ACADÉMIE
DE NANTES
Liberté
Égalité
Fraternité

Manipulation de fichiers CSV et Python

Table des matières

1. Le format CSV 2
Création d'un fichier csv (Méthode 1) 2
Création d'un fichier csv (Méthode 2) 3
Remarques importantes sur le type des données 3
2. Prise en main : extraction des données dans Python 4
a. Ouverture d'un fichier CSV 4
Bilan : 4
b. Extraction d'une colonne 4
c. Création d'un fichier à partir de la liste tableau 5
d. Ajout d'une ligne 5
3. A vous de jouer 6
4. Pour commencer votre carrière de Data Scientist 6
Correction de l'activité 7

Une des utilisations principales de l'informatique est le traitement de quantités importantes de données dans des domaines très variés. Un site de commerce en ligne peut avoir à gérer des bases données pour des dizaines de milliers (voire plus) d'articles en vente, de clients, de commandes. Un hôpital doit pouvoir accéder efficacement à tous les détails de traitements de ses patients, etc.
En France, la loi reconnaît le droit d'obtenir la communication des documents détenus par une administration, quels que soient leur forme ou leur support. C'est ce que l'on appelle l'OpenData (données ouvertes).
Ainsi, de nombreux sites mettent à la disposition des citoyens des données. En voici quelques exemples.

https://www.data.gouv.fr/fr/
http://www.geonames.org/
http://www.opendatafrance.net/
https://data.nantesmetropole.fr/pages/home/

Les logiciels de traitement de base de données sont des programmes hautement spécialisés. Pour effectuer ce genre de tâches le plus efficacement possible, il est facile de mettre en œuvre les opérations de base dans un langage de programmation comme Python. Dans l'activité suivante, nous allons en illustrer quelques-unes.

Les fichiers mis à disposition dans le cadre de l'OpenData sont de différents formats (voir les liens ci-dessus). On trouve essentiellement le format csv et json. Dans la suite on s'intéressera au format csv.

1. Le format CSV

Le format CSV (pour *Comma Separated Values*, soit en français *valeurs séparées par des virgules*) est un format très pratique pour représenter des données structurées. Dans ce format, chaque ligne représente un enregistrement et, sur une même ligne, les différents champs de l'enregistrement sont séparés par une virgule (d'où le nom). En pratique, on peut spécifier le caractère utilisé pour séparer les différents champs. On utilise fréquemment un point-virgule, une tabulation ou deux points. Notons enfin que la première ligne d'un tel fichier est souvent utilisée pour indiquer le nom des différents champs.

Python permet d'extraire et de traiter des données contenues dans un fichier CSV et donc dans un fichier tableur (Excel ou Libre Office) . Dans ce qui suit, nous utiliserons plutôt Libre Office, car ce logiciel est libre et gratuit.

L'objectif de cette activité est de comprendre **comment avec des fichiers CSV, très simples, on peut extraire et traiter des fichiers CSV plus volumineux** comme ceux que l'on peut obtenir en se rendant sur les sites dont les liens figurent ci-dessus.
Pour construire un fichier csv, il y a deux méthodes qui vont être présentées dans la suite.

Création d'un fichier csv (Méthode 1)

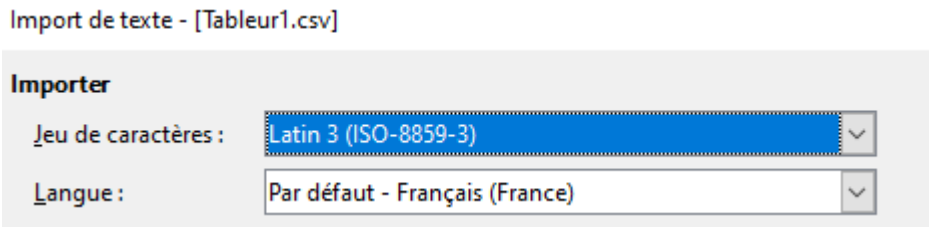
Dans un fichier tableur Libre Office, créez le tableau de notes suivant, puis enregistrez le fichier sous le format csv en le nommant Tableur1.csv. Pour cela, il suffit de faire Fichier/Enregistrer sous/ puis de choisir le type csv et de le nommer Tableur1.

Noms	Maths	Français	Histoire
Achille	12	14	15
Bertille	17	11	9
Carole	15	15	16
Damien	11	13	12
Eric	19	15	18
Fabrice	14	17	17

Après avoir enregistré ce fichier dans un dossier que vous pourrez nommer, par exemple, *TraitementDonnees*, fermez-le puis ouvrez-le. Vous constaterez que le C cédille (ç) du mot "Français" ne s'affiche pas correctement.

	Standard
1	Noms,Maths,Fran  ais,Histoire
2	Achille,12,14,15
3	Bertille,17,11,9
4	Carole,15,15,16
5	Damien,11,13,12
6	Eric,19,15,18
7	Fabrice,14,17,17

Si c'est le cas, modifiez le *Jeu de caractères* de manière à ce qu'il soit réglé sur : *Latin 3 (ISO-8859-3)*.



Après avoir enregistré, toujours au format csv, puis fermé le fichier, l'ouverture suivante permettra d'obtenir :

	Standard
1	Noms,Maths,Français,Histoire
2	Achille,12,14,15
3	Bertille,17,11,9
4	Carole,15,15,16
5	Damien,11,13,12
6	Eric,19,15,18
7	Fabrice,14,17,17

Dans la suite, nous ne toucherons plus au fichier *Tableur1.csv*.

Création d'un fichier csv (Méthode 2)

On ouvre un Bloc-Note dans lequel on copie les lignes suivantes.

```
Noms,Maths,Français,Histoire
Achille,12,14,15
Bertille,17,11,9
Carole,15,15,16
Damien,11,13,12
Eric,19,15,18
Fabrice,14,17,17
```

On enregistre le fichier en le nommant *Tableur1.csv*. À nouveau, un réglage du jeu de caractères sera peut-être nécessaire.

Remarques importantes sur le type des données

On remarque tout d'abord que les valeurs contenues dans ce tableau sont entières. Lorsqu'elles seront décimales, il faudra peut-être utiliser la fonction Python `replace(',','.')` qui permet de remplacer une virgule par un point. En effet, le séparateur décimal dans un tableur est la virgule alors que c'est le point en Python.

Attention, si le tableur considère que les notes des élèves sont des entiers, une fois extraites, Python les considère comme des chaînes de caractères. Il faudra donc, pour pouvoir les traiter, utiliser la fonction *`int(…)`* de Python. Mais cette utilisation sera précisée lorsqu'elle sera nécessaire dans la suite de cette activité.

Dans ce qui suit, nous allons récupérer, ces données avec Python, les traiter puis créer un nouveau fichier dans lequel on aura ajouté une ligne avec un nouvel élève, une ligne dans laquelle on aura la moyenne des matières et une colonne dans laquelle on aura les moyennes des élèves.
Pour plus de renseignements sur le la manipulation des fichiers csv :
https://docs.python.org/3/library/csv.html

2. Prise en main : extraction des données dans Python

a. Ouverture d'un fichier CSV

Créez un fichier Python contenant le code suivant. *On prendra bien soin d'enregistrer le fichier dans le même dossier que le fichier Tableur1.csv..*

```
#ouverture d'un fichier CSV...
#... création de la liste des lignes nommée tableau...
#... et affichage des lignes.
import csv
with open('Tableur1.csv',newline='') as f:  #Ouverture du fichier CSV
    tableau=[]
    lire=csv.reader(f)                       #chargement des lignes du fichier csv
    print('',end='\n')
    print('Affichage des lignes du tableau',end='\n')
    for ligne in lire:                  #Pour chaque ligne...
        print(ligne, end='\n')                #...affichage de la ligne dans la console ...
        tableau.append(ligne)                 #...on ajoute la ligne dans la liste ...
                                              #...de liste nommée tableau
print(tableau)                          # Affichage du tableau
print(tableau[1])                       # Affichage de la deuxième ligne
print(type(tableau[1]))                 # Type de la variable tableau[1]
print(tableau[1][2])                    # Affichage de la variable tableau[1][2]
print(type(tableau[1][2]))              # Type de la variable tableau[1][2]
```

Lorsqu'on lance le script, on obtient dans la console des affichages dont voici quelques explications.

L'instruction `print(tableau)` permet d'afficher la variable `tableau` qui contient toutes les lignes du fichier Tableur1.csv. Plus précisément, il s'agit d'un tableau de tableaux qui contiennent les lignes du fichier Tableur1.

L'instruction `print(tableau[1])` permet d'afficher le tableau de la deuxième ligne du fichier csv.

`[Achille,12,14,15]`

En effet, les indices du tableau sont numérotés à partir de 0. L'indice 1 permet donc d'obtenir la deuxième valeur de `tableau` c'est-à-dire la deuxième ligne du fichier Tableur1.csv.

De même, l'instruction `tableau[1][2]` permet d'obtenir la troisième valeur de la deuxième valeur de `tableau`. Cela correspond à la note `14` qui est la note de Français d'Achille.

L'instruction `type(tableau[1][2])` confirme que `tableau[1][2]` est de type chaîne de caractère (string) et que pour faire des calculs avec cette variable, il faudra la convertir en type entier (`int`) à l'aide de la fonction Python `int(…)`.

Bilan :

1. pour extraire la ii^{ème} ligne de notre tableau de données et mettre ses valeurs dans une liste Python que l'on peut nommer par exemple lignei, il faut écrire l'instruction suivante.

`lignei=tableau[i-1]`

2. Pour extraire la valeur `x` se trouvant dans la i^{ième} ligne et la j^{ième} colonne du tableau, il suffit d'écrire l'instruction suivante.

`x=tableau[i-1][j-1]`

b. Extraction d'une colonne

On a vu comment extraire une ligne de notre tableau de données et mettre son contenu dans une liste Python. Nous avons vu comment extraire une donnée et mettre son contenu dans une variable Python (de type chaîne de caractère). On souhaite maintenant extraire les données d'une colonne du tableau et les mettre dans une liste Python.

Exemple : Le code suivant permet d'obtenir toutes les notes d'histoire.

```
NoteHistoire=[]                              # Création d'une liste vide
n=len(tableau)                               # n est le nombre de ligne de tableau
for k in range(len(tableau)):                # pour chaque ligne de tableau :
      NoteHistoire.append(tableau[k][3])           # on ajoute dans la liste NoteHistoire
                                                   # la 4ième valeur de la ligne
print(NoteHistoire)                          # affichage dans la console de la liste des
                                             #  notes d'histoire
```

On obtient dans la console, l'affichage suivant.

```
[Histoire,15, 9, 16, 12, 18, 17]
```

Bilan :
Pour mettre dans une liste Python la $j^{ième}$ colonne du tableau de données à partir de la $i^{ième}$ ligne, on peut utiliser le code suivant.

```
colonne=[]
for k in range(i-1,len(tableau)):
      colonne.append(tableau[k][j-1])

print(colonne)
```

c. Création d'un fichier à partir de la liste tableau

Créez un deuxième fichier Python contenant le code précédent et le suivant. *On prendra bien soin d'enregistrer ce nouveau fichier dans le même dossier que le fichier Tableur1.csv.* Vérifiez qu'un fichier *Tableur2.csv* a été créé dans votre dossier.

```
#Création d'un fichier à partir de la liste tableau
with open('Tableur2.csv','w',newline='') as f:  #Ouverture du fichier CSV en écriture
    ecrire=csv.writer(f)                        # préparation à l'écriture
    for i in tableau:                           # Pour chaque ligne du tableau...
        ecrire.writerow(i)          # Mettre dans la variable ecrire cette nouvelle ligne
print('',end='\n')
print('longueur du tableau : ',len(tableau))
```

Remarque : le deuxième argument `'w'` passé dans la fonction open(…,…,…) signifie que l'on va écrire dans le fichier `Tableur2.csv`.

`'w'` pour **write** (écrire en anglais)

d. Ajout d'une ligne

Créez un troisième fichier Python contenant le code précédent et le suivant. *On prendra bien soin d'enregistrer ce nouveau fichier dans la même dossier que le fichier Tableur1.csv.*. Vérifiez qu'un fichier Tableur3.csv a été créé dans votre dossier et qu'il contient une nouvelle ligne.

```
#Ajout d'une ligne
with open('Tableur3.csv','a',newline='') as f:   #Ajout d'une ligne dans le fichier csv
    ecrire=csv.writer(f)                          # préparation à l'écriture
    ecrire.writerow(['Gilles','15','13','9'])   # Mettre dans ecrire cette nouvelle ligne
```

Remarque : le deuxième argument `'a'` passé dans la fonction open(…,…,…) signifie que l'on va ajouter quelque chose dans le fichier `Tableur3.csv`.

`'a'` pour **add** (ajouter en anglais)

3. A vous de jouer

On souhaite coder le programme Python qui permet de créer un fichier csv que l'on appellera *nomPrenom_Tableau4.csv* dans lequel auront été ajoutés par rapport au fichier Tableur1.csv :
- la ligne des moyennes des différentes matières ;
- La colonne des moyennes des différents élèves.

Pour cela, on pourra utiliser la fonction Python *moyenne* qui prend en argument une liste python `lst` et qui renvoie la moyenne des valeurs contenues dans la liste `lst`. On utilisera les codes présentés dans la partie 2 de ce document.

```
def moyenne(lst):
    somme=0
    effectif=0
    for element in lst:
        somme=somme+element
        effectif=effectif+1
    moyenne=somme/effectif
    return moyenne
```

4. Pour commencer votre carrière de Data Scientist

Le site institutionnel français **data.gouv.fr** est la plate-forme ouverte des données publiques françaises. Elle permet de télécharger de très nombreuses données structurées et de les traiter ensuite.

Téléchargez sur le site **data.gouv.fr** un fichier de données (au format CSV) de votre choix. Présentez-le et expliquez quels en sont les objets et les descripteurs. Manipulez les données afin d'en extraire de l'information.

Commencez par un exemple d'exploitation basique et ensuite poursuivez avec un exemple d'exploitation plus fin de ces données.

Correction de l'activité

On souhaite coder le programme Python qui permet de créer un fichier csv que l'on appellera *nomPrenom_Tableau4.csv* dans lequel auront été ajoutés par rapport au fichier Tableur1.csv :
- la ligne des moyennes des différentes matières ;
- La colonne des moyennes des différents élèves.

Pour cela, on réutilisera la fonction Python *moyenne*.

On commence par importer le module CSV.

```
import csv
```

Nous avons besoin de la fonction Python qui permet de calculer la moyenne des valeurs contenues dans une liste. Nous l'avions créée dans l'exercice 6 de l'activité « Parcours séquentiel d'une liste ou d'un tableau ».

Voici cette fonction.

```
def moyenne(tableau):
    somme=0
    effectif=0
    for element in tableau:
        somme=somme+element
        effectif=effectif+1
    moyenne=somme/effectif
    return moyenne
```

Mais attention, si on reprend le premier code et si on ajoute à la fin, la ligne :

```
print(type(tableau[1][1]))
```

On constate que les valeurs contenues dans la liste de liste, `tableau` sont de type string.
Pour pouvoir calculer les moyennes des notes, il faut, tout d'abord, convertir ces string en entier (car dans notre cas, il s'agit de nombres entiers). On modifie donc le code de la fonction ci-dessous de la manière suivante.

```
def moyenne(tableau):
    somme=0
    effectif=0
    for element in tableau:
        somme=somme+int(element)
        effectif=effectif+1
    moyenne=somme/effectif
    return moyenne
```

On copie, à la suite, le code qui permet de récupérer les données contenues dans le fichier `Tableur1.csv`.

```
with open('Tableur1.csv',newline='') as f:      #Ouverture du fichier CSV
    tableau=[]
    lire=csv.reader(f)              #chargement des lignes du fichier csv
    for ligne in lire:           #Pour chaque ligne...
        tableau.append(ligne)     #...on ajoute la ligne dans la liste ...
                                  #...de liste nommée tableau
```

Pour chaque élève, on calcule sa moyenne et on l'ajoute à la liste de ses notes

```
for i in range(1,len(tableau)):       # Ligne concernant un élève i
     tab=[]                              # tableau des notes de l'élève i
     for j in range(1,len(tableau[0])):  # Pour chaque note de l'élève i
          tab.append(tableau[i][j])         # on ajoute ses notes dans tab
          moy=moyenne(tab)                  # on calcule moy dans tab
     tableau[i].append(moy)              # On ajoute la moyenne de l'élève
                                         # i en bout de ligne
tableau[0].append('Moyenne')          # On ajoute le mot 'Moyenne' au
                                      # bout de la première ligne
```

Création des moyennes des différentes matières.
Remarque : On peut aller plus vite en utilisant une double structure Pour.

```
tab_math=[]
for i in range(1,len(tableau)):
    tab_math.append(tableau[i][1])
print("La moyenne de math est : ",moyenne(tab_math))

tab_fran=[]
for i in range(1,len(tableau)):
    tab_fran.append(tableau[i][2])
print("La moyenne de math est : ",moyenne(tab_fran))

tab_hg=[]
for i in range(1,len(tableau)):
    tab_hg.append(tableau[i][3])
print("La moyenne de math est : ",moyenne(tab_hg))
```

Ajout dans la variable tableau de la liste des moyennes

```
tableau.append(['Moyenne des
matières',moyenne(tab_math),moyenne(tab_fran),moyenne(tab_hg)])
```

Dans la version suivante, on utilise une double boucle pour et on prévoit l'ajout de nouvelles matières

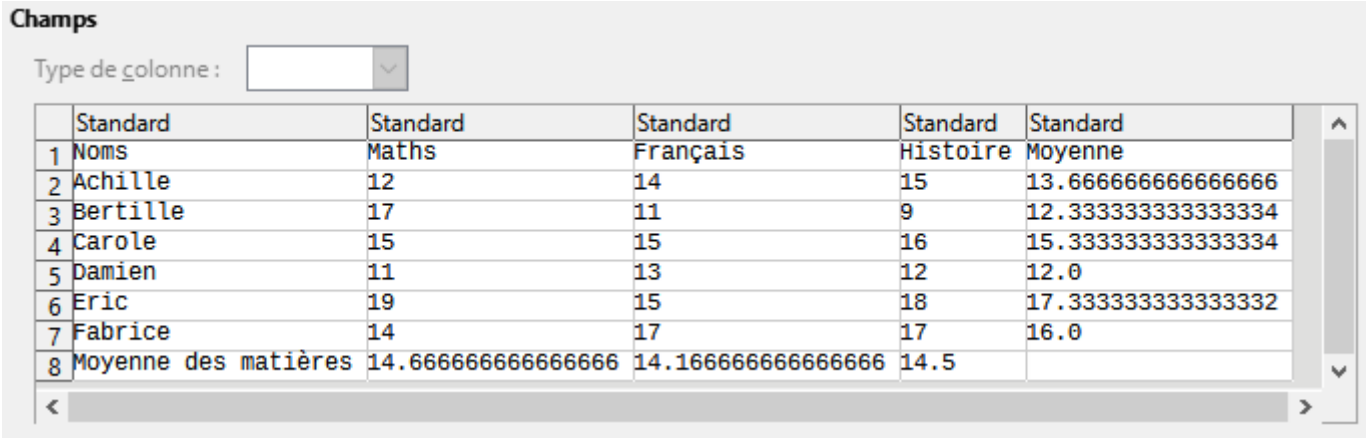
```
tabMoyMatiere=['Moyenne par matière']          # Création d'un tableau de
                                               # moyennes des matières

for j in range(1,len(tableau[0])):             # Pour chaque matière j
    tabNoteMatiere=[]                            # On stocke les notes de la
    for i in range(1,len(tableau)):              # matière j
        tabNoteMatiere.append(tableau[i][j])     # dans un tableau
    moyMatiere=moyenne(tabNoteMatiere)           # on calcule la moyenne de
                                                 # la matière j
    tabMoyMatiere.append(moyMatiere)             # on ajoute cette moyenne
                                                 # dans tabMoyMatiere
tableau.append(tabMoyMatiere)                  # On ajoute la ligne des
                                               # moyennes des matières dans
                                               # tableau
```

Création d'un fichier à partir de la liste tableau

```
with open('Tableur4.csv','w',newline='') as f:  #Ouverture du fichier CSV
                                                # ... en écriture
    ecrire=csv.writer(f)                        # préparation à l'écriture
    for i in tableau:        # Pour chaque ligne du tableau...
        ecrire.writerow(i)       # Mettre dans la variable ecrire …
                                 # cette nouvelle ligne
```

On a créé le fichier `Tableur4.csv` qui contient la colonne des moyennes des élèves et la ligne des moyennes des matières.



	Standard	Standard	Standard	Standard	Standard
1	Noms	Maths	Français	Histoire	Moyenne
2	Achille	12	14	15	13.666666666666666
3	Bertille	17	11	9	12.333333333333334
4	Carole	15	15	16	15.333333333333334
5	Damien	11	13	12	12.0
6	Eric	19	15	18	17.333333333333332
7	Fabrice	14	17	17	16.0
8	Moyenne des matières	14.666666666666666	14.166666666666666	14.5	

	A	B	C	D	E
1	Noms	Maths	Français	Histoire	Moyenne
2	Achille	12	14	15	13.666666666666666
3	Bertille	17	11	9	12.333333333333334
4	Carole	15	15	16	15.333333333333334
5	Damien	11	13	12	12.0
6	Eric	19	15	18	17.333333333333332
7	Fabrice	14	17	17	16.0
8	Moyenne des matières	14.666666666666666	14.166666666666666	14.5	