

Résumé

Données en table et CSV

CSV

CSV pour *Comma Separated Values*, des données séparées par des virgules, désigne un format d'encodage de documents.

Il permet d'enregistrer un tableau à 2 dimensions dans un fichier texte très simples.

- La première ligne indique généralement les *champs*,
- Un même *séparateur* (par défaut , mais parfois ;) sépare les éléments entre eux,
- Chaque ligne est un *enregistrement*.

Exemple : `employees.csv`

```
Name,Branch,Year,CGPA
Nikhil,COE,2,9.0
Sanchit,COE,2,9.1
Aditya,IT,2,9.3
```

Représente

Name	Branch	Year	CGPA
Nikhil	COE	2	9.0
Sanchit	COE	2	9.1
Aditya	IT	2	9.3

Usage courant

Les tableurs permettent d'importer ce genre de documents et de les manipuler.

Lorsque l'importation échoue (souvent à cause du séparateur), on obtient un fichier quasi brut, inutilisable.

Exemple avec Python

Python propose la librairie CSV qui est trop riche pour être présentée entièrement.

Voici un exemple **dans lequel le séparateur est ,**
À adapter avec soin.

```

import csv

def charger(fichier: str) -> list:
    """Charge un fichier csv, renvoie une list de dict"""
    with open(fichier, encoding="UTF-8") as f:
        reader = csv.DictReader(f)
        lines = [dict(line) for line in reader]
    return lines

def selectionner(contenu: list, champ: str, valeur: str):
    """Enregistrements de contenu pour lesquels champ=valeur"""
    return [ligne for ligne in contenu if ligne[champ] == valeur]

def projeter(contenu: list, champ: str) -> list:
    """Valeurs de 'champ' pour chaque enregistrement de contenu"""
    return [ligne[champ] for ligne in contenu]

def sauvegarder(fichier: str, contenu: list, champs: list):
    """Sauvegarde un fichier au format csv. Les fieldnames doivent correspondre"""
    with open(fichier, "w", encoding="UTF-8") as f:
        writer = csv.DictWriter(f, fieldnames=champs)
        writer.writeheader()
        writer.writerows(contenu)

```

Charger, sélectionner, projeter

```

# Les employés de Calcutta
fichier1 = "employees.csv"
contenu1 = charger(fichier1)
print(contenu1)
coe = selectionner(contenu1, "Branch", "COE")
print(coe)
names = projeter(contenu1, "Name")
print(names)

```

Produit l'affichage

```

[{'Name': 'Nikhil', 'Branch': 'COE', 'Year': '2', 'CGPA': '9.0'},
 {'Name': 'Sanchit', 'Branch': 'COE', 'Year': '2', 'CGPA': '9.1'},
 {'Name': 'Aditya', 'Branch': 'IT', 'Year': '2', 'CGPA': '9.3'}]
[{'Name': 'Nikhil', 'Branch': 'COE', 'Year': '2', 'CGPA': '9.0'},
 {'Name': 'Sanchit', 'Branch': 'COE', 'Year': '2', 'CGPA': '9.1'}]
['Nikhil', 'Sanchit', 'Aditya']

```

Remarquez que toutes les valeurs (même les 'Year' et 'CGPA') sont des `str` !

Fusionner

```
# Les employés de Paris
fichier2 = "employees2.csv"
contenu2 = charger(fichier2)

# On fusionne tous les employés
# exactement les mêmes champs dans les 2 contenus !
contenu_fusion = contenu1 + contenu2
```

Sauvegarder un nouveau fichier csv

```
# Penser à écrire les champs à la main
champs = ["Name", "Branch", "Year", "CGPA"]

fichier3 = "employees_fusion.csv"
sauvegarder(fichier3, contenu_fusion, champs)

# La "branch" IT après la fusion
it_fusion = selectionner(contenu_fusion, "Branch", "IT")
print(it_fusion)
fichier4 = "it_fusion.csv"
sauvegarder(fichier4, it_fusion, champs)
```

Produit l'affichage

```
[{'Name': 'Aditya', 'Branch': 'IT', 'Year': '2', 'CGPA': '9.3'},
{'Name': 'Jose', 'Branch': 'IT', 'Year': '2', 'CGPA': '9.4'},
{'Name': 'Hamid', 'Branch': 'IT', 'Year': '4', 'CGPA': '9.5'}]
```

Faire des calculs

```
# ATTENTION, pour faire des calculs, il faut penser à convertir en nombre !!!
# anciennete_moyenne = sum(projeter(contenu_fusion, "Year")) / len(contenu_fusion)
# Provoque une erreur "TypeError", les "Year" ont pour valeur des `str` !!!
ages = [int(age) for age in projeter(contenu_fusion, "Year")]
print(ages) # des int
anciennete_moyenne = sum(ages) / len(ages)
print(anciennete_moyenne)
```

Produit l'affichage

```
[2, 2, 2, 3, 2, 4]
2.5
```

Fichiers utilisés et produits

- script : employees.py
- employés de Calcutta : employees.csv
- employés de Paris : employees2.csv

Après la fusion, on produit ce nouveau CSV

- employés après la fusion : employees_fusion.csv
- employés IT après la fusion : employees_it.csv