

Tout le travail doit être sauvegarder et sera récupéré dans le dossier « Restitution de devoirs ».

I. Créer une carte personnalisé avec Python

Nous allons utiliser les cartes proposées par Open Street Map et le langage Python afin de générer des cartes personnalisées. Plus exactement, nous allons utiliser une bibliothèque Python nommée Folium. Une bibliothèque Python permet de rajouter des fonctionnalités au langage de base. Folium va donc nous permettre de créer nos propres cartes à partir des cartes proposées par Open Street Map.

Ouvrir le logiciel **EduPython**,

Exercice 1

1. Créez un dossier et nommez-le "cartoOSM_NomDeFamille"
2. Utilisez le logiciel **EduPython** pour saisir le programme ci-dessous. Le fichier devra être enregistré dans le répertoire créé précédemment (le fichier pourra être nommé "Carte.py").

```
import folium
c= folium.Map(location=[43.2965, 5.3698])
c.save('maCarte1.html')
```

Une fois le code ci-dessus exécuté, rendez-vous dans le répertoire que vous avez créé. Vous devriez trouver un fichier "maCarte.html". Double-cliquez sur ce fichier, cela devrait normalement ouvrir un navigateur web : la carte centrée sur la ville de Marseille est à votre disposition. Notez bien que nous avons une véritable carte et pas une simple image (il est possible de zoomer ou de se déplacer).

Le programme est simple à comprendre :

- La première ligne : "import folium" permet d'importer la bibliothèque folium afin de pouvoir l'utiliser
- La deuxième ligne est le coeur de notre programme, nous définissons une variable "c" qui va contenir notre objet carte. "folium.Map(location=[43.2965, 5.3698])" génère cet objet carte, la carte sera centrée sur le point de latitude "43.2965" et de longitude "5.3698". Plus généralement nous avons "folium.Map(location=[latitude, longitude])". Il suffit donc de renseigner la bonne longitude et la bonne latitude pour que la carte soit centrée sur le point désiré.
- La dernière ligne permet de générer la page HTML qui va permettre d'afficher la carte.

3. Modifiez le programme pour qu'il génère une carte centrée sur la ville de votre choix (la longitude et la latitude d'une ville sont facilement trouvables sur le web). Et changez le nom de la carte générée par 'maCarte2.html'.

4. Il est possible d'obtenir un niveau de zoom différent en ajoutant un paramètre "zoom_start". Saisissez et testez le programme ci-dessous (si vous désirez garder la carte précédente, n'oubliez pas de modifier la dernière ligne en changeant le nom du fichier HTML généré).

```
import folium
c= folium.Map(location=[43.2965, 5.3698],zoom_start=15)
c.save('maCarte3.html')
```

Plus la valeur de "zoom_start" sera grande et plus le zoom sera important.

5. Afin de vraiment personnaliser la carte, il est possible d'ajouter des marqueurs sur la carte. Un marqueur sera simplement défini par ses coordonnées (latitude et longitude). Saisissez et testez le programme ci-dessous.

```
import folium
c= folium.Map(location=[43.2965, 5.3698],zoom_start=20)
folium.Marker([43.2965, 5.3698]).add_to(c)
c.save('maCarte4.html')
```

Nous avons uniquement ajouté la ligne "folium.Marker...", il faut juste renseigner les coordonnées souhaitées (ici 43.2965 pour la latitude et 5.3698 pour la longitude).

Il est possible d'ajouter plusieurs marqueurs sur une même carte, il suffira d'ajouter autant de ligne "folium.Marker([latitude, longitude]).add_to(c)" que de marqueurs désirés.

6. Il est possible d'associer une information à un marqueur en ajoutant le paramètre "popup" Saisissez et testez le programme ci-dessous :

```
import folium
c= folium.Map(location=[43.2965, 5.3698],zoom_start=20)
folium.Marker([43.2965, 5.3698],popup="Marseille").add_to(c)
c.save('maCarte5.html')
```

Il suffit de cliquer sur le marqueur pour que l'information définie par le paramètre "popup" apparaisse à l'écran (ici en cliquant sur le marqueur nous verrons donc apparaître "Marseille").

7. Modifier le programme précédent pour qu'il génère une carte centrée sur le lieu de votre choix en ajoutant plusieurs marqueurs ainsi que l'information associée à chacun de ces marqueurs.

II. Calculs d'itinéraires

La base de données OSM contient les routes (enfin, la plupart des routes...). En utilisant ces données, il est possible de développer des outils capables de calculer des itinéraires routiers (comme le propose tous les logiciels "GPS" : Waze, ViaMichelin, Mappy...) : vous renseignez votre lieu de départ, votre lieu d'arrivée puis le logiciel calcule votre itinéraire. Ce calcul d'itinéraire repose sur des algorithmes relativement complexes, qui permet d'obtenir le plus court chemin entre deux points.

Exercice 2 : Sans entrer dans les détails, l'algorithme de Dijkstra travaille sur des graphes (chaque ville est un sommet du graphe et chaque route est une arête du graphe), visionner cette vidéo pour en savoir plus.

<https://youtu.be/JPcMkFrKio>

Expliquer en quelques lignes le principe de l'algorithme de Dijkstra.

III. Contribuer à OpenStreetMap de façon collaborative

OpenStreetMap (OSM - <https://www.openstreetmap.org>) est un projet de cartographie libre du monde (permettant par exemple de créer des cartes sous licence libre), en utilisant le système GPS et d'autres données libres. Il a la particularité d'être collaboratif, c'est à dire que chaque utilisateur peut participer à titre individuel à OpenStreetMap, pour décrire des lieux qu'il connaît bien.

Il a été mis en route en juillet 2004 par Steve Coast au University College de Londres et compte aujourd'hui plus de 4 500 000 contributeurs.

En France, il est de plus en plus utilisé par les collectivités (communes, départements) pour collecter de façon participative des données sur leur territoire.



1) Découverte d'OpenStreetMap

- a. Dans OpenStreetMap (<https://www.openstreetmap.org>) utiliser la recherche pour aller aux coordonnées : (-22.91213,-43.23013). Où est-on arrivé (lieu, ville, pays) ?
- b. Où serait-on allé si on avait cherché le lieu de coordonnées (-43.23013, -22.91213) ?

2) Utilisation des couches de cartes

- a. Chercher la ville de Combourg et faire apparaître la couche de carte « transport ».
- b. Trouver une ligne de bus, son début et son terminus.