

Nombre: Nicole Zambrano.

Fecha: 18/08/2020.

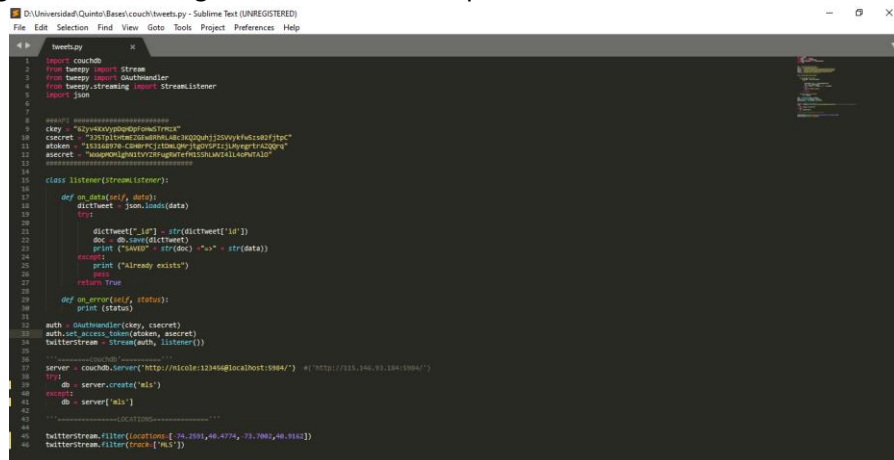
Prueba.

Twitts de Nueva York a Couch DB (newyork.py).

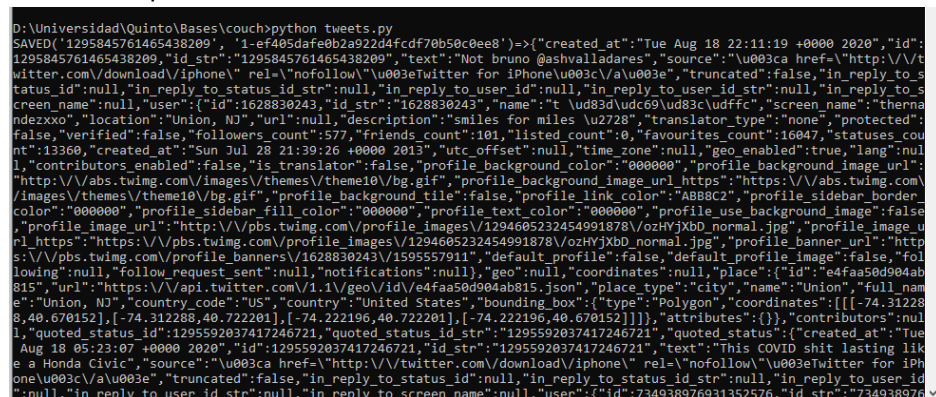
1. Acceder a Bounding Box para conseguir las coordenadas de Nueva York.



2. Agregamos en el código las coordenadas respectivas.



3. Corremos el script.



4. Corremos otro script con un tema diferente para hacer la réplica, en este caso utilizaré NBA.

```

D:\Universidad\Quinto\Bases\couch\tweets.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

tweets.py
1 import couchdb
2 from tweepy import Stream
3 from tweepy import OAuthHandler
4 from tweepy.streaming import StreamListener
5 import json
6
7
8 #####
9 ckey = "Gdy4Kx0y0p0QWfma5TmUx"
10 csecret = "315TPltHME2G6W8HhLAbC3KQ2QWhj25Vvykfw5s2fjpc"
11 atoken = "153168378-CB8eRfCjEDmLQwJtGOVSPiI:LMYegRtRAZQ0q"
12 asecret = "h0apn0lghNltV2RfRagMeFh15hLwY14140WtAl0"
13 #####
14
15 class listener(StreamListener):
16
17     def on_data(self, data):
18         dicttweet = json.loads(data)
19         try:
20             dicttweet['_id'] = str(dicttweet['id'])
21             doc = db.save(dicttweet)
22             print ("SAVED" + str(doc) + "\n" + str(data))
23         except:
24             print ("Already exists")
25             pass
26         return True
27
28     def on_error(self, status):
29         print (status)
30
31
32 auth = OAuthHandler(ckey, csecret)
33 auth.set_access_token(atoken, asecret)
34 twitterstream = Stream(auth, listener)
35
36 #####CouchDB#####
37 server = couchdb.Server('http://nicole:123456@localhost:5984/') #('http://135.146.93.104:5984/')
38 try:
39     db = server.create('nba')
40 except:
41     db = server['nba']
42
43 #####LOCATION#####
44 #twitterstream.filter(locations=[-74.2591,-40.4774,-73.7802,40.5162])
45 twitterstream.filter(track=['nba'])
46

```

- En nuestro CouchDB procedemos a hacer la réplica entre mls-nba y viceversa.

mls	32.8 KB	14	No
nba	358.7 KB	123	No

Job Configuration

Source

Type: Local database
Name: mls
Authentication: Username and password
nicole

Target

Type: Existing local database
Name: nba
Authentication: Username and password
nicole

Options

Replication type: One time

Job Configuration

Source

Type: Local database
Name: nba
Authentication: Username and password
nicole

Target

Type: Existing local database
Name: mls
Authentication: Username and password
nicole

Options

Replication type: One time

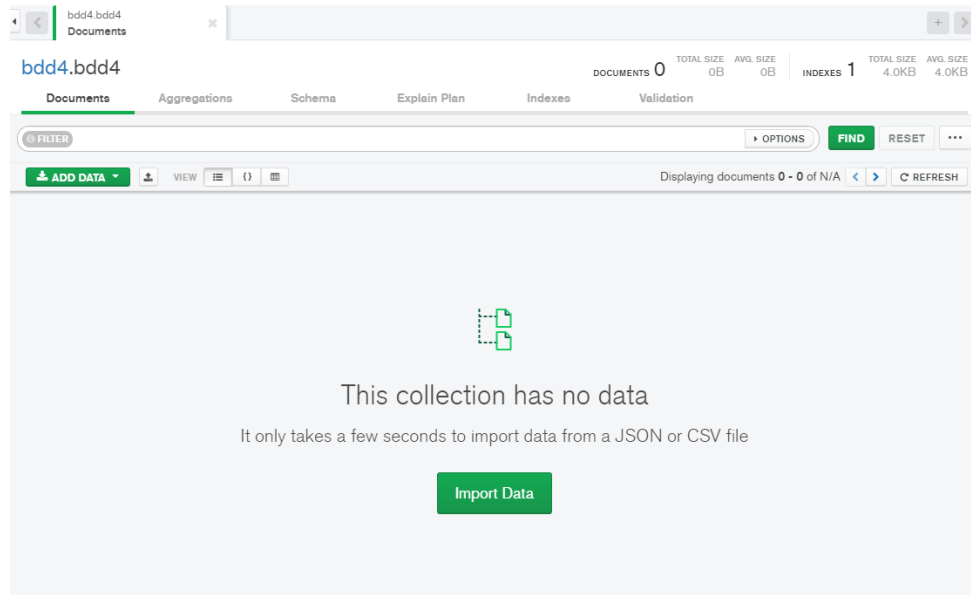
- Una vez realizadas las réplicas, tendremos la misma cantidad de documentos en ambas bases de datos, en este caso 73.

Tiktok a MongoDB.

1. Ponemos el comando

```
D:\Universidad\Quinto\Bases\couch>tiktok-scraper user hannahstocking -n 300 -t csv
CSV path: D:\Universidad\Quinto\Bases\couch\hannahstocking_1597789679340.csv
```

2. Vamos a MongoDB local y creamos una nueva base de datos con su respectiva colección para importar los datos desde nuestro csv, en este caso la base será bdd4.



3. Importamos el CSV y lo delimitamos por comma.

D:\Universidad\Quinto\Bases\couch\hannahstocking_1597789679340.csv [BROWSE](#)

Select Input File Type

☐ JSON ☒ CSV

Options

Select delimiter: **COMMA**

☒ Ignore empty strings

☐ Stop on errors

Specify Fields and Types

	id	text
	String	text
1	6862149705875361030	I CANT BELIEVE THAT HAPPENED!!! 😭😭 I swear I'm a good barber. WAI
2	6861768559891074310	How many Military Sit-Ups can you do? 😊 (wait for it) thanks @desto
3	6861444552939687174	THANK YOU FOR 15.6945 MILLION! 🥰🥰 I love you all so much 🥰🥰
4	6861112310379941125	Student films be like... 🤔 (I'm still laughing btw) 🤔🤔🤔🤔🤔
5	6860944721628450054	Not today boys 🤔 idea by @thot_ting_noi
6	6860681650632772870	I hate when ppl assume things 🤔 @adamw @destormpower @official_j
7	6860311015448464645	Whos down to play? 🤔🤔
8	6860219858609540357	I WON! 🥰🥰 check out @juanpazurita 's page to see who won the cham
9	6859944619220782342	Share this vid w/ 3 people & DONT SAY ANYTHING 🤔🤔🤔 vid by @anwar
10	6859895851691052293	My transformation into #LOBA from #apexlegends 🤔🤔 #cosplay outfit

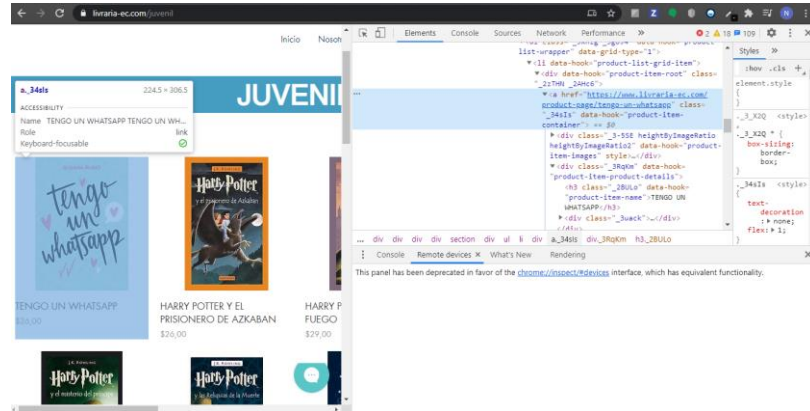
Import completed 300 (100%)

[DONE](#)

Web Scrapping (livraria.py).

1. Seleccionar las páginas para hacer Web Scrapping e inspeccionarlas, para buscar el nombre de una clase sobre la cual vamos a iterar para almacenarla en nuestra base de datos.

<https://www.livreria-ec.com/juvenil>



Clase: `_34Is`.

2. En el código ubicaremos el link de nuestra página, el nombre de la clase, pymongo y el nombre de nuestra base de datos y la respectiva colección en este caso será librería la base de datos y la colección libreria.

```
import requests
from bs4 import BeautifulSoup
# Import MongoClient from pymongo so we can connect to the database
from pymongo import MongoClient

if __name__ == '__main__':
    # Instantiate a client to our MongoDB instance
    db_client = MongoClient()
    libreria = db_client.livreria
    libreria = libreria.posts

    response = requests.get("https://www.livreria-ec.com/juvenil")
    soup = BeautifulSoup(response.content, "lxml")

    post_titles = soup.find_all("a", class_="_34Is")

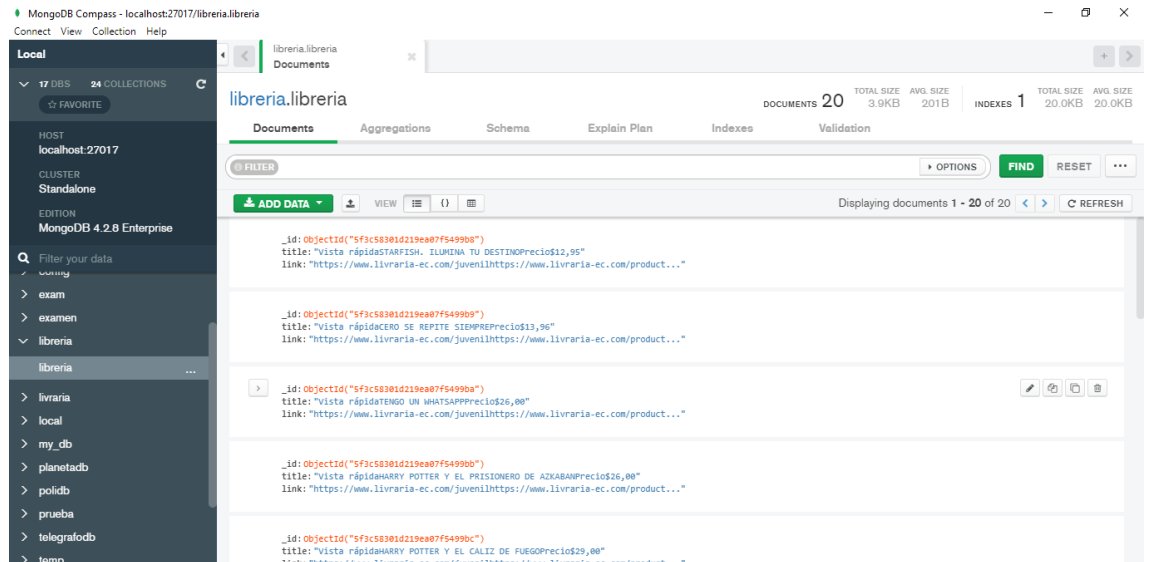
    extracted = []
    for post_title in post_titles:
        extracted.append({
            'title': post_title.text,
            'link': "https://www.livreria-ec.com/juvenil" + post_title['href']
        })

    for post in extracted:
        if db_client.livreria.livreria.find_one({'link': post['link']}) is None:
            print("Found a new listing at the following url: ", post['link'])
            db_client.livreria.livreria.insert(post)
```

3. Ejecutamos el archivo .py

```
D:\Universidad\Quinto\Bases\couchpython livreria.py
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/s
tarfish-ilumina-tu-destino
livreria.py:28: DeprecationWarning: insert is deprecated. Use insert_one or insert_many instead.
  db_client.livreria.livreria.insert(post)
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/c
ero-se-repite-siempre
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/t
engo-un-whatsapp
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/h
arry-potter-y-el-prisionero-de-azkaban
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/h
arry-potter-y-el-caliz-de-fuego
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/h
arry-potter-y-la-orden-del-fenix
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/h
arry-potter-y-el-misterio-del-principe
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/h
arry-potter-y-las-reliquias-de-la-muerte
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/a
-dos-metros-de-ti
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/t
rono-destrozado-relatos-de-la-reina-roja
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/c
orona-cruel-la-reina-roja-relatos
Found a new listing at the following url: https://www.livreria-ec.com/juvenilhttps://www.livreria-ec.com/product-page/l
a-jaula-del-rey-todo-ardera-la-reina-roja-iii
```

4. Comprobamos el MongoDB.



Donald Trump a MongoDB (harvesting_tweets_mongodb.py).

1. En el siguiente script, filtrar los tweets que sean relacionados con Donald Trump y almacenarlos en una base de datos en MongoDB, en este caso será en la misma base de datos de webscrapping en la colección bdd3.

```
import pymongo
from tweepy import Stream
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener
import json

ckey = "6Zyv4XvYpDqHDPFoHwSTrMzX"
csecret = "3J5TpiHtmEZGEW8RRLABc3KQ2Quhjj2SVvykfw5zs02fjtpc"
atoken = "153168970-C8H0rPCjztdMLQMrjtgoYSPizJLMyegtrAZQQrq"
asecret = "WddpMOM1ghN1tVYRFugRNTefM1SShLWVI41L4oPWTAlO"

class listener(StreamListener):

    def on_data(self, data):
        dictTweet = json.loads(data)
        try:
            dictTweet["_id"] = str(dictTweet['id'])
            doc = mycol.insert_one(dictTweet)
            print("SAVED" + str(doc) + ">" + str(data))
        except:
            print("Already exists")
            pass
        return True

    def on_error(self, status):
        print(status)

auth = OAuthHandler(ckey, csecret)
auth.set_access_token(atoken, asecret)
twitterStream = Stream(auth, listener())

'''Mongo'''

myclient = pymongo.MongoClient("mongodb://localhost:27017")
mydb=myclient["libreria"]
mycol = mydb["bdd3"]

'''=====LOCATIONS====='''

twitterStream.filter(track=['Donald Trump'])
```

2. Correr el archivo .py

3. Comprobar en MongoDB.



1. Exportamos en nuestro MongoDB las colecciones de nuestra base de datos librería.

Export Collection *libreria.bdd3*

☐

Export query with filters — 90 results (Recommended)

```
db.bdd3.find(
  {}
)
```

☒

Export Full Collection

CANCEL

SELECT FIELDS

Export Collection libreria.libreria

Select Export File Type

JSON

CSV

Output

D:\Universidad\Quinto\Bases\couch\libreria.csv

BROWSE

< BACK

CANCEL

EXPORT

- Posteriormente vamos a la base de datos bdd4 y en la colección ya creada importamos los datos previamente exportados desde librería.

Import to Collection bdd4

Select File

D:\Universidad\Quinto\Bases\couch\bdd3.csv

BROWSE

Select Input File Type

JSON

CSV

Options

Select delimiter

COMMA

☒ Ignore empty strings
 ☐ Stop on errors

Specify Fields and Types

	<input checked="" type="checkbox"/> id String	<input checked="" type="checkbox"/> created_at String	<input checked="" type="checkbox"/> id String	<input checked="" type="checkbox"/> id_str String
1	1295854962623340546	Tue Aug 18 22:47:53 +0000 2020	1295854962623340546	129585496262334
2	1295854962984079368	Tue Aug 18 22:47:53 +0000 2020	1295854962984079368	129585496298407
3	1295854963864081283	Tue Aug 18 22:47:53 +0000 2020	1295854963864081283	129585496386408
4	1295854963692900352	Tue Aug 18 22:47:53 +0000 2020	1295854963692900352	129585496369290
5	1295854963818725377	Tue Aug 18 22:47:53 +0000 2020	1295854963818725377	129585496381872

Import completed 90 (100%)

DONE

Select File

D:\Universidad\Quinto\Bases\couch\libreria.csv

BROWSE

Select Input File Type

JSON

CSV

Options

Select delimiter

COMMA

☒ Ignore empty strings
 ☐ Stop on errors

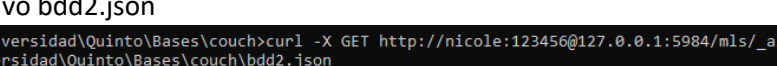
Specify Fields and Types

	<input checked="" type="checkbox"/> id String	<input checked="" type="checkbox"/> title String
1	5f3c58381d219ea07f5499b8	Vista rápidaSTARFISH. ILUMINA TU DESTINOPrecio\$12,95
2	5f3c58381d219ea07f5499b9	Vista rápidaCERO SE REPITE SIENPREPrecio\$13,96
3	5f3c58381d219ea07f5499ba	Vista rápidaTENGO UN WHATSAPPPrecio\$26,00
4	5f3c58381d219ea07f5499bb	Vista rápidaHARRY POTTER Y EL PRISIONERO DE AZKABANPrecio\$26,00
5	5f3c58381d219ea07f5499bc	Vista rápidaHARRY POTTER Y EL CALIZ DE FUEGOPrecio\$29,00
6	5f3c58381d219ea07f5499bd	Vista rápidaHARRY POTTER Y LA ORDEN DEL FENIXPrecio\$30,00
7	5f3c58381d219ea07f5499be	Vista rápidaHARRY POTTER Y EL MISTERIO DEL PRINCIPEPrecio\$30,00
8	5f3c58381d219ea07f5499bf	Vista rápidaHARRY POTTER Y LAS RELIQUIAS DE LA MUERTEPrecio\$30,00
9	5f3c58381d219ea07f5499c0	Vista rápidaA DOS METROS DE TIPrecio\$26,00
10	5f3c58381d219ea07f5499c1	Vista rápidaTRONO DESTROZADO (RELATOS DE LA REINA ROJA)Precio\$1

Import completed 20 (100%)

DONE

CouchDB a MongoDB.

- Con el comando `curl -X GET http://nicole:123456@127.0.0.1:5984/mls/_all_docs?include_docs=true` exportamos los datos almacenados en este caso en nuestra base de datos mls y lo guardamos en un archivo `bdd2.json`


```
D:\Universidad\Quinto\Bases\couch>curl -X GET http://nicole:123456@127.0.0.1:5984/mls/_all_docs?include_docs=true
D:\Universidad\Quinto\Bases\couch\bdd2.json
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100    41    100    41    0    0    41      0  0:00:01 --:--:--  0:00:01  262
```
- El archivo recién creado lo importamos en MongoDB en nuestra base de datos `bdd4` y en la colección `bdd4`.

The screenshot shows the MongoDB Compass interface during a document import process. At the top, the path "Import To Collection bdd4.bdd4" is displayed. Below this, the "Select File" section shows the file "D:\Universidad\Quinto\Bases\couch\nba.json" selected, with a "BROWSE" button. The "Select Input File Type" section has "JSON" highlighted. In the "Options" section, "Ignore empty strings" is checked, and "Stop on errors" is unchecked. The "Specify Fields and Types" section shows four columns: "id" (String), "key" (String), "value.rev" (String), and "doc._id" (String). A table below shows six rows of sample data. At the bottom, the "Importing documents..." status bar indicates "0 (0%)" completion. On the right side of the interface, there are "CANCEL" and "IMPORTING" buttons. The left sidebar shows the database structure with "bdd4.bdd4" selected under "COLLECTIONS". The main panel displays the "Documents" tab for "bdd4.bdd4", showing a list of documents with fields like "_id", "created_at", "lat", "lon", "source", "truncated", "is_reply", "is_status", "is_reply_to_status", "is_reply_to_user_id", "is_reply_to_reply_id", "is_reply_to_screen_name", "user", "geo", "coordinates", "place", "contributors", "retweet_status", "quote_status", "reply_count", "retweet_count", "favorite_count", and "entities".

MongoDB local a Atlas.

1. Exportamos el json desde nuestra base de datos bdd4.

Export Collection bdd4.bdd4

Select Export File Type

JSON CSV

Output

D:\Universidad\Quinto\Bases\couch\bdd4.json

BROWSE

Export completed 485/485 (100%)

< BACK CLOSE SHOW FILE

2. Acceder al cluster y creamos una nueva base de datos bdd5.

MongoDB Compass - cluster0.mongodb.net/2017/bdd5.bdd5

Connect View Collection Help

Local

bdd5.bdd5 Documents

Documents Aggregations Schema Explain Plan Indexes Validation

ADD DATA VIEW

Displaying documents 0 - 0 of N/A

Refresh

This collection has no data

It only takes a few seconds to import data from a JSON or CSV file

Import Data

3. Importamos nuestro archivo json.

Import To Collection bdd5.bdd5

Select File

D:\Universidad\Quinto\Bases\couch\bdd4.json

BROWSE

Select Input File Type

JSON CSV

Options

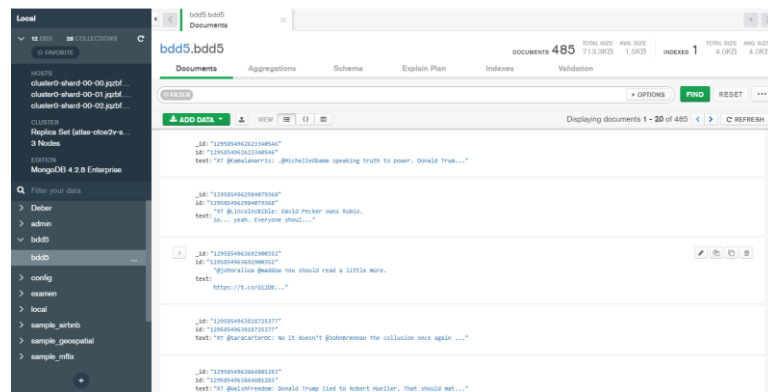
☒ Ignore empty strings

☐ Stop on errors

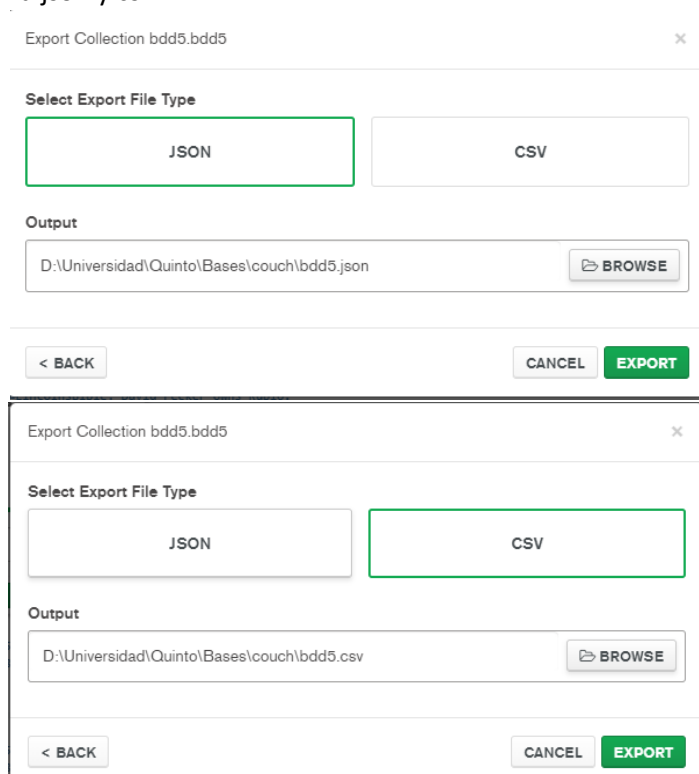
Specify Fields and Types

	<input checked="" type="checkbox"/> id ObjectID	<input checked="" type="checkbox"/> id String	<input checked="" type="checkbox"/> text string
1	5f3c56bce9aa5a04ac7a7c63	6862149705875361030	I CANT BELIEVE THAT HAPPENED!!! 🤔😭 I SW
2	5f3c56bce9aa5a04ac7a7c64	6861768559891074310	How many Military Sit-Ups can you do? 🤔 I
3	5f3c56bce9aa5a04ac7a7c65	6861444552939687174	THANK YOU FOR 15.6945 MILLION! 🤔😭 I lov
4	5f3c56bce9aa5a04ac7a7c66	6861112310379941125	Student films be like... 🤔 (I'm still la
5	5f3c56bce9aa5a04ac7a7c67	6860944721628450054	Not today boys 🤔 idea by @that_ting_noi
6	5f3c56bce9aa5a04ac7a7c68	6860681650632772870	I hate when ppl assume things 🤔😭 @adamv
7	5f3c56bce9aa5a04ac7a7c69	6860311015448464645	Whos down to play? 🤔🤔

CANCEL IMPORT



4. Exportar a json y csv.



Cadena de conexión:

mongodb+srv://nicole:0529nicoleZ@cluster0.jqzbf.mongodb.net/test?authSource=admin&replicaSet=atlas-otoe2v-shard-0&readPreference=primary&appName=MongoDB%20Compass&ssl=true