TP N° 1

Outils pour la collecte et la sauvegarde de données issues du web

Dans la cadre de la SAE 15, vous devez vous approprier les outils permettant de collecter des données issues du web, de les mettre en forme, de les sauvegarder, de les traiter, les analyser et les publier.

L'objectif de ce TP est d'expérimenter l'emploi d'outils de base pour la collecte de données disponible sur le web et la sauvegarde dans des fichiers.

Partie 1 : requêtes http en Python

La première étape consiste à collecter les données dans le but de les analyser. Comme indiqué lors de la présentation de la SAE 15, nous utiliserons les données fournies par le site https://data.montpellier3m.fr. Plus particulièrement, nous nous intéressons dans ce TP aux données d'occupation des parkings de la ville de Montpellier. Ces données sont disponibles sur le lien suivant : https://data.montpellier3m.fr/dataset/disponibilite-des-places-dans-les-parkings-de-montpellier-mediterranee-metropole

Pour chaque parking, un fichier xml est fourni et donne diverses informations sur le parking.

A titre d'exemple, pour le parking « gare » situé à proximité de la gare ferroviaire St-Roch de Montpellier, une requête permet d'obtenir les informations suivantes :

```
<park>
<DateTime>2022-01-17T17:30:08</DateTime>
<Name>GARE</Name>
<Status>Open</Status>
<Free>346</Free>
<Total>656</Total>
<DisplayOpenIf>-1</DisplayOpenIf>
</park>
```

Ces informations sont disponibles sur le lien suivant :

https://data.montpellier3m.fr/sites/default/files/ressources/FR_MTP_GARE.xml

Chaque parking possède un lien particulier permettant d'obtenir les données le concernant (voir sur le site).

Afin d'envoyer une requête http pour récupérer les données, nous utiliserons la librairie « **requests** »

1)- Exécuter le code python suivant et expliquer ce qu'il fait :

```
import requests

response=requests.get("https://data.montpellier3m.fr/sites/default/files/ressources/FR_MTP_COME.xml")

print(response.text)
```

2)- On peut remarquer que le lien pour un parking est une chaine de caractères constituée de trois parties :

https://data.montpellier3m.fr/sites/default/files/ressources/ suivi du nom du parking (par exemple **FR_MTP_GARE**) puis de l'extension indiquant le type de fichier (ici **.xml**)

La liste de tous les parkings est donnée ci-dessous :

```
parkings=['FR_MTP_ANTI','FR_MTP_COME','FR_MTP_CORU','FR_MTP_EURO','FR_MTP_FOCH','FR_MTP_GAMB','FR_MTP_GARE','FR_MTP_TRIA','FR_MTP_ARCT','FR_MTP_PITO','FR_MTP_CIRC','FR_MTP_SABI','FR_MTP_GARC','FR_CAS_SABL','FR_MTP_MOSS','FR_STJ_SJLC','FR_MTP_MEDC','FR_MTP_OCCI','FR_CAS_VICA','FR_MTP_GA109','FR_MTP_GA250','FR_CAS_CDGA','FR_MTP_ARCE','FR_MTP_POLY']
```

Ecrire un programme permettant de récupérer et d'afficher toutes les données de tous les parkings de la ville.

Partie 2 : analyse d'un fichier xml

Dans la partie précédente, nous avons récupéré les données utiles provenant des différents parkings de la ville. Les données sont fournies sous la forme d'un fichier xml.

Afin de récupérer une donnée précise (par exemple le nombre de places libres dans un parking, on peut, bien entendu, traiter la chaine de caractère obtenue afin de récupérer la donnée utile. Cependant, il est possible d'utiliser une librairie pour effectuer cette tâche. On parle alors de « parser » ou analyser un fichier .xml.

Dans ce but, nous allons utiliser la librairie « **lxml** ».

3)- Etudier le programme suivant et décrire ce qu'il effectue

```
import requests

from lxml import etree


response=requests.get("https://data.montpellier3m.fr/sites/default/files/ressources/FR_MTP_COME.xml")

print(response.text)

f1=open("FR_MTP_COME.txt","w", encoding='utf8')

f1.write(response.text)

f1.close()
```

```
tree = etree.parse("FR_MTP_COME.txt")
for user in tree.xpath("Name"):
    print('Nom du parking :',user.text)
for user in tree.xpath("Total"):
    print('Nombre total de places :',user.text)
for user in tree.xpath("Free"):
    print('Nombre de places libres :',user.text)
```

4)- Ecrire un programme qui récupère le nombre de places libres dans chaque parking et qui sauvegarde ces données dans un fichier texte (une donnée par ligne contenant uniquement le nombre de places libres. Il faudra vérifier pour chaque parking, s'il est ouvert.

5)- modifier votre programme pour sauvegarder un fichier contenant dans chaque ligne : le nom du parking puis le nombre de places libres.

6)- écrire un programme qui donne le pourcentage de places libres pour chaque parking ainsi que le pourcentage de places libres dans toute la ville.

Partie 3 : gestion du temps

Comme vous l'avez sans-doute remarqué, les informations sur les parkings fournies sont datées. Les données sont valables au moment de la requête. Bien entendu, le nombre de places disponibles évolue au cours du temps en fonction du nombre de voitures qui entrent ou sortent du parking. Si l'on souhaite suivre l'évolution de l'occupation d'un parking au cours du temps (par exemple sur une journée), il convient tout d'abord de définir une période d'échantillonnage Te (intervalle de temps entre deux mesures) puis, de lancer une requête à chaque période d'échantillonnage Te. Nous avons donc besoin d'automatiser cette tâche (à moins que vous ayez du temps à perdre pour lancer manuellement la requête à chaque Te !!! 😊).

Il existe plusieurs façons d'exécuter une tâche de manière répétitive à une cadence Te. Pour ce TP nous allons employer un moyen simple. Pour cela, nous allons utiliser la librairie « time »

7)- Etudier le programme suivant et expliquer ce qu'il affiche :

```
import time
temps=int(time.time())
print(temps)
```

Que représente le nombre obtenu ? (indication : epoch, UNIX)

8)- Ecrire un programme qui permet de récupérer l'occupation du parking « FR_MTP_GARE » toutes les 10 secondes pendant 5 minutes et qui sauvegarde ces données dans un fichier.

9)- Ecrire un programme qui permet le suivi de l'occupation de tous les parkings de Montpellier en permettant à l'utilisateur de choisir la période d'échantillonnage Te et la durée de l'acquisition. Il pourra également indiquer le nom du fichier dans lequel seront enregistrées les données.

10)- vous possédez à présent la majorité des outils vous permettant d'acquérir et de sauvegarder des données ! il est tant pour vous de fabriquer votre propre librairie de fonctions afin de faciliter la réalisation de votre projet de SAE (et d'autres projets dans le futur !). Expliquer votre démarche et commentez correctement vos fonctions.