

Buenos días/tardes/noches. En este archivo voy a explicar cómo he llevado a cabo la migración de la base de datos del archivo .db a mongodb. Como podrá comprobar, lo estoy realizando en un jupyter notebook exportado a pdf, ya que me parece la mejor forma de compartir código y texto explicativo a la vez

Lo primero que hay que hacer es importar las bibliotecas convenientes. Sqlite3 nos permitirá establecer conexión con la base de datos .db. Pandas nos permitirá hacer consultas en lenguaje SQL a esta base de datos. Pymongo nos permitirá conectarnos a nuestro servidor local mongod y crear ahí una base de datos.

In [1]:

```
import sqlite3
from pandas import read_sql_query
from pymongo import MongoClient
```

El siguiente paso es obtener datos de la base de datos. Para tenerlo todo más ordenado, lo primero que haremos será hacer una consulta que nos permita obtener el nombre de todas las tablas. Una vez hecho esto, para cada tabla hacemos una consulta en la que pedimos toda la información. Todo esto lo guardamos en un diccionario de tal forma que la clave es el nombre de la tabla y el valor es una dataframe que contiene la información de esta tabla.

In [2]:

```
def read_sqlite(dbfile):
    with sqlite3.connect(dbfile) as dbcon:
        tables = list(read_sql_query("SELECT name FROM sqlite_master WHERE type='table';", dbcon)['name'])
        out = {tbl : read_sql_query(f"SELECT * from {tbl}", dbcon) for tbl in tables}
    return out
```

Una vez tenemos los datos en este formato, lo que hay que hacer es manipularlos de tal forma que cada entrada en cada una de las tablas aparezca como un diccionario de python, que luego meteremos como objeto (o documento) en la colección correspondiente de la base de datos Para ello creamos la siguiente función (que luego usaremos para cada entrada de la base de datos). Usando en id que asigna pandas, la función crea un diccionario (que pasaremos como objeto) de cada entrada en una tabla

In [3]:

```
def object_dict(table, id):
    res = dict()
    for field in table:
        res[field] = table[field][id]
    return res
```

Una vez tenemos estas dos funciones definidas, falta unir las piezas del rompecabezas. Contando con que tengamos el servidor mongod levantado en el puerto 127.0.0.1:27017, nos conectamos a él con pymongo. Usamos la función creada anteriormente read_sqlite para obtener los datos de la base de datos en un formato en el que se pueda trabajar cómodamente. Creamos una base de datos en MongoDB con el mismo nombre que la anterior (quitando el formato del archivo).

Después, iteramos por cada tabla de la base de datos .db para crear colecciones del mismo nombre en la base de mongo. En cada iteración, pasamos cada tabla a diccionario, y pasamos cada una de las entradas de esa tabla a diccionario gracias a la función que hemos creado anteriormente (object_dict). Estos diccionarios los metemos en una lista y los pasamos a mongoDB como objetos.

In [4]:

```
def db_to_mongo(db_name):
    client = MongoClient('mongodb://localhost:27017/') # Conectamos con el servidor MongoD
    old_db = read_sqlite(db_name) # Obtenemos un diccionario de dataframes de las tablas de la base de datos
    mongo_db_name = db_name.replace('.db', '')
    new_db = client[mongo_db_name] # Creamos una base de datos en nuestro servidor MongoD
    for table in old_db:
        table_dict = old_db[table].to_dict() # Pasamos cada tabla a diccionario
        objects = [object_dict(table_dict, id) for id in range(len(table_dict[list(table_dict.keys())[0]]))] # Convertimos cada entrada en objeto y los metemos en una lista de objetos
        collection = new_db[table] # Creamos en nuestra base de datos en mongod una colección del nombre de cada tabla
        collection.insert_many(objects) # Insertamos los objetos en la colección correspondiente
```

Ahora sólo queda ejecutar nuestra función con el nombre de la base de datos que queremos migrar a MongoDB. No nos olvidamos de tener el programa y la base de datos en la misma carpeta.

In [5]:

```
db_to_mongo('love4pets.db')
```

¡Ya tenemos nuestra base de datos migrada a MongoDB!