

# NSI 1ère - Tables en csv - Introduction

QK

January 16, 2019

## Tables en csv - Introduction

### CSV

Comma Separated Values, des données séparées d'une virgule.

C'est une format de données qu'on rencontre souvent. Son avantage est immédiat : il ne comporte que les données.

Les langages haut niveau et les tableurs le gèrent tous. Nous allons en manipuler dans Python.

### Documents

Outils de Python sur le csv : 1. `python csv` dans google, 2. Repérer la doc en français, 3. Choisir la bonne version de Python, 4. Adapter les exemples.

### Consignes

A l'aide de la liste `2NDE14.csv`, créer un script qui réalise les étapes suivantes :

1. ouvrir le fichier, lister les noms
2. ajouter les noms (et les noms seulement !) de tous les élèves à une liste propre (pas les notes ni les titres de colonnes) : `listeclean`
3. créer une fonction `correction(n)` qui :
  1. Tire au sort  $n$  élèves **différents** de la liste.
  2. Affiche les noms de ces élèves choisis,
  3. Crée un fichier **notes.csv** avec `' , '` comme séparateur contenant les noms de tous les élèves de la classe et la note 0 pour les élèves choisis, la note 20 pour ceux qui n'ont pas été choisis.

4. Votre fichier doit se présenter ainsi :  
nom,note  
Camille CLAUDEL,20  
Albert DUPONTEL,0  
David HOCKNEY,20  
Bernard LAVILLIER,20  
Edgard Allan POE,0 ...
5. Votre fonction doit afficher une erreur si le nombre  $n$  demandé dépasse l'effectif de la classe ou si  $n$  n'est pas un entier.
6. Elle est appelée ainsi : `correction(5)`.

Tous les caractères accentués de la langue française doivent être gérés sans problème.

J'ai mis plus de temps à rédiger les consignes que la correction, cela devrait vous prendre moins d'une heure.

## Correction

```
import csv
import random

# on crée les listes vides.
listecsv = [] # les données brutes
listeclean = [] # les données après traitement

with open('tstmg2-liste.csv', newline='') as csvfile:
    nomeleve = csv.reader(csvfile, delimiter=';')
    for row in nomeleve:
        # print('{}'.format(row))
        listecsv.append(row)

# la première et dernière ligne du fichier sont inutiles.
# on se débarrasse aussi des ; encombrants
for eleve in listecsv[1:-1]:
    # print(eleve[0])
    listeclean.append(eleve[0])
# print listeclean

def correction(n):
    # les messages d'erreur
    if type(n) != int:
        print("n doit être un entier")
```

```

elif n > len(listeclean):
    print("n ne doit pas dépasser l'effectif de la classe")
else:
    echantillon = random.sample(listeclean, n)

    for eleve in echantillon:
        print(eleve)

    with open('notes.csv', 'w', newline='') as csvfile:
        csvwriter = csv.writer(csvfile, delimiter=',', quoting=csv.QUOTE_MINIMAL)
        csvwriter.writerow(["nom", "note"])
        for eleve in listeclean:
            csvwriter.writerow([eleve, 0 if eleve in echantillon else 20])

correction(3.14)
correction(121)
correction(5)

```