

SAE-15 : Traiter les données

synopsis :

La société **Huile de Code SARL** (<http://huiledecode.org>) qui gère le site Internet de la société **ControlTower** (<http://controltower.fr>), ce dernier réalisant des ventes de disques vinyles et dont les clients sont internationaux, vous demande de réaliser un logiciel en Python 3.

Huile de Code vous demande de **réaliser des statistiques** à partir des données de connexion au site Internet. Pour ce faire, la société **HDC** vous **met à disposition un fichier de log du serveur Apache 2** sur lequel est hébergé le site Internet de **ControlTower**.

Vous pouvez télécharger le fichier de log à l'adresse Internet suivante :

http://eismall.otf.cloud/iut/controltower_access.zip

Travail à réaliser :

1 Analyse du fichier de log apache 2

Vous devrez dans un premier temps réaliser une analyse des données du fichier de log, pour comprendre sa structure. Vous devrez donc effectuer des recherches sur Internet pour savoir quelles données sont stockées dans ce fichier.

Vous pourrez faire un tableau permettant de voir rapidement à quoi correspondent les informations qui sont stockées dans ce fichier.

2 Quelles statistiques peut-on faire sur ces données ?

- Connaître l'adresse IP et la géolocaliser via un API en ligne : '165.225.76.120' via <https://ip-api.com> (vous chercherez comment utiliser ce service depuis un programme Python)
- Afficher une courbe du nombre de requêtes en fonction de la date de chaque ligne : "[09/Nov/2021:11:36:21', '+0100]"
- Connaître et dénombrer les systèmes d'exploitation qui se connectent sur le serveur : "(Windows', 'NT', '10.0;', 'Win64;', 'x64)"
- Connaître et dénombrer les navigateurs Internet qui se connectent sur le serveur : "Chrome/87.0.4280.141"
- Connaître le nombre d'erreur de réponse en fonction du code de retour HTTP : "200", "404", "500" ...
- Ainsi que toutes statistiques que vous jugerez pertinentes

3 Liste des fonctionnalités à coder avec Python3

- Fonction "**parsing(file)**" : avec en paramètre « file », le nom du fichier au format chaîne de caractères pour la lecture des lignes du fichier et stockage des données dans une structure facilement exploitable pour exporter des formats comme CVS, JSON, XML.
- Fonction "**getIP_infos(ip)**" : permettant de connaître des informations sur l'adresse IP via le site : <https://ip-api.com>
- Fonction "**exportToCSVFile(liste)**" : permettant d'exporter l'ensemble des données au format CSV (chercher des informations sur ce format)
- Fonction "**exportToJSONFile(liste)**" : permettant d'exporter l'ensemble des données au format JSON (chercher des informations sur ce format)
- Fonction "**exportToXMLFile(liste)**" : permettant d'exporter l'ensemble des données au format XML (chercher des information sur ce format)

NB : vous direz quel format vous semble le mieux adapté pour réaliser des graphiques et des statistiques selon vous.

4 Rendu pour évaluation de votre travail : un Rapport PDF.

Vous devrez rendre un document d'analyse qui présentera :

- Ce que contient le fichier de log apache 2 (le tableau est à réutiliser ici)
- Expliquez quelle structure Python vous allez utiliser pour stocker les données extraites du fichier de log dans la méthode « **parsing(file)** » et pourquoi celle là.
- Comment vous avez codé les fonctions demandées et éventuellement les fonctions supplémentaires
- L 'explication du code de chaque fonction en Annexe de votre rapport.
- Votre démarche et le résultat de vos recherches pour utiliser le service en ligne de :
<https://ip-api.com>
- Des graphiques répondant aux demandes des statisitiques, qui pourront être réalisés dans un Tableur à partir des données exportées par votre programme. Choisissez le bon format pour l'importation dans le Tableau.
- Vous pourrez afficher les données dans une page HTML avec une carte de géolocalisation basée sur OpenStreetMap. Réalisez des recherches pour comprendre comment on peut afficher un fond de carte OpenStreetMap et y placer des points d'intérêts que représentent les adresses IP.