

I Découverte de la structure du fichier

Ouvrez le fichier *pays.csv* dans l'éditeur de texte de spyder et explicitez la structure de chaque ligne. Il s'agit de comprendre quelles sont les données présentes et comment elles sont séparées les unes des autres.

Exercice 1

A l'aide de la console et de l'aide python (la fonction `help`), comprendre les fonctions `s.strip` quand `s` est une variable contenant une chaîne de caractères.

Même question avec `s.split`.

Quelques exemples pour vous guider (attention, chaque caractère est important!) :

```
1 s = ' ceci est une chaine '
2 s.strip()
3 s.split()
4 s.split('e')
```

Exercice 2

Créer une fonction `converti` qui prend comme argument une ligne de notre fichier (qui peut comprendre des blancs, à retirer) et retourne (`nom_pays`, `population`, `superficie`) où la `population` est un **int** et `superficie` un **int**.

```
1 # pour convertir une chaine en entier on utilise
2 int(ma_chaine) # retourne un entier
```

II Accès à un fichier en python

On veut maintenant lire les données contenues dans le fichier *pays.csv* en python pour pouvoir traiter ces données.

Chemins

Il existe deux manières d'indiquer la position exacte d'un fichier :

- son chemin absolu, qui commence sous windows par une lettre de lecteur.
Par exemple `C:\Users\moi\Documents\TP6.pdf`
- son chemin relatif, c'est à dire son emplacement par rapport au *dossier de travail courant*. Ce dossier courant est déterminé au lancement du programme (quand on lance python, il s'agit au départ du dossier contenant le programme `python.exe`) et peut être modifié à tout moment.

```
1 import os
2 os.getcwd() # retourne, sous forme d'une chaîne de caractère
3 # le nom du dossier courant
```

```
1 os.chdir('C:\\Users') # change le dossier courant
2 os.getcwd() # vérification du dossier courant
```

Exercice 3

Dans Spyder, exécutez le fichier `tp6.py` en utilisant F5 ou la flèche verte. Observez attentivement la ligne `runfile` qui vient d'apparaître dans la console. Elle contient

- le chemin absolu vers votre script python
- le changement de dossier de travail (*working directory*, ou encore *wdir*)

Vérifier maintenant quel est le dossier (ou répertoire) courant et que ce dossier contient bien notre fichier de données (`pays.csv`). Ainsi le chemin relatif de ces fichiers est très simple : il s'agit seulement du nom.

Lecture de fichier en python

On peut lire en python le contenu de n'importe quel fichier (à condition d'avoir les droits de lecture sur le fichier en question) par la syntaxe

```
1 # on précise la table de caractères utilisée par encoding='utf-8'
2 f = open(chemin, 'r', encoding='utf-8')
3 # puis au choix :
4 texte = f.read() # on récupère le contenu en entier
5 # ou si on préfère
6 lignes = f.readlines() # on récupère la liste des lignes
7 # et enfin, à ne pas oublier
8 f.close()
```

Il faut choisir le mode de lecture (texte entier ou par lignes) : si vous exécutez les deux commandes de lecture à la suite, la variable `lignes` contiendra une liste vide car le fichier aura déjà été lu.

Pour éviter d'avoir à fermer le fichier manuellement (et même s'il y a une erreur le fichier sera correctement fermé) :

```
1 with open(chemin, 'r', encoding='utf-8') as f:
2     texte = ...
3     # ou
4     lignes = ...
5 # ici le fichier est bien fermé et on peut travailler avec les données lues.
```

Exercice 4

Créer une fonction `lit_pays(chemin)` qui prend en argument le chemin du fichier csv (ce chemin est une chaîne) et retourne 3 listes : celle des noms de pays, celle des populations et celle des superficies.

On n'oubliera pas de fermer le fichier ouvert après lecture.

Exercice 5

Compléter la fonction `somme` qui prend une liste de nombres et retourne la somme de tous ces nombres.

Exercice 6

Créer une liste `proportions` qui contient un nombre par pays de notre fichier : la proportion de la population de ce pays par rapport à la population mondiale. Ce nombre devra être un flottant entre 0 et 1 et la somme des proportions doit être 1.

Exercice 7

Trouver le pays ayant la plus forte et celui ayant la plus faible densité de population et afficher leurs noms.

III Graphiques

Rappels sur l'utilisation

```
1 import matplotlib.pyplot as plt
```

L'affichage de courbes se fait de la manière suivante : on considère deux listes de nombres X et Y de même longueur n

```
1 plt.plot(X, Y)
```

trace la ligne brisée qui relie tous les points de coordonnées $(X[i], Y[i])$ pour i entre 0 et $n - 1$.

Voici un exemple

```
1 import math
2 X = [math.pi*i/100 for i in range(201)]
3 Y = [math.sin(x) for x in X]
4 plt.plot(X, Y)
```

Exercice 8

Adapter l'exemple précédent pour tracer la courbe représentative de \exp sur l'intervalle $[-2, 2]$ en utilisant 401 points (on utilise 401 points pour couper l'intervalle $[-2, 2]$ en 400 segments de même longueur).

Exercice 9

Créer la liste `pop` telle que `pop[i]` contient le nombre de pays ayant une population (en millions) dans $[20i, 20(i + 1)[$. Créer également la liste `abscisses` des $20*(i + 1)$ pour toutes les valeurs de i dont vous aurez besoin pour `pop` et afficher le graphique correspondant.

Comparer à

```
1 plt.hist(Lpop, abscisses)
```

où `Lpop` est la liste de toutes les populations.

IV Ecriture

Exercice 10

Ecrire dans le fichier `densité.csv` les données : nom du pays ; densité.

On pourra au choix donner des titres, ou non, aux colonnes dans la première ligne.

Pour écrire dans un fichier situé à l'adresse `chemin` :

```
1 f = open(chemin, 'w')
2 ...
3 f.write(ici mettre une chaine de caractères à écrire dans le fichier)
4 # on peut utiliser plusieurs appels à f.write
5 ...
6
7 f.close()
```

Pour passer à la ligne, on utilise le caractère spécial `\n`

Exercice 11

Écrire une fonction **combine(L1, L2, L3)** qui prend 3 listes de même longueur comme arguments et retourne la liste `[[L1[0], L2[0], L3[0]], ...]` où chaque éléments est une liste de longueur 3.

Exercice 12

Expérimenter la fonction

```
1 L.sort()
```

où `L` est une liste construite avec la fonction **combine**. On fera au moins 3 expériences pour 3 valeurs pertinentes de `L1`.

Exercice 13

Créer les fichiers csv *pays-par-population.csv* et *pays-par-superficie.csv* qui contiennent la liste de nos pays triées dans l'ordre croissant suivant leur titres...

V Bonus

Trouver comment tracer un cercle de centre donné et rayon donné, comment tracer un polygone régulier à n côtés, une droite d'équation $y = ax + b$ (on donne a, b), cette même droite mais limitée à un rectangle donné (comment se donner un rectangle le plus efficacement possible?).