste[i][2]) - moy
        if ecart < 0 : ecart = -ecart
        if ecart < min :
            min = ecart
            indice = i
    return indice

# Main
eleves = lecture("world_nsi.txt")
moy = moyenne(eleves)
print(f"La moyenne est de : {round(moy,2)}")
i = eleve_moyen(eleves,moy)
print(f"{eleves[i][0]} {eleves[i][1]} a comme note : {eleves[i][2]}")
```

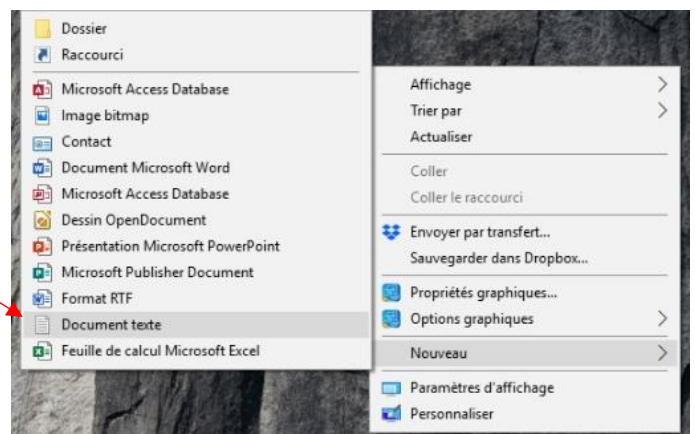
*Corrigé possible*

## 2. Traitement d'un fichier .csv téléchargé sur un data public. :

⇒ Aller sur le site github : <https://github.com/fivethirtyeight/data> qui propose des datas U.S. originaux.

Aller dans la catégorie  `births` et copier le fichier  `US_births_2000-2014_SSA.csv` en cliquant sur Raw

pour ensuite copier l'ensemble de la page (ctrl A , puis ctrl C) dans un fichier texte que vous nommerez ***naissance.txt***



Ce fichier contient pour chacune des années entre 2000 et 2014, pour chaque mois de l'année et jour du mois, le nombre de naissance aux U.S.A. On trouve aussi sur chacune

```
year,month,date_of_month,day_of_week,births
2000,1,1,6,9083
2000,1,2,7,8006
2000,1,3,1,11363
2000,1,4,2,13032
2000,1,5,3,12558
```

des lignes, juste avant le nombre de naissance, le jour de la semaine : 1 pour lundi, 2 pour mardi, .... et 7 pour dimanche.

⇒ Créer un code composé du programme principal suivant :

```
# Main
naissance = lecture("naissances.txt")
n_total = total(naissance)
print(f"Il y a eu {n_total} naissances aux U.S. entre 2000 et 2014")
pourcentage = stat(naissance,"mardi",n_total)
print(round(pourcentage,1)," % des naissances, c'était le mardi")
```

A l'exécution il doit donner dans le shell :

```
>>> (executing file "exercice_4.py")
Il y a eu 62176233 naissances aux U.S. entre 2000 et 2014
16.5 % des naissances, c'était le mardi
```

Ce code sera enregistré sous le nom *exercice\_4.py* .

```
# Fonctions
def lecture(nom) :
    """
    Lit le fichier dont le nom est en argument.
    Retourne une liste double
    """
    f = open(nom,"r",encoding='utf8')
    lst = []
    c = f.readline()
    c = f.readline()
    while c != "" :
        c = c[:-1]
        l = c.split(",")
        lst.append(l)
        c = f.readline()
    f.close()
    return lst
```

*Corrigé possible*

```
def total(l) :  
    """  
    | Renvoie le nombre total de naissance déclarés dans  
    | le fichier naissance.txt  
    | ...  
    tot = 0  
    for sl in l :  
    |     tot = tot + int(sl[-1])  
    return tot
```

*Corrigé possible*

```
def stat(l,jour,total) :  
    """  
    | Renvoie le pourcentage de naissances ayant eu lieu  
    | sur un jour donné de la semaine, par rapport au nombre  
    | total de naissances.  
    | ...  
    s = ["lundi", "mardi", "mercredi", "jeudi", "vendredi", "samedi", "dimanche"]  
    n = 0  
    for sl in l :  
    |     i = int(sl[3])  
    |     if s[i-1] == jour :  
    |         n = n + int(sl[-1])  
    return n*100/total
```

*Corrigé possible*

```
# Main  
naissance = lecture("naissances.txt")  
n_total = total(naissance)  
print(f"Il y a eu {n_total} naissances aux U.S. entre 2000 et 2014")  
pourcentage = stat(naissance, "mardi", n_total)  
print(round(pourcentage,1), " % des naissances, c'était le mardi")
```