TRAITEMENT DE DONNÉES EN TABLES

I) Qu'est-ce qu'une « table » ?

Ex : Les personnages de la saga starwars (d'après www.kaggle.com/datasets/jsphyg/star-wars)

id_char	name	height	mass	hair_color	skin_color	eye_color	birth_year	gender	planet	species
1	Luke Skywalker	172	77	blond	fair	blue	19BBY	male	59	Human
2	C-3PO	167	75		gold	yellow	112BBY		59	Droid
3	R2-D2	96	32		white, blue	red	33BBY		7	Droid
4	Darth Vader	202	136	none	white	yellow	41.9BBY	male	59	Human
5	Leia Organa	150	49	brown	light	brown	19BBY	female	1	Human
6	Anakin Skywalker	188	84	blond	fair	blue	41.9BBY	male	59	Human
7	Obi-Wan Kenobi	182	77	auburn, white	fair	blue-gray	57BBY	male	19	Human

Remarques :

- En informatique, les données sont souvent organisées en grands tableaux appelés **« tables »**.
- Chaque colonne d'une table correspond à une des informations recueillies. Ces informations sont appelées **« descripteurs »** ou **« attributs »**.
- Chaque ligne d'une table est appelée **« enregistrement »**.
- Dans l'exemple ci-dessus, chaque personnage est unique il y a un unique enregistrement par personnage.
  Or plusieurs personnages peuvent avoir la même taille, même couleur de cheveux ou date de naissance. Il est donc important qu'au moins une des colonnes n'ait pas de **« doublons »** de façon à pouvoir distinguer de façon unique les personnages. Le champ « name » ne peut pas forcément tenir ce rôle car même si nous ne connaissons qu'un seul Luke Skywalker, peut-être y en a-t-il plusieurs dans la galaxie ?
  Pour résoudre cette difficulté, on ajoute souvent une colonne (nommée ici « id_char ») associant à chaque enregistrement un **« identifiant unique »**.

II) Les fichiers « CSV »

Parmi les formats de fichiers permettant de stocker un tableau de données, le format CSV (Comma-Separated Values), est un format texte très répandu.

Exemple : Les personnages de la saga starwars au format csv

```
id_char,name,height,mass,hair_color,skin_color,eye_color,birth_year,gender,planet,species
1,Luke Skywalker,172,77,blond,fair,blue,19BBY,male,59,Human
2,C-3PO,167,75,,gold,yellow,112BBY,,59,Droid
3,R2-D2,96,32,,"white, blue",red,33BBY,,7,Droid
4,Darth Vader,202,136,none,white,yellow,41.9BBY,male,59,Human
5,Leia Organa,150,49,brown,light,brown,19BBY,female,1,Human
6,Anakin Skywalker,188,84,blond,fair,blue,41.9BBY,male,59,Human
7,Obi-Wan Kenobi,182,77,"auburn, white",fair,blue-gray,57BBY,male,19,Human
```

- Chaque ligne du texte correspond à une ligne du tableau précédent et des virgules permettent de séparer le contenu des colonnes.
- La première ligne permet en général de préciser les noms des descripteurs.
- Si une cellule du tableau est vide, alors il y a deux virgules à la suite.
- Si une cellule contient une virgule, alors on « protège » le contenu de cette cellule en l'entourant de guillemets pour que cette virgule ne soit pas interprétée comme un séparateur entre deux colonnes.

Remarques :

La syntaxe des fichiers csv n’est pas complètement standardisée. Quand on ouvre un fichier csv avec un tableur, il est donc prudent de faire quelques vérifications :

- Est-ce que l'encodage des caractères a été reconnu : les accents et caractères spéciaux sont-ils bien retranscrits ?
- Quel est le séparateur entre les colonnes ? Normalement c'est la virgule, mais dans les documents francophones, on choisit souvent le point-virgule car la virgule est déjà utilisée comme séparateur décimal.
- Quel est le séparateur décimal : point ou virgule ? Les nombres décimaux sont-ils bien reconnus ?

III) Avec un tableur

Quand les traitements que l'on souhaite faire sur les données sont simples, un tableur suffit souvent (Microsoft Excel, Libre Office Calc, Google Sheet,...)
Même si ces tableurs ont leurs propres formats de fichiers (.xlsx, .ods), ils savent aussi lire les fichiers csv.

Remarque : Excel a souvent du mal à formater automatiquement les fichiers csv séparés par des virgules. Pour forcer le formatage manuel, choisir le menu « Données », puis « A partir du texte » pour afficher l'assistant d'importation.

A faire :

1) Ouvrir le fichier « characters.csv » avec un tableur. Vérifier qu'il est lu correctement : colonnes séparées au bon endroit, encodage.
2) Combien y a-t-il d'enregistrements en tout ?
3) En vous aidant des possibilités du tableur (tri, recherche, fonction =nb.si(),...), répondre aux questions suivantes :
   - Combien y a-t-il de personnages humains ?
   - Combien y a-t-il de droïdes aux yeux rouges ?
   - Combien y a-t-il de personnages de plus de 2 mètres ?
   - Créer dans un nouvel onglet une feuille de calcul contenant le nom et la couleur de peau des personnages dont la peau contient du marron. (On pourra utiliser =nb.si avec comme critère de recherche "*brown*")
   - Trier la feuille de calcul initiale par taille décroissante.
   - Retrier la feuille de calcul initiale par ordre alphabétique des noms.
   - Créer un nouvel onglet dans lequel on copiera le contenu de planet.csv, puis ajouter à l'onglet contenant characters.csv une colonne avec les noms des planètes d'origine. (On pourra utiliser =recherchev)

IV) Avec Python

Les fichiers csv étant des fichiers textes, ils sont faciles à lire avec Python.
Testons successivement les 3 méthodes ci-dessous pour lire un fichier csv :

1) Méthode 1 : Lecture du fichier csv en Python natif (sans utiliser le module csv)

```
with open("characters.csv") as fichiercsv:
    liste_de_lignes = fichiercsv.readlines()
liste_de_listes = []
for ligne in liste_de_lignes:
    liste_de_listes.append(ligne.strip().split(","))
```

Questions :

- A quoi sert le `.strip()` ?
- A quoi sert le `.split(",")` ?
- Que contient `liste_de_listes[0]` ?
- `liste_de_listes[3]` contient quel enregistrement du fichier ?
- Comment afficher juste la taille du droïde R2-D2 ?
- Est-ce que sa couleur de peau a été bien interprétée ?

2) Méthode 2 : Lecture du fichier csv avec la méthode "reader" du module csv

```
import csv
with open("characters.csv", newline='') as fichiercsv:
    liste_de_listes = list(csv.reader(fichiercsv))
```

Questions :

- De quel type est l'objet `csv.reader` ?
- A quoi sert le `list()` ?
- Que contient `liste_de_listes[0]` ?
- `liste_de_listes[3]` contient quel enregistrement du fichier ?
- Comment afficher juste la taille du droïde R2-D2 ?
- Est-ce que sa couleur de peau a été bien interprétée ?

3) Méthode 3 : Lecture du fichier csv avec la méthode "DictReader" du module csv

```
import csv
with open("characters.csv", newline='') as fichiercsv:
    liste_de_dictionnaires = list(csv.DictReader(fichiercsv))
```

Questions :

- Que contient `liste_de_listes[0]` ?
- `liste_de_listes[3]` contient quel enregistrement du fichier ?
- Comment afficher juste la taille du droïde R2-D2 ?
- Laquelle des 3 méthodes ci-dessus vous paraît préférable ?