**NSI**

06

08

**Automne**



TRAITEMENT DES DONNÉS EN TABLES (2)

[Stéphane beaudet – Frédéric PEURIERE]

*Savoir lire et traiter les données d´un fichier CSV*

*Mises en pratique de l´ensemble de nos connaissance : traitement de données sismiques*

Pour cette activité, créez un répertoire CSV dans lequel nous placerons tous les fichiers en téléchargement sur le site.

**LECTURE ET TRAITEMENT DE FICHIERS CSV:**

**Rappels sur les types construits de données:** Commençons avec le fichier *rappels.py*  qui définit trois types construits de données:

1. eleve1= {'prenom':'Marta','classe':'1B','moyenne': 17,'spe1':'NSI','spe2':'Arts','spe3':'SVT'}
2. eleve2= {'prenom':'Jean','classe':'1D','moyenne': 14,'spe1':'NSI','spe2':'Maths','spe3':'Physique'}
3. tab = [eleve1, eleve2]

Exécutez ce script dans THONNY puis affichez le contenu des trois variables dans la console:

1) ***eleve1*** et ***eleve2*** sont des: ☐ tuples ☐ flottants ☐ dictionnaires ☐ listes ☐ caractères

2) ***tab***est un(e): ☐ tuple de liste ☐ liste de dictionnaires ☐ dictionnaire de tuples

|  |
| --- |
| Testez les instructions suivantes dans la console, observez et analysez les valeurs retournées  >>> eleve1 >>> eleve2 >>> tab  >>> **len**(tab) >>> eleve1**['classe']** >>> eleve2**['spe2']** >>> tab[1]>>> **for** i **in** eleve1.values():  **print** (i)  >>> **for** i **in** tab:  **print** (i)  >>> **for** i,j **in** enumerate(donnees):  **print**(i,"---->",j) |

3) Cette dernière instruction nous sera très utile pour la suite. Après l´avoir testée dans la console, expliquez comment fonctionne cette boucle et ce que représentent les variables i et j.

**Fichiers csv:** Continuons avec les fichiers *eleves.py* et eleves.csv

Le fichier eleves.csv contient une liste complète d´élèves de première qui suivent la spécialité NSI.

4) Quelle est la signification de CSV (faites une recherche)?

5) Ouvrez le fichier dans le *bloc note* de Windows. Comment les données sont-elles stockées dans un fichier csv? Ouvrez ensuite ce même fichier dans le tableau d´Office et observez comment les données sont représentées.

Avec Python nous allons maintenant traiter et filtrer ces données, ouvrez le fichier *eleves.py*  dans THONNY.

6) Après exécution du programme, affichez la valeur de la variable **donnees** dans la console. Sous quelle forme les données sont-elles stockées par Python?

7) Quelle instruction faut-il écrire pour afficher le nombre d´élèves de NSI?

Dans la partie qui suit, nous allons utiliser la **boucle for** et la fonction **enumerate** pour stocker dans la liste **tab**, les valeurs désirées:

8) Ecrivez sur votre feuille de réponses puis testez les instructions qui vont permettre d´afficher:

**- le prénom des élèves de la classe de première A.  
- le prénom des élèves de la classe de première B qui suivent la spécialité Arts.  
- le prénom des élèves qui suivent Maths et Physique en plus de NSI.  
- le prénom des élèves dont la moyenne est inférieure à 10** (les valeurs numériques doivent être converties en entiers).  
**- la moyenne des élèves de la spécialité NSI.**

**TRAITEMENT DE données sismiques:**

Le fichier *sismicite.csv*, téléchargé sur le site <http://www.edusismo.org> stocke tous les séismes survenus depuis le début de l´année 2020. Comme dans l´activité précédente, nous traiterons et filtrerons les données avec le fichier *sismicite.py*.

**Le but de l´activité est d´afficher à l´aide du module *folium,* l´emplacement de tous les séismes survenus depuis le début de l´année 2020 sur une carte.  
  
Nous en créerons ensuite une deuxième ne représentant que les séismes de magnitude supérieure à 4.**

UTILISATION DU MODULE FOLIUM:  
  
Création d´une carte et affectation à la variable ***carte***. location: on indique la latitude et la longitude du centre de la carte (ici, Lisbonne). zoom\_start: plus la valeur est grande, plus le zoom sur le centre de la carte est important:

**carte= folium.Map(location=[38.7222524, -9.1393366],zoom\_start=3)**

Création d´un repère sur la carte: On indique la latitude et la longitude du séisme. popup: affiche un commentaire lorsqu´on clique sur le repère (la magnitude du séisme par exemple). Pour placer tous les repères, on pourra utiliser une *boucle for*:

**folium.Marker([46.078637266899, 6.4111924884134],popup="commentaire").add\_to(carte)**

On sauvegarde la carte au format HTML, le fichier est créé dans le répertoire CSV. On clique sur le fichier, la carte s´affiche dans le navigateur:

**carte.save('maCarte.html')**