

TD5 : Dénivelés de randonnées

Ce sujet est une synthèse des travaux avec GraphHopper et avec Json.

Le fichier mini-rando-gps.json fournit des séries de positions GPS correspondant à des traces GPS de sorties randonnée de M. Toulemonde. On cherchera dans ce TD à écrire un programme calculant les dénivélés cumulés positif et négatif de chacune de ces randonnées. Pour cela, vous utiliserez l'API GraphHopper via le module Python graphh. On rappelle que la documentation de graphh est disponible à cette adresse : <https://graphh.readthedocs.io/en/latest/>.

Préambule

- Dans votre dossier PythonOpenData, créez un sous-répertoire TD5 dans lequel vous réaliserez l'ensemble des travaux.
- Lancer l'éditeur Visual Studio Code.
- Dans Visual Studio Code, ouvrir le dossier PythonOpenData/TD5.
- Télécharger le fichier mini-rando-gps.json dans le dossier PythonOpenData/TD5.
- Copier le fichier credentials.json du TD1 dans le dossier PythonOpenData/TD5.
- Créer un fichier td5.py dans lequel vous écrirez le code du TD.

Exercice 1 : Altitudes avec GraphHopper

- Q.1.1. Écrire une fonction qui prend en entrée un lieu et un client GraphHopper, et retourne l'altitude de ce lieu. Tester en affichant l'altitude de Rennes, de Saint-Malo puis de Chamonix.
- Q.1.2. Écrire une fonction qui prend en entrée une liste de positions GPS (chacune codée sous la forme d'un dictionnaire comme précisé plus bas) et un client GraphHopper et retourne une liste d'altitudes. Vous pourrez utiliser l'exemple suivant pour vos tests :

```
lst_gps = [  
    {"lng": -1.426533, "lat": 48.005135},  
    {"lng": -1.418127, "lat": 47.986058},  
    {"lng": -1.427611, "lat": 47.989871},  
    {"lng": -1.430202, "lat": 48.000354}  
]
```

Exercice 2 : Dénivelés

En utilisant la fonction écrite à la question précédente, écrire une fonction qui prend en entrée une liste de positions GPS et un client GraphHopper et retourne la somme des dénivélés positifs (d'une part) et négatifs (d'autre part), entre les points GPS successifs.

Par exemple, en appelant cette fonction avec la liste de coordonnées GPS précédente, on devrait obtenir la paire de dénivelé (29.07, 28.55). En effet, les altitudes des points GPS étant les suivantes : [35.29, 64.36, 48.78, 35.81], la somme des dénivélés positifs et négatifs donne (29.07, 28.55).

Exercice 3 : Analyse de randonnées

En utilisant la fonction écrite à la question précédente, écrire une fonction qui prend en entrée un nom de fichier JSON (contenant des informations sur diverses randonnées) et un client GraphHopper et affiche, pour chaque randonnée, son nom (attribut "name") et la somme de ses dénivelés positifs (d'une part) et négatifs (d'autre part).

Pour le fichier mini-mini-rando_gps.json, on doit obtenir une sortie du type :

```
TraceGPS Le long de la quincampoix - Pire-sur-Seiche
    - Dénivelé positif cumulé : 15.08
    - Dénivelé négatif cumulé : 15.08
TraceGPS Issued Messac - CIRCUIT DU PORT
    - Dénivelé positif cumulé : 4.47
    - Dénivelé négatif cumulé : 4.47
```

Exercice 4 : Fichier de synthèse des randonnées

Écrire une fonction qui prend en entrée un nom de fichier JSON (contenant des informations sur diverses randonnées) et un client GraphHopper, et écrit dans un nouveau fichier JSON (dont le nom sera passé en paramètre de la fonction) une liste de dictionnaires contenant le nom de la randonnée et les informations de dénivelés positif et négatif), soit quelque chose du type :

```
[
  {
    "name": "TraceGPS Le long de la quincampoix - Pire-sur-Seiche",
    "D+": 15.08,
    "D-": 15.08
  },
  {
    "name": "TraceGPS Issued Messac - CIRCUIT DU PORT",
    "D+": 4.47,
    "D-": 4.47
  }
]
```