SER – Labo 3 JSON

# Descriptif des classes

Nous avons deux classes. La première « Country » représente un pays. Ella a donc un nom, une abréviation, un type et une liste de liste de coordonnées. Le type permet de savoir si le pays représenté possèdes des enclaves ou des îles et la liste de liste de coordonnées représente toutes les frontières du pays. Le fait que se soit une liste de liste permet de séparer les différentes frontières des îles ou des enclaves. Cette classe nous permet de rendre plus lisibles l’accès aux données de celle-ci (comparé à l’accès via les classe JSONObject ou JSONArray).

La seconde classe est la « Main ». Elle contient le programme principal et s’occupe donc du parsing du fichier countries.gejson et de la création d’un fichier .kml.

# Difficultés rencontrées

Durant ce projet, nous avons rencontré quelques problèmes.

Pour commencer, il a fallut comprendre le fichier countries.geojson que nous avions pour savoir quelles données étaient à notre disposition pour construire le fichier KML demandé.

De plus, nous devions comprendre le KML pour savoir quoi faire de ces données et cela à été le plus compliqué. En effet, la documentation ne nous a pas énormément aidé. Ce qui nous a le plus aider était d’utiliser le convertisseur fourni pour transformer le .geojson en .kml.

Enfin, nous avons dû comprendre ce que nous retournais le convertisseur et faire quelques recherches en plus pour comprendre comment avoir juste les lignes de la frontière car le convertisseur ne faisaient pas les lignes du pays mais une forme pleine ce qui n’était pas ce que nous voulions. Nous avons donc dû faire plusieurs essaies pour trouver quelles balises utiliser pour que le rendu sur GoogleEarth nous convienne. Bien que cette méthode ne soit pas la plus efficace c’est celle qui nous a permis d’arriver au résultat voulu le plus facilement.

* Documentation KML
* Comprendre le geojson et quelles balises utiliser

# Problèmes connus

Dans les fichiers .kml générés par le convertisseur, la première balise (<kml>) possédait un attribut xmlns avec comme valeur l’adresse url d’un site internet. Nous n’avons pas réussi à ajouter cet attribut car, lors de l’exécution, cela jetait une exception. Le fichier kml généré fonctionne quand même malgré le manque de cet attributs.

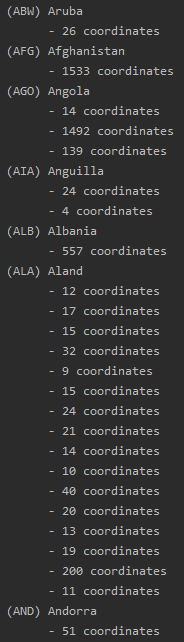
Nous avons aussi remarqué que le pays « Akrotiri Sovereign Base Area » avait pour abréviations « -99 » ce que nous avons trouvé bizarre. Mais après avoir rechercher ce pays dans le fichier .geojson, nous avons remarquer que c’est ce qui était noté donc cela ne semble pas posé de problème.

Etant donnée que notre programme semble bien fonctionner ce ne sont pas vraiment des disfonctionnement mais plus des zones d’ombre pour nous.

* Xmlns
* -99

# Affichages

Une copie d’écran de l’affichage lors du parsing du fichier geojson



Une copie d’écran représentative après le chargement de votre fichier KML dans l’application GoogleEarth :

# Apprentissages

Grace a ce laboratoire, nous avons pu apprendre de nouveaux formats de fichier dérivés de ceux vus en cours : le KML et le gesjson.

De plus, nous avons aussi pu apprendre a utiliser la librairie JSONSimple qui est assez intuitive.

Enfin, nous avons appris que le Portugal avait des îles. : )

* KML
* Le Portugal a des îles.
* Le format geojson
* JSONSimple

# Conclusion

Ce travail était assez amusant ? Oui je sais pas quoi mettre dans cette conclution…