

TP pymongo

pymongo est la bibliothèque permettant d'interagir avec une base de données MongoDB.

Activité 1 – Installation de pymongo

```
pip install pymongo
```

Activité 2 – Test de la connexion

Démarrer un serveur mongo, en utilisant docker ou une installation locale comme vu dans le premier TP.

```
docker run -it -p 28000:27017 --name mongo-exemple mongo:latest
```

Depuis un autre terminal, exécuter le code python suivant :

```
import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:28000/")

print(myclient.list_database_names())
```

Vérifier que la sortie affichée correspond bien à `show dbs` de mongosh

Activité 3 – Créer une base, une collection, insérer un document, faire une recherche

```
import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:28000/")

print(myclient.list_database_names())

mydb = myclient["mydatabase"]

collection = mydb["test_collection"]

collection.insert_one({"valeur": 1})

for doc in collection.find():

    print(doc)

collection.insert_many([{"valeur": 2}, {"valeur": 3}, {"valeur": 4}, {"valeur": 5}])

results = collection.find({ "valeur": { "$gt": 3 } })
```

Cœur de l'activité du TP4

Dans le dossier de TP, trois fichiers de type jsonl sont donnés.

Avec un programme python et une interface streamlit, concevoir une petite application qui :

- * permet de charger un fichier en utilisant un bouton de type `st.fileuploader`
- * ajoute le contenu du fichier dans une collection de la base « datasets » (qu'il aura fallu créer précédemment. La collection ajoutée sera construite à partir du nom du document (attention, pas d'espaces ni de tirets))
- * un widget de type `st.selectbox` permet de choisir une des collections présente dans la base
- * un champ de texte permet de saisir un critère de recherche. Le critère de recherche sera le dictionnaire passé à la méthode `find()`
- * quand on appuie sur un bouton « rechercher », on effectue la recherche considérée sur la collection sélectionnée, et on affiche les résultats renvoyés (soit directement, soit en transformant le résultat dans un pandas dataframe).