BUT Science des Données - Semestre 3

COMPETENCE 4 : Développer un outil décisionnel - parcours VCOD

Collecte de données web

F.GARNIER

Plan de la Saé

Chapitre 1 : Web Scraping

Chapitre 2 : API

Sujet noté alternant

---

1) Contexte Web scraping

Pour ce sujet, vous serez libre de choisir le site à scraper. Pensez à analyser les données à traiter, à évaluer la difficulté de récupération de ces données par la technique de web scraping et enfin à imaginer des indicateurs qui aient du sens pour l'application.

TRAVAIL A FAIRE
Etapes à suivre pour réaliser l'application Python : - Recherche des données et du site à scaper en lien avec notre domaine de compétences et contenant des données géographiques : réfléchir à l'intérêt des données et aux indicateurs potentiels à produire. - Validation par votre enseignant - Scraping des données du site web dans un dataFrame. - Tableau de bord à partir du dataframe en Python : au moins une carte et un graphique significatif.

2) Contexte API : Villes du monde

Objectif du programme python : Filtrer les villes par population et afficher celles d'un pays spécifique

1. Utiliser l'API `GeoNames` pour récupérer uniquement les villes ayant une population minimale spécifiée et appartenant à un pays donné.
2. Afficher ces villes sur une carte interactive avec Folium, avec des marqueurs colorés en fonction de la population.
3. Ajouter un indicateur supplémentaire, comme l'altitude moyenne des villes affichées.

Allez sur la documentation de l'API (ou sur Postman) pour maîtriser les paramètres potentiels et les différentes colonnes du jeu de données (population, altitude, type de lieu, …). **Il est recommandé de créer un compte gratuit sur GeoNames pour obtenir votre propre nom d'utilisateur** et utiliser l'API (un compte demo existe mais est limité).

Par exemple l'URL suivante permet d'obtenir les villes de plus de 100 000 habitants en France : http://api.geonames.org/searchJSON?country=FR&minPopulation=100000&username=demo

TRAVAIL A FAIRE

L'objectif de ce contexte est de faire une application python permettant :

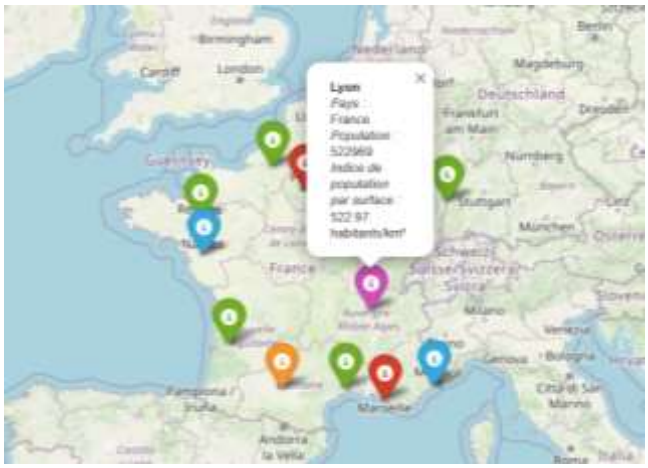
a) **de charger le jeu de données initial dans un dataframe pandas ;**

```
data = pandas.DataFrame(data['results'])
```

b) **d'afficher une carte folium lancée automatiquement représentant les villes correspondants à certains critères. Le marqueur sera d'une certaine couleur selon un criète que vous aurez défini**

Exemple ici **:** Couleur des marqueurs en **France** pour les villes de plus de **200 000** habitants avec une popup indiquant certaines informations

- *Population inférieure à 300 000 : couleur verte*
- *Entre 300 000 et 400 000 : couleur bleue*
- *Entre 400 000 et 500 000 : couleur orange*
- *Entre 500 000 et 600 000 : couleur violet*
- *Les autres : couleur rouge*



c) de créer deux autres indicateurs que vous jugerez utile à l'application.

Remarques :

- Pensez à nettoyer vos données en ôtant toutes les lignes non nécessaires ou contenant des valeurs manquantes ou aberrantes pour les variables nécessaires à l'établissement de ces indicateurs.
- Ne chargez vos dataframes qu'avec les données utiles
- Un menu en mode web sera apprécié avec l'accès aux fonctionnalités (b, c et d)

3) Evaluation

Les deux livrables seront à restituer sur Updago pour le **dimanche 10 novembre** 23h59 au plus tard. Les éléments d'évaluation seront les suivants :

- Opérationnalité (*réponses aux besoins exprimés et respect du cahier des charges*)
- Automatisation de l'outil et optimisation, ergonomie,
- Bonnes pratiques de développement : Commentaires, Indentation, **Découpage modulaire si besoin**, Gestion des erreurs avec try catch si besoin…

Enfin, une petite évaluation orale de 5 à 10 minutes permettant de juger de l'appropriation de l'application pour tous les membres du groupe sera effectuée la semaine suivant le dépôt.