

Traitement des données en tables

qkzk

2020/07/31

Objectifs du TP : manipuler des données en table dans Python

1. Importer un fichier csv en Python,
2. sélectionner des enregistrements,
3. projeter une table,
4. exporter une table vers un fichier csv.

Exercice 1.

1. Importer

1. Exécuter le code suivant afin de télécharger le fichier csv utilisateurs.
2. Créer une fonction Python permettant d'importer le fichier dans une table. La table renvoyée est une liste de dictionnaires Python.
3. Utiliser cette fonction pour créer la table `utilisateurs`.

2. Sélections

Pour toutes les sélections à effectuer, on partira de la table créée à la question précédente.

On donnera une fonction Python par question (pouvant appeler d'autres fonctions).

1. Créer la table des enregistrements des joueurs ayant moins de 300 points au premier jeu.
2. Créer la table des enregistrements des filles.
3. Créer la table des enregistrements des utilisateurs dont l'adresse mail fait partie du domaine `@example.fr`

3. Projections

Pour chaque *projection* on créera une fonction qui retourne une table.

Celle-ci sera toujours une liste de dictionnaires, comme précédemment. précédemment

1. *Meilleurs scores*
 - a. Créer les projections des meilleurs scores pour chaque jeu.
 - b. Utiliser cette fonction pour créer la liste des meilleurs scores pour chaque jeu, triées par score croissant.
 - c. Calculer le score moyen de chaque jeu.
2. *Adresses mail*
 - a. Créer la projection des adresses mail. Le résultat attendu est la liste de toutes les adresses des joueurs.
 - b. Recommencer en ne conservant que les 10 joueurs ayant le plus mauvais score pour chaque jeu. Enfin, effacer les doublons.

Exercice 2.

1. Récupérer le fichier `ville_virgule.csv`. Examinez soigneusement son contenu (vous pouvez l'ouvrir localement dans sublime text). Importez le dans Python sous la forme d'une table, appelée `villes` et qui est une liste de dictionnaires python. Attention : ce fichier étant volumineux, il est préférable de n'afficher qu'une partie de la table avec `print(villes[:10])`, par exemple.

2. Combien de villes contient ce fichier ? Quelles sont les différents champs ? Vous utiliserez des outils de Python pour répondre.
3. Votre ville figure-t-elle dans le fichier ? Afficher l'enregistrement qui lui correspond. Si ce n'est pas le cas, vous pouvez travailler avec Lille.
Écrire un programme qui affiche les informations de votre ville dans une phrase rédigée : "J'habite Lille, qui comporte x habitants et est située à y de latitude et z de longitude. Son altitude minimale est w" etc.
4. Après avoir sélectionné toutes les villes situées à plus de 1500m d'altitude, calculer la moyenne de leur nombre d'habitants pour le dernier relevé de population.
5. Sélectionner une table avec `nom`, `altitude minimale`, `densité` pour toutes les villes ayant une densité inférieure à 50 (habitants au km²). Enregistrer cette table au format csv.
6. Quelle est la seule ville française ayant une densité de population supérieure à 50 (h/km²) et une altitude minimale supérieure à 1500m ? Répondre à l'aide de Python.
7. Sélectionnez les altitudes minimales des villes françaises. Quelle est l'altitude moyenne des villes françaises ?
8. Recommencez avec la population.
9. Déterminez :
 - a. La ville la plus peuplée,
 - b. La ville ayant l'altitude minimale la plus élevée,
 - c. La ville la plus dense.