nicio**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**FACULTAD DE ELECTRICIDAD Y COMPUTACIÓN**

**Materia:** Sistemas de Base de Datos Avanzados **Paralelo:** 2

**Docente:** Ing. Vanessa Ivonne Echeverria Barzola

**Grupo:** 4

**Integrantes:**

* Dayse Joselyne Maroto Lema

**Proyecto del 1 Parcial**

**Enlace de Github:** <https://github.com/dmaroto98/MapReduce_MongoDB_ProyectoBDA.git>

**DATASET:** <https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks>

# Descripción del Dataset

El archivo "data.csv" contiene más de **170.000** canciones recopiladas de Spotify Web API. Contiene 19 columnas detalladas a continuación:

## Primary:

* id (Id of track generated by Spotify) **column** **9**

## Numerical:

Los datos mostrados a continuación son de tipo numérico.

* valence (Ranges from 0 to 1) **column** **1**
* year (Ranges from 1921 to 2020) **column** **2**
* acousticness (Ranges from 0 to 1) **column** **3**
* danceability (Ranges from 0 to 1) **column**  **5**
* duration\_ms (Integer typically ranging from 200k to 300k) **column**  **6**
* energy (Ranges from 0 to 1) **column** **7**
* instrumentalness (Ranges from 0 to 1) **column** **10**
* liveness (Ranges from 0 to 1) **column** **12**
* loudness (Float typically ranging from -60 to 0) **column** **13**
* popularity (Ranges from 0 to 100) **column** **16**
* speechiness (Ranges from 0 to 1) **column 18**
* tempo (Float typically ranging from 50 to 150) **column 19**

## ****Dummy****:

Los datos mostrados a continuación son de tipo boleano.

* explicit (0 = No explicit content, 1 = Explicit content) **column 8**
* mode (0 = Minor, 1 = Major) **column** **14**

## ****Categorical****:

Los datos mostrados a continuación son de tipo cadena.

* artists (List of artists mentioned) **column** **4**
* key (All keys on octave encoded as values ranging from 0 to 11, starting on C as 0, C# as 1 and so on…) **column** **11**
* name (Name of the song) **column** **15**
* release\_date (Date of release mostly in yyyy-mm-dd format, however precision of date may vary) **column 17**

# Descripción del Código

Para las operaciones CRUD se utilizó la librería de pymongo que nos proveé la herramienta de python [3].

## Create

Para la demostración de la inserción de datos se utilizaron 2 datos de prueba.

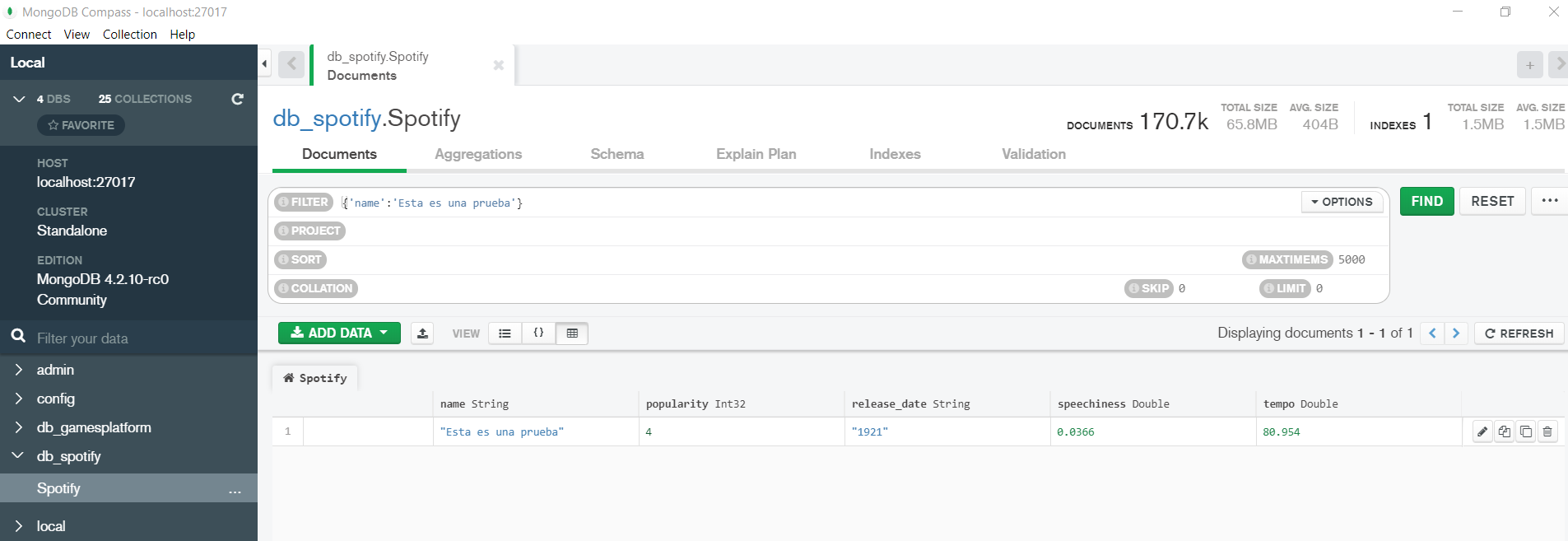
db[collectionName].insert\_many([

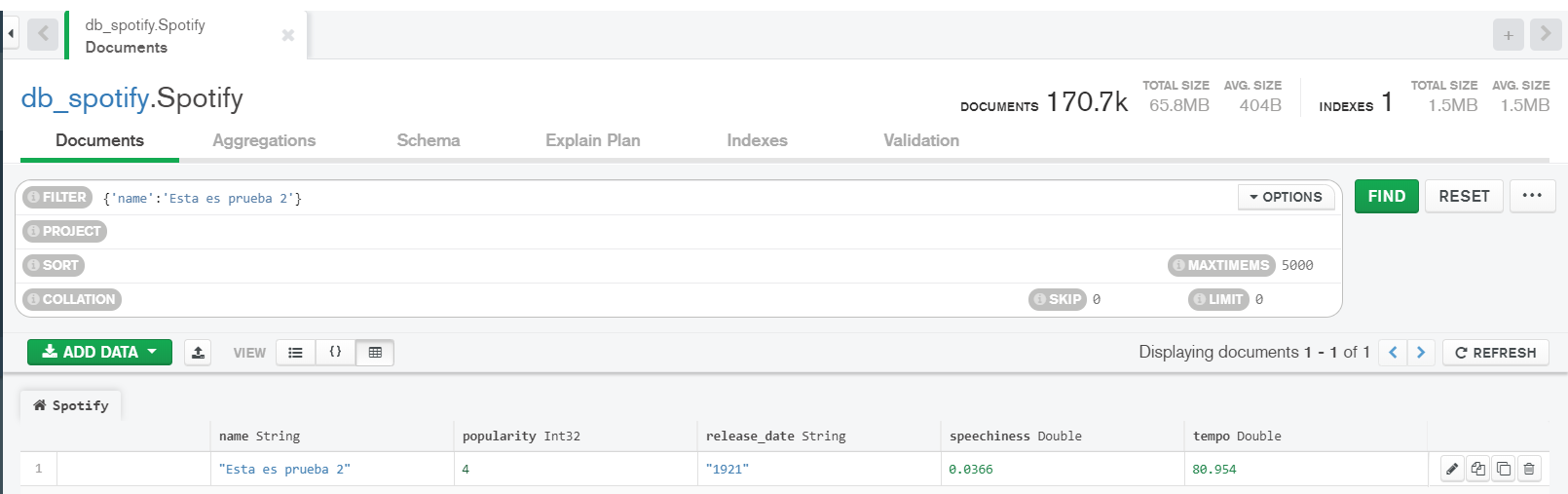
    {"valence":0.0594,"year":2020,"acousticness":0.232,"artists":"['Camila Cabello', 'Shawn Mendez']","danceability":0.99,"duration\_ms":74512,"energy":0.21100000000000002,"explicit":0,"id":"4BJqT0PrAfrxzMOxytFOIz","instrumentalness":0.878,"key":10,"liveness":0.665,"loudness":-20.096,"mode":1,"name":"Esta es una prueba","popularity":4,"release\_date":"1921","speechiness":0.0366,"tempo":80.954},

    {"valence":0.0594,"year":2020,"acousticness":0.562,"artists":"['Lauren Jauregui', 'Steven Aoki']","danceability":0.75,"duration\_ms":831667,"energy":0.21100000000000002,"explicit":0,"id":"4BJqT0PrAfrxzMOxytFOIz","instrumentalness":0.878,"key":10,"liveness":0.665,"loudness":-20.096,"mode":1,"name":"Esta es prueba 2","popularity":4,"release\_date":"1921","speechiness":0.0366,"tempo":80.954}

])

A continuación, se puede observar la evidencia de la inserción de datos en la base de datos a través de la herramienta de MongoDB Compass [2].



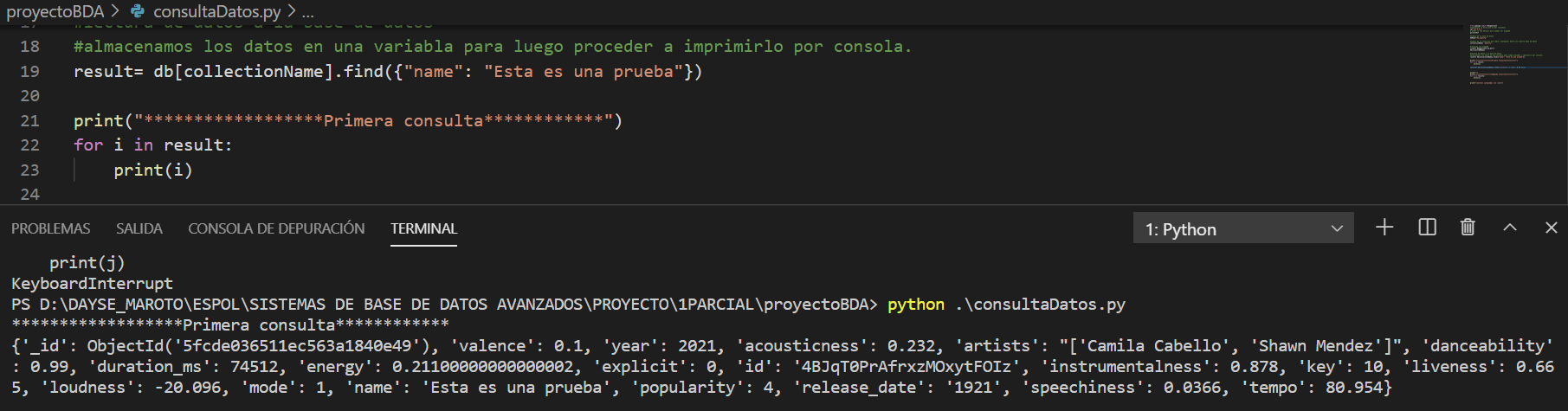


## Read

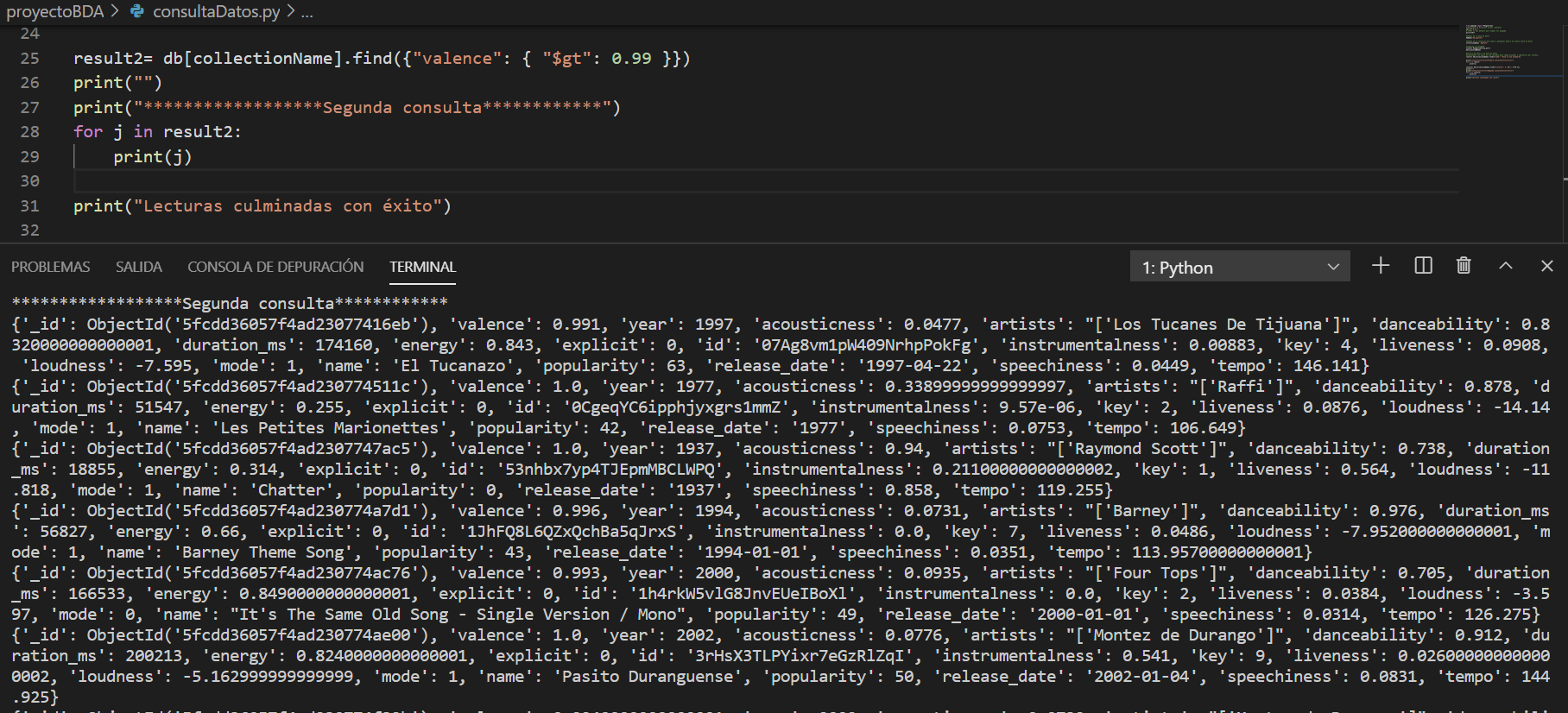
Para la demostración de la consulta de datos se utilizaron 2 datos de prueba.

Para poder realizar consultas en la base de datos MongoDB se debe de utilizar el comando find, al cual se le puede ingresar un query para una búsqueda en específico.

Aquí se extrajo la data cuyo nombre es “Esta es una prueba” y se lo imprime por consola.

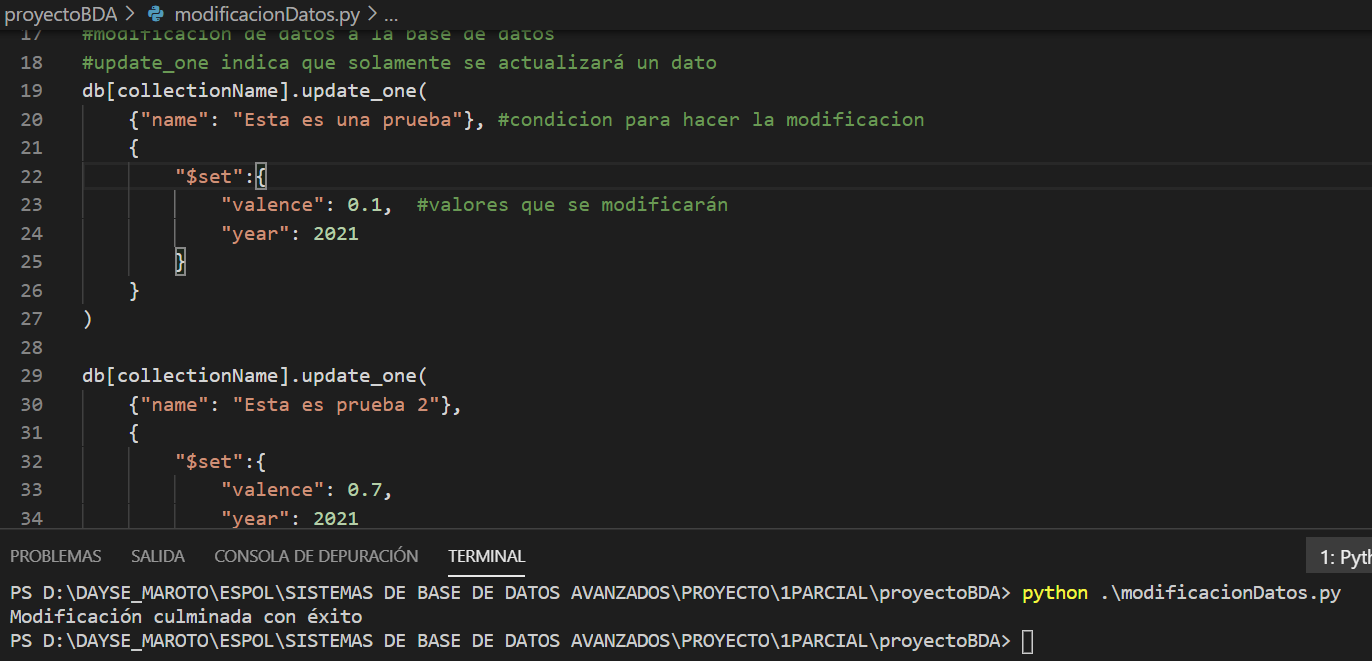


Aquí se extrajo la data cuyos valores de “valence” sean mayores a 0.99 y se los imprime por consola [4].



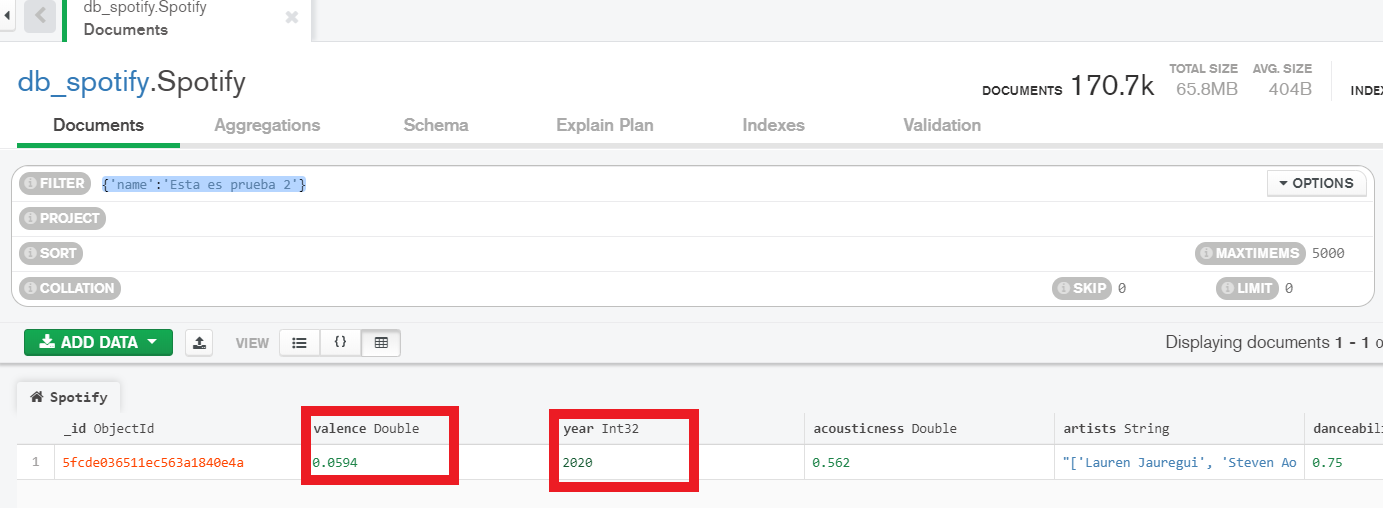
## Update

Para la demostración de la modificación de datos se utilizaron 2 datos de prueba.

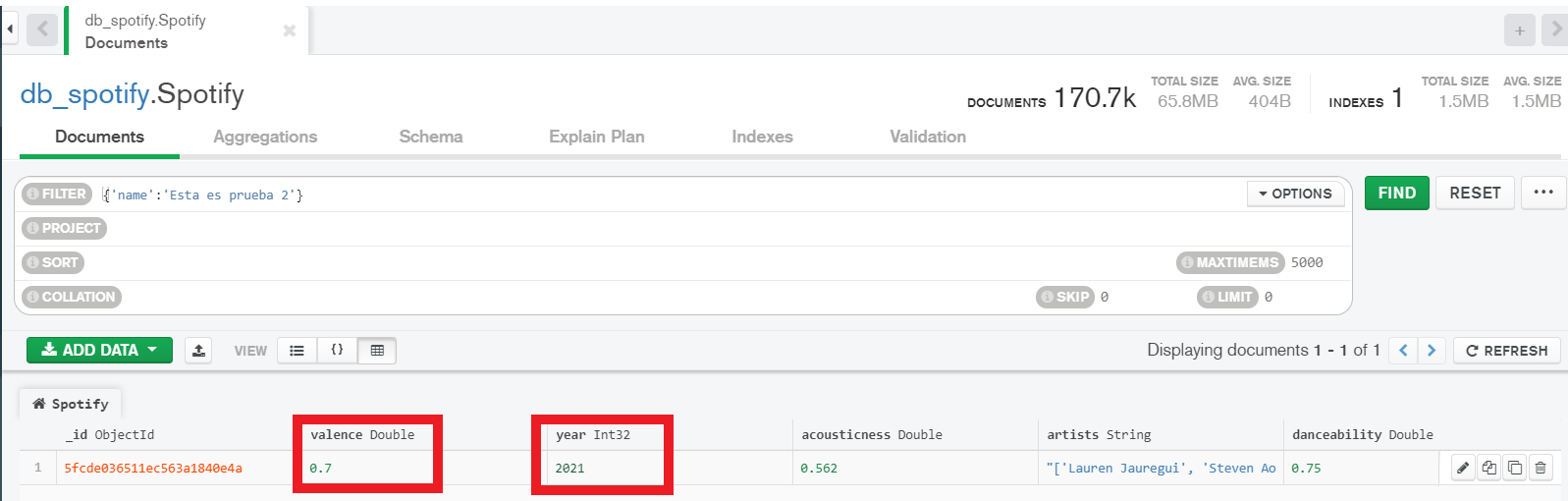


A continuación, se puede observar la evidencia de la modificación de datos en la base de datos a través de la herramienta de MongoDB Compass.

**Antes**



**Después**

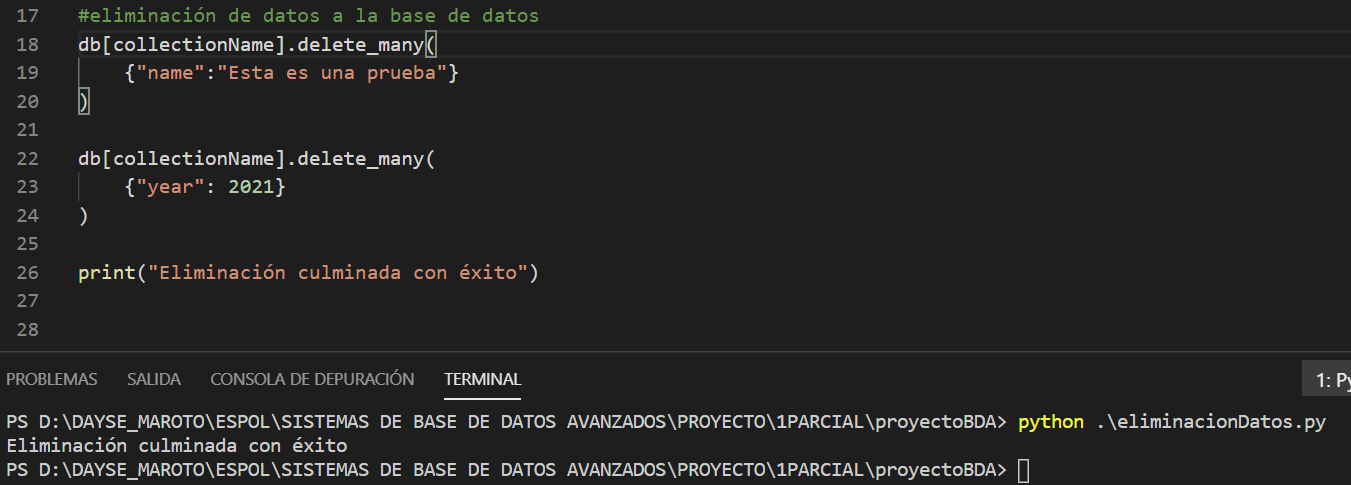
****

## Delete

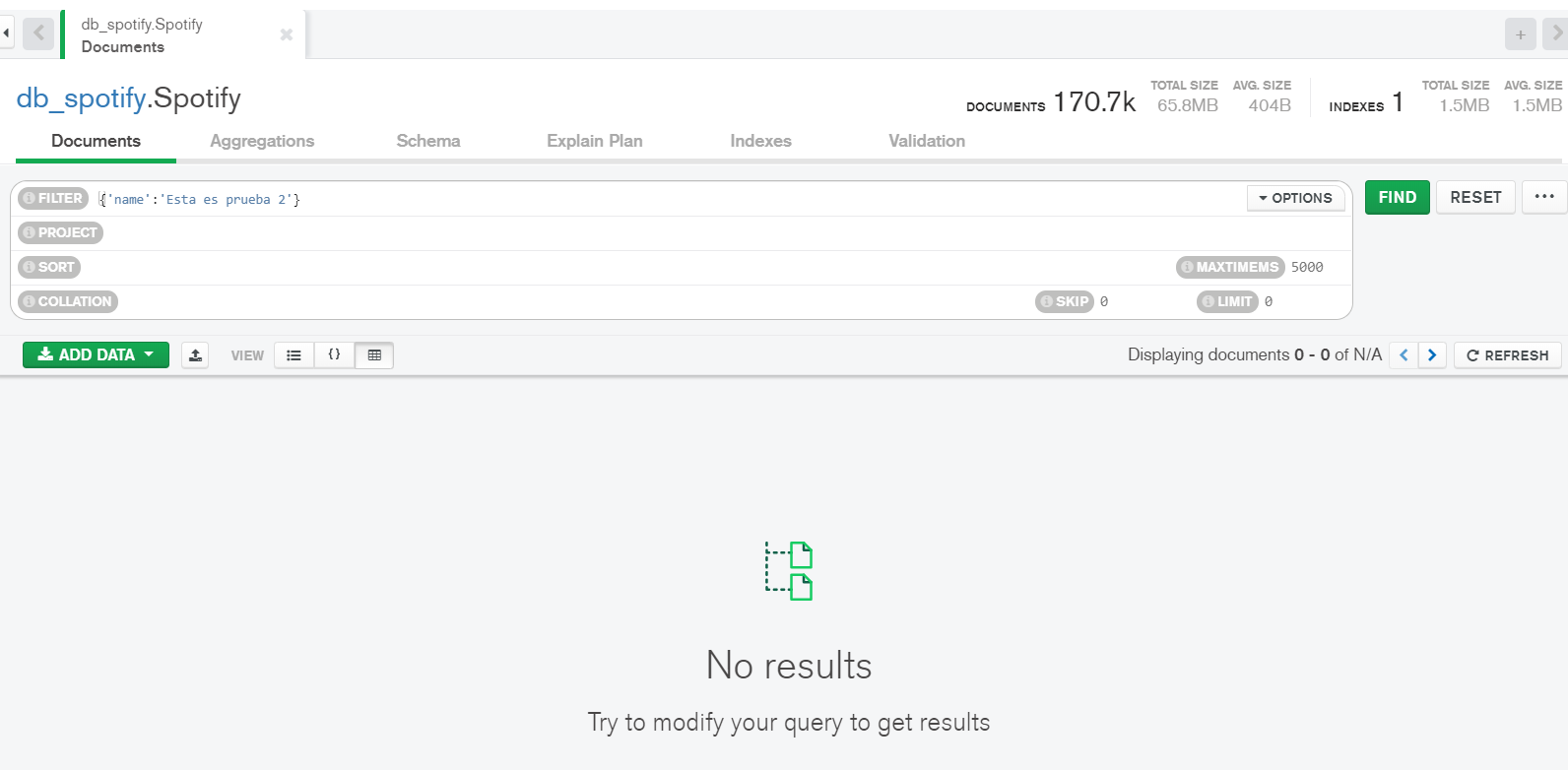
Para la demostración de la eliminación de datos se utilizaron los 2 datos creados anteriormente.

En la primera eliminación observamos que se eliminan los datos que en el campo “name” es igual a “Esta es una prueba”

En la segunda eliminación observamos que se eliminarán todos los datos que sean del año 2021.



Como evidencia observamos que ya no nos arroja datos con la consulta anteriormente utilizada.



# Descripción de la tarea a realizar con map-reduce.

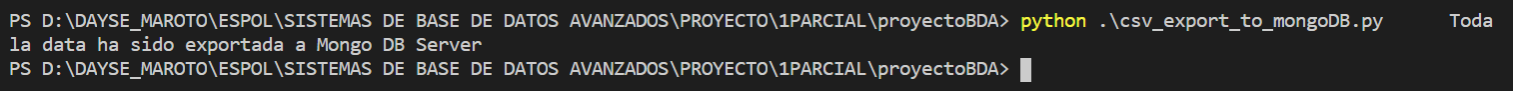
Utilizando la base de datos de Spotify, es conocer la cantidad de músicas que fueron lanzadas en el 2020, con una popularidad mayor a 30 y además que tenga el porcentaje de bailable mayor a 0.8, con la finalidad de conocer cuáles fueron los hits más bailables del 2020.

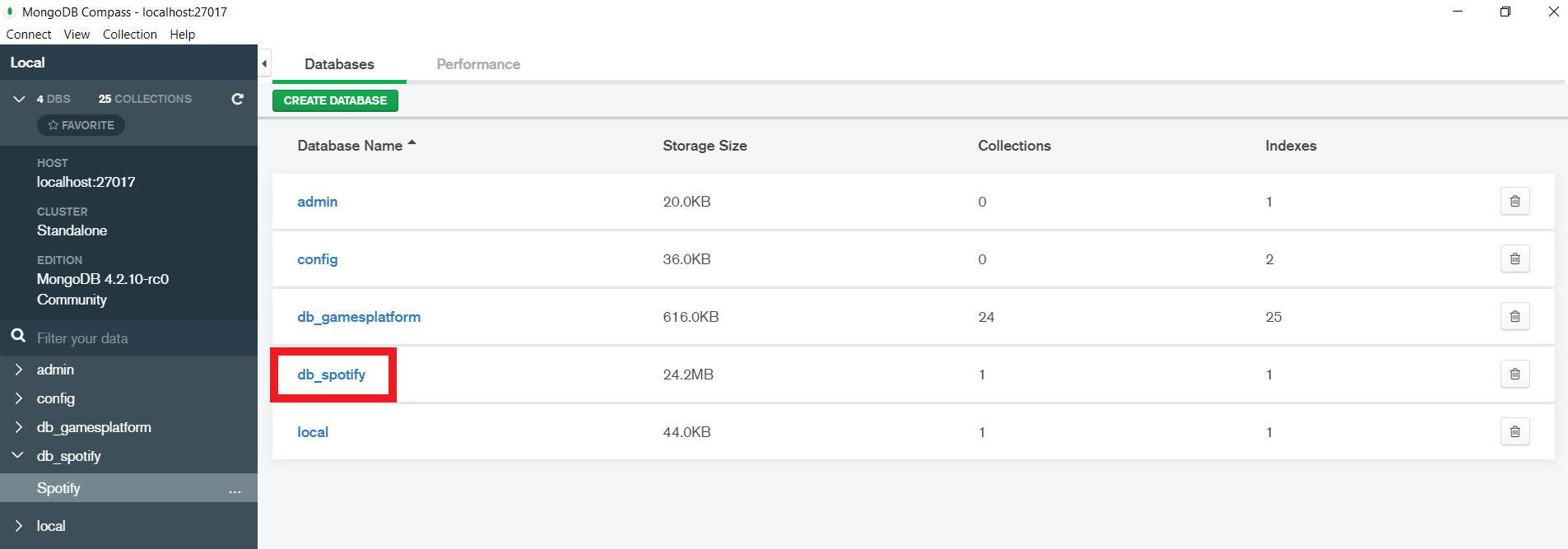
# Descripción de la implementación realizada

Se utilizó el lenguaje de programación Python para el proyecto, en conjunto con la base de datos no relacional MongoDB.

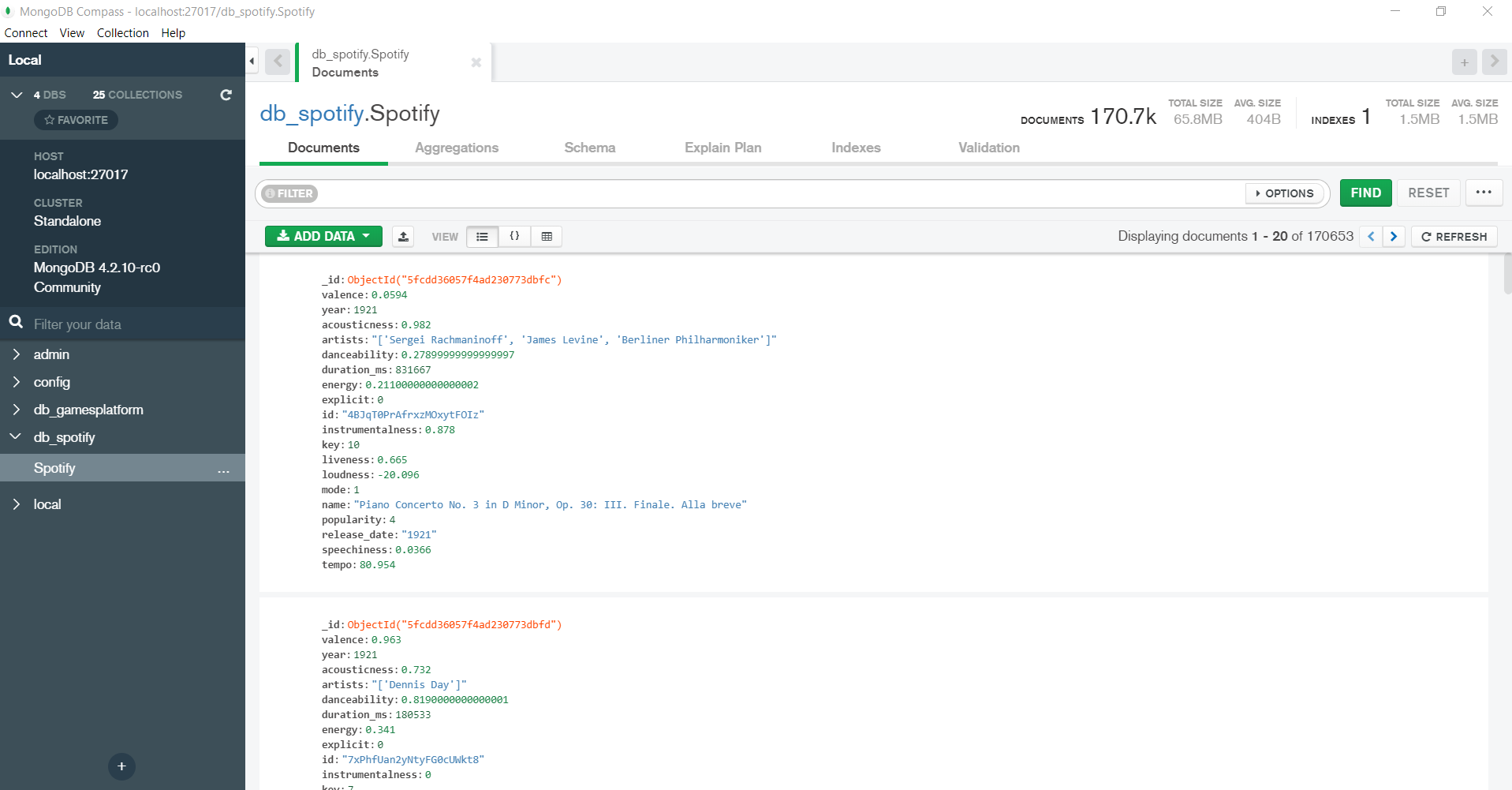
# Descripción de los resultados obtenidos.

Exportación de los datos del archivo data.csv a una base de datos de mongoDB [1].





Se creó el documento Spotify, el cual representa cada fila del archivo data.csv



# Bibliografía

[1]

soumilshah1995, “Export your Csv File Data to MongoDb using Python,” *YouTube*. Dec. 31, 2019, Accessed: Dec. 07, 2020. [Online]. Available: https://www.youtube.com/watch?v=CkckKPNbics&list=LL&index=1.

[2]

Redes Plus, “MongoDB04Como realizar Consultas en Compass de forma gráfica,” *YouTube*. Apr. 27, 2020, Accessed: Dec. 07, 2020. [Online]. Available: https://www.youtube.com/watch?v=fl\_3Jt5MIqo.

‌

[3]

Craftech, “Mongodb Crud operations using pymongo,” *YouTube*. Apr. 06, 2019, Accessed: Dec. 07, 2020. [Online]. Available: https://www.youtube.com/watch?v=orWal92Y-dU.

‌

[4]

“$gt — MongoDB Manual,” *Mongodb.com*, 2020. https://docs.mongodb.com/manual/reference/operator/query/gt/index.html (accessed Dec. 07, 2020).

‌