Traitement de données en tables

Capacités attendues

- ✓ Importer une table depuis un fichier texte tabulé ou un fichier CSV.
- ✓ Rechercher les lignes d'une table vérifiant des critères exprimés en logique propositionnelle.
- ✓ Construire une nouvelle table en combinant les données de deux tables.

Définition

Un fichier CSV (*Comma-Separated Values*) est un fichier texte.
Le format CSV permet de représenter les informations contenues dans un tableau :

- chaque ligne du fichier correspond à une ligne du tableau ;
- les virgules (ou éventuellement les points-virgules ou tout autre caractère pertinent) indiquent les séparations entre colonnes.

La première ligne du fichier donne le nom des **descripteurs** de chacun des **champs**.

Exemple

Le tableau suivant présente la liste des premiers lauréats du prix Turing.

Année	Prénom	Nom	Nationalité
1966	Alan	Perlis	États-Unis
1967	Maurice	Wilkes	Royaume-Uni
1968	Richard	Hamming	États-Unis
1969	Marvin	Minsky	États-Unis
1970	James H.	Wilkinson	Royaume-Uni
1971	John	McCarthy	États-Unis
1972	Edsger	Dijkstra	Pays-Bas

Dans le format CSV, ce tableau est représenté par le texte (chaîne de caractères) ci-dessous.

```
Année,Prénom,Nom,Nationalité
1966,Alan,Perlis, États-Unis
1967,Maurice,Wilkes,Royaume-Uni
1968,Richard,Hamming,États-Unis
1969,Marvin,Minsky ,États-Unis
1970,James H.,Wilkinson,Royaume-Uni
1971,John,McCarthy ,États-Unis
1972,Edsger,Dijkstra,Pays-Bas
```

Travaux dirigés n° 1

Partie A

1. Aller chercher le fichier `indicateurs.csv` et le fichier `td1-e.py` sur notre site ;
2. Ouvrir ce fichier avec un tableur (*LibreOffice Calc* idéalement).
3. Aller dans `Données > Auto-filtres` et chercher les lycées de la ville de Champigny-sur-Marne.

Dans toute la suite, on répondra aux questions posées à l'aide de scripts écrits en Python. On pourra vérifier ses résultats à l'aide du tableur et de filtres.

Partie B

1. Ouvrir le fichier `td1-e.py` dans l'éditeur de code et compléter le programme : on veut faire en sorte de stocker chacune des lignes du fichier `indicateurs.csv` dans une liste.
2. Quel est le type des éléments contenus dans cette liste ? Quelle est sa longueur ?
3. Afficher la première ligne : elle contient le nom des descripteurs.
4. Afficher la ligne d'indice 8274 : que représente les informations qu'elle contient ?

Partie C

1. Écrire une fonction `champs` qui prend en argument une ligne et renvoie la liste des champs.
   On rappelle que les champs sont séparés par une virgule.
2. Modifier le script initial pour créer une liste `base` dont les éléments sont les listes des champs correspondant à chacune des lignes du fichier `indicateurs.csv`.

   En guise d'exemple, les instructions :

```
print(base[9819])
print(base[13210])
```

   doivent produire les affichages suivants :

```
['LYCEE LANGEVIN-WALLON (GENERAL ET TECHNO.)', '2016', 'CHAMPIGNY SUR MARNE', 'CRETEIL',
'PU', '', '76', '69', '', '74', '73']
['LYCEE LANGEVIN-WALLON (GENERAL ET TECHNO.)', '2017', 'CHAMPIGNY SUR MARNE', 'CRETEIL',
'PU', '', '83', '63', '', '70', '72']
```

Partie D

Concevoir des scripts et/ou fonctions Python qui permettent de répondre aux questions ci-dessous.

1. Quel est le taux de réussite du lycée Langevin-Wallon dans la série S en 2018 ?
2. (a) Combien de lycées ont obtenu un taux de 100% de réussite au bac S en 2017 ?

   (b) Parmi eux, combien étaient des lycées publics ?
3. Combien de lycées de l'académie de Créteil ont obtenu un taux de 100% de réussite au bac L en 2016 ?
4. Quels lycées publics de l'Académie de Créteil ont obtenu 100% de réussite toutes séries confondues en 2018 ? (Attention, si le taux n'est pas renseigné, c'est que le lycée ne propose pas cette série.)

Partie E

1. Quelle est le taux de réussite national au bac général en 2018 ?
2. Quel est le lycée de l'Académie de Créteil qui a effectué la meilleure progression au bac S entre 2017 et 2018 ?

Travaux dirigés n° 2

Le site internet IMDb (Internet Movies Database – `www.imdb.com`) met à disposition plusieurs fichiers de données sur les œuvres cinématographiques du monde entier. Ces fichiers sont au format TSV (extension `.tsv`), semblable au format CSV à la différence près que le séparateur de champs est la tabulation (caractère `"\t"` en Python).

En raison du volume des fichiers (de 50 Mo à 1 Go), on utilisera pour cet exercice une version appauvrie de ces données, compressée dans l'archive `imdb.zip` présente dans notre espace partagé sur le réseau du lycée.
Pour chacun des fichiers contenus dans cette archive, on donne ci-dessous un extrait des premières lignes et la signification des descripteurs. La chaîne de caractères `"\N"` indiquent que le champ correspondant est inconnu.

- `movies.tsv` : table de données concernant les films (année de sortie postérieure à 2014)

tconst	primaryTitle	year	runtime	genres
tt0062336	El tango del viudo y su espejo deformante	2020	70	Drama
tt0069049	The Other Side of the Wind	2018	122	Drama
tt0069204	Sabse Bada Sukh	2018	\N	Comedy,Drama
tt0100275	The Wandering Soap Opera	2017	80	Comedy,Drama,Fantas
tt0111414	A Thin Life	2018	75	Comedy
tt0112502	Bigfoot	2017	\N	Horror,Thriller

  - `tconst` : identifiant unique du film
  - `primaryTitle` : titre international du film
  - `year` : année de sortie
  - `runtime` : durée du film, en minutes
  - `genres` : genres associés au film (au plus trois)

- `names.tsv` : table de données concernant les personnes

nconst	primaryName	birthYear	deathYear
nm0000005	Ingmar Bergman	1918	2007
nm0000006	Ingrid Bergman	1915	1982
nm0000007	Humphrey Bogart	1899	1957
nm0000008	Marlon Brando	1924	2004
nm0000010	James Cagney	1899	1986
nm0000018	Kirk Douglas	1916	2020

  - `nconst` : identifiant unique de la personne
  - `primaryName` : nom et prénom de la personne
  - `birthYear` : année de naissance, au format YYYY
  - `deathYear` : année de mort (si connue), au format YYYY

- `directors.tsv` : table de données concernant les réalisateurs des films

tconst	directors
tt0062336	nm0749914,nm0765384
tt0069049	nm0000080
tt0069204	nm0611531
tt0100275	nm0765384,nm0749914
tt0111414	nm0398271
tt0112502	nm6883878

  - `tconst` : identifiant unique du film
  - `directors` : identifiant(s) unique(s) du/des réalisateurs

- `casting.tsv` : table de données concernant les rôles

```
tconst      nconst      characters
tt0069049   nm0001379   ["Jake Hannaford"]
tt0069049   nm0462648   ["The Actress"]
tt0069049   nm0000953   ["Brooks Otterlake"]
tt0069049   nm0001782   ["Julie Rich"]
tt0069204   nm0315917   ["Lalloo"]
tt0069204   nm0037026   ["Shankar (Bhompu)"]
```

  - `tconst` : identifiant unique du film
  - `nconst` : identifiant unique de l'acteur/actrice
  - `character` : nom du personnage joué

Partie A

Concevoir une fonction `get_table(filename)` qui prend en argument le nom d'un fichier TSV et renvoie une liste dont chacun des éléments est la liste des champs de chaque ligne.

Par exemple, les instructions :

```
movies = get_table("movies.tsv")
print(movies[68761])

names = get_table("names.tsv")
print(names[942])

cast = get_table("casting.tsv")
print(cast[129523])
```

doivent produire les affichages :

```
['tt7286456', 'Joker', '2019', '122', 'Crime,Drama,Thriller']
['nm0001618', 'Joaquin Phoenix', '1974', '\\N']
['tt7286456', 'nm0001618', '["Arthur Fleck"]']
```

Partie B

Écrire des scripts qui permettent de répondre aux questions ci-dessous.

1. Combien de films sont sortis en 2019 ?
2. Parmi les films sortis en 2018, combien étaient de genre `"Comedy"` ?
3. Quel est le plus long film sorti en 2016 ? Quelle est sa durée ?
4. (a) Quel est l'identifiant unique de Virginie Efira dans les tables de données IMDb ?
   (b) Depuis 2015, dans combien de films Virginie Efira a-t-elle joué ? Utiliser le code de la question précédente.
5. Quels sont les titres des films que la réalisatrice Catherine Corsini a tournés depuis 2015 ?

Partie C

Concevoir un script qui permette d'enregistrer dans un nouveau fichier TSV `movies_and_directors.tsv` les valeurs des champs `tconst`, `primaryTitle` et `directors` pour chacun des films contenus dans les fichiers `movies.tsv` et `directors.tsv`.

On admettra que les identifiants `tconst` des films sont classés dans le même ordre dans ces deux fichiers.

Le fichier produit commencera par les lignes :

```
tconst     primaryTitle                                directors
tt0062336  El tango del viudo y su espejo deformante   nm0749914,nm0765384
tt0069049  The Other Side of the Wind                  nm0000080
tt0069204  Sabse Bada Sukh                             nm0611531
tt0100275  The Wandering Soap Opera                    nm0765384,nm0749914
tt0111414  A Thin Life                                 nm0398271
tt0112502  Bigfoot                                     nm6883878
```

Vérifier ensuite que le nouveau fichier est bien apparu dans le répertoire de travail après exécution du code et qu'il contient les bonnes données.

Partie D

On souhaite maintenant enregistrer dans un nouveau fichier TSV `movies_and_directors_names.tsv` les valeurs des champs `tconst`, `primaryTitle` et `directorsName` pour chacun des films contenus dans les fichiers `movies.tsv`, mais où le champ `directorsName` désigne le nom des réalisateurs (champ `primaryName` du fichier `names.tsv`).

Le fichier produit commencera par les lignes :

```
tconst     primaryTitle                                directorsName
tt0062336  El tango del viudo y su espejo deformante   Raoul Ruiz,Valeria Sarmiento
tt0069049  The Other Side of the Wind                  Orson Welles
tt0069204  Sabse Bada Sukh                             Hrishikesh Mukherjee
tt0100275  The Wandering Soap Opera                    Valeria Sarmiento,Raoul Ruiz
tt0111414  A Thin Life                                 Frank Howson
tt0112502  Bigfoot                                     Mc Jones
```

Si le nom n'est pas renseigné dans le fichier `names.tsv`, on indiquera la valeur `\N`.

$\rightarrow$ Indication : on pourra commencer par écrire une fonction `get_name` qui prend en argument l'identifiant unique d'un réalisateur (`nconst`) et renvoie son nom (`primaryName`).