python, cartographie, carte, code, em

Rédaction démarrée le 31 mai 2021

# Cartographie en python

Quelques notes pour réaliser des cartes statiques en téléchargeant des «tuiles» d'OpenStreetMap sur lesquelles sont ajoutées des données saisies avec umap.
Avec pour but de les imprimer.

nb : soyez indulgent, je débute en python

Deux pistes

- Static Map : https://github.com/komoot/staticmap
- py-staticmaps : https://github.com/flopp/py-staticmaps

## Tests

### Static Map

#### Installation de Static Map

```
pip3 install staticmap
pip3 show staticmap # permet de savoir quelle version installée
```

#### Premier script

```
# test avec https://github.com/komoot/staticmap
# Python 3.5.3 / pip 9.0.1 / staticmap 0.5.5
# Debian 9.5 @ kirin / 20210531

from staticmap import StaticMap, Line

m = StaticMap(3000, 4000, 10)
m.add_line(Line(((13.4, 52.5), (2.3, 48.9)), 'blue', 3))
image = m.render()
image.save('map.png')
```

Et ça produit bien une carte en haute définition, c'est encourageant!

On peut choisir le fournisseur de «tuiles» à la création de la carte (ex. en noir et blanc)

```
m = StaticMap(3000, 4000, 10, url_template='http://a.tile.stamen.com/toner/{z}/{x}/{y}.png')
```

J'en ai trouvé une liste ici : https://wiki.openstreetmap.org/wiki/Tile_servers

### py-staticmaps

Je tente l'installation avec

```
pip3 install py-staticmaps
```

Mais ça bloque sur une erreur, et quand j'essaie de la résoudre, c'est la réaction en chaîne de messages d'erreur que je ne comprends qu'à moitié (la moitié vide). Alors, on verra plus tard...

## Réalisation

### Étape 1 : le fond de carte

Un fond de carte de Quimper, en noir et blanc, en A4 paysage 300 dpi (avec une petite marge de 5 mm sur chaque bord)

- définition x : (29.7 - 1.0) / 2.54 * 300 = 3390 pixels
- définition y : (21 - 1.0) / 2.54 * 300 = 2362 pixels

Pour trouver le niveau de zoom et centrer la carte sur un point, on peut utiliser openstreetmap . Dans l'URL, on peut y lire les coordonnées lat/lon et le niveau de zoom, exemple : https://www.openstreetmap.org/#map=13/47.9968/-4.1043

- zoom : 13
- lat. : 47.9968
- lon. : -4.1043

Au sujet des niveaux de zoom : https://wiki.openstreetmap.org/wiki/Zoom_levels

Pour le fond de carte :

```
# test avec https://github.com/komoot/staticmap
# Python 3.5.3 / pip 9.0.1 / staticmap 0.5.5
# Debian 9.5 @ kirin / 20210531

from staticmap import StaticMap, CircleMarker

m = StaticMap(3390, 2362, url_template='http://a.tile.stamen.com/toner-lite/{z}/{x}/{y}.png')

marker_outline = CircleMarker((-4.1043, 47.9968), 'white', 18)
marker = CircleMarker((-4.1043, 47.9968), '#0036FF', 12)

m.add_marker(marker_outline)
m.add_marker(marker)

image = m.render(zoom=16)
image.save('fond_de_carte_quimper.png')
```

Le jeu de tuile est toner-lite de Stamen (voir http://maps.stamen.com/ )
Pour une raison qui m'échappe, le niveau de zoom n'est pas celui que j'avais trouvé avec OSM ?

Ce qui donne (extrait seulement, le fichier complet fait 4.8 MO)

## Étape 2 : les données

Une carte de test : https://umap.openstreetmap.fr/fr/map/quimper_test_data_620279 (edit / Les données sont exportées en geojson

Les marqueurs sont numérotés, ils devront être reliés comme suit : (1,2), (2,3), (3,4), (4,5), (5,6), (6,7) mais par la suite ça devra fonctionner avec n'importe quelle paire de nombre.

### Installation de geojson

```
pip3 install geojson
```

### Lire les données avec geojson

```
import geojson
with open("./quimper.geojson") as f:
    data = geojson.load(f)
for feature in data['features']:
    nom = feature['properties']['name']
    lon = feature['geometry']['coordinates'][0]
    lat = feature['geometry']['coordinates'][1]
    print("nom : ", nom, " lon : ", lon, ", lat : ", lat)
```

**Chemins** Les chemins qui relient les points sont indiqués dans une liste de liste

```
chemins =   [
```

```
                [1, 2],
                [2, 3],
                [3, 4],
                [4, 5],
                [5, 6],
                [6, 7]
            ]
for chemin in chemins:
    print("Tracer un chemin entre", chemin[0], " et ", chemin[1])
```

## Ressources

Des alternatives / services alternatifs : https://wiki.openstreetmap.org/wiki/Static_map_images
Différents fournisseurs de tuiles (fonds de carte) : https://wiki.openstreetmap.org/wiki/Tile_servers

Article extrait de : http://lesporteslogiques.net/wiki/ - **WIKI Les Portes Logiques**
Adresse : **http://lesporteslogiques.net/wiki/ressource/code/cartographie/start?rev=1622501642**
Article mis à jour: **2021/06/01 00:54**