

Bases de datos NoSQL

Práctica 1

Daniel Pérez Efremova

Aclaraciones

Para realizar la práctica se ha utilizado la consola de MongoDB en la máquina virtual del curso.

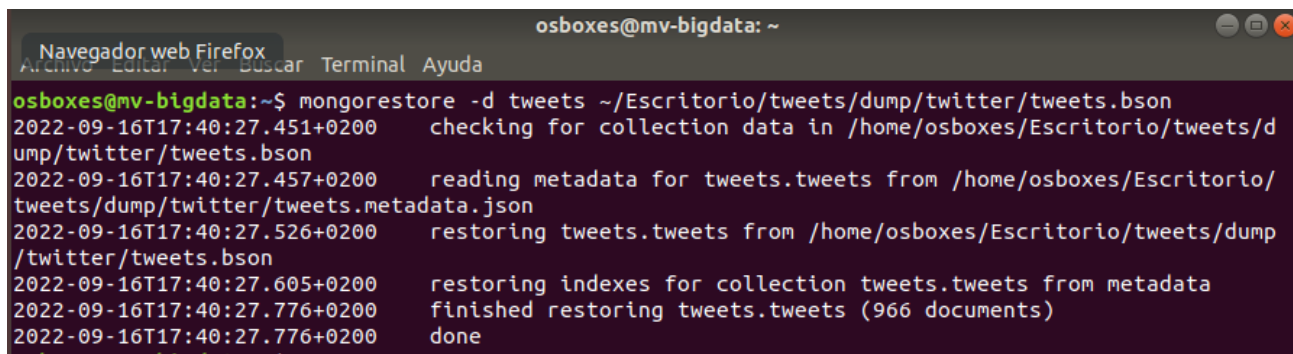
Ejercicios

1. (1 punto) Una vez descargado el fichero, debemos importarlo a MongoDB. Enumera los pasos que has tenido que ejecutar hasta que finalmente consigas tener dicho archivo cargado.

Se cargan los datos mediante la instrucción:

mongorestore -d tweets path

pasando la ruta al fichero .bson provisto en el curso. La herramienta permite cargar datos a la instancia mongod, ya sea una base de datos en formato binario (dump) o formato estándar de ficheros .bson.



```
osboxes@mv-bigdata: ~  
Navegador web Firefox  
Archivo Editor Ver Buscar Terminal Ayuda  
osboxes@mv-bigdata:~$ mongorestore -d tweets ~/Escritorio/tweets/dump/twitter/tweets.bson  
2022-09-16T17:40:27.451+0200 checking for collection data in /home/osboxes/Escritorio/tweets/dump/twitter/tweets.bson  
2022-09-16T17:40:27.457+0200 reading metadata for tweets.tweets from /home/osboxes/Escritorio/tweets/dump/twitter/tweets.metadata.json  
2022-09-16T17:40:27.526+0200 restoring tweets.tweets from /home/osboxes/Escritorio/tweets/dump/twitter/tweets.bson  
2022-09-16T17:40:27.605+0200 restoring indexes for collection tweets.tweets from metadata  
2022-09-16T17:40:27.776+0200 finished restoring tweets.tweets (966 documents)  
2022-09-16T17:40:27.776+0200 done
```

Fig 1. Creación de base de datos y colección

En la Figura 2 se ve la comprobación de la creación.

```
osboxes@mv-bigdata: ~
Archivo Editar Ver Buscar Terminal Ayuda
osboxes@mv-bigdata:~$ mongo
MongoDB shell version v4.0.12
connecting to: mongodb://127.0.0.1:27017/?gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("53c40698-92fa-4e37-90f0-6d2b99fa03ce") }
MongoDB server version: 4.0.12
Server has startup warnings:
2022-09-13T23:06:49.306+0200 I STORAGE [initandlisten]
2022-09-13T23:06:49.306+0200 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is s
trongly recommended with the WiredTiger storage engine
2022-09-13T23:06:49.306+0200 I STORAGE [initandlisten] ** See http://dochub.mongodb.org
/core/prodnotes-filesystem
2022-09-13T23:06:55.425+0200 I CONTROL [initandlisten]
2022-09-13T23:06:55.425+0200 I CONTROL [initandlisten] ** WARNING: Access control is not enabled
for the database.
2022-09-13T23:06:55.425+0200 I CONTROL [initandlisten] ** Read and write access to data
and configuration is unrestricted.
2022-09-13T23:06:55.425+0200 I CONTROL [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
> show dbs
admin 0.000GB
config 0.000GB
fwd 0.000GB
local 0.000GB
tweets 0.002GB
usuarios 0.000GB
> show collections
> use tweets
switched to db tweets
> show collections
tweets
> db.tweets.find().limit(1)
{ "_id" : ObjectId("553bbecae8f1e57878b72a1c"), "created_at" : "Sat Apr 25 16:19:03 +0000 2015",
"id" : 591999874540904400, "id_str" : "591999874540904448", "text" : "RT @webinara: RT: http://t.
co/tovD750cHb #webinar #TrueTwit #TechTip \n\n Node.js API development webinar:\nhttps://t.co/pBik
```

Fig 2. Consola mongo activa y BBDD tweets cargada

2. (3 puntos) Analizar qué estructura tiene un documento de la colección tweets, indicando: campos, tipo de datos de cada campo, para qué se utiliza cada campo, si hay campos que únicamente admiten un conjunto de valores, etc.

Se analizan los campos sin entrar al detalle de los subdocumentos. Se añade un esquema¹ de la estructura del documento. Se puede encontrar la documentación completa de los campos en la página oficial de Twitter para el soporte a desarrolladores².

1 [Dutta, Lalmohan & Maji, Giridhar & Sen, Soumya. \(2018\). A study on spatio-temporal topical analysis of Twitter Data. 10.1007/978-981-13-7403-6. Consultado 18/09/2022](#)

2 <https://developer.twitter.com/en/docs/twitter-api/v1/data-dictionary/object-model/tweet>

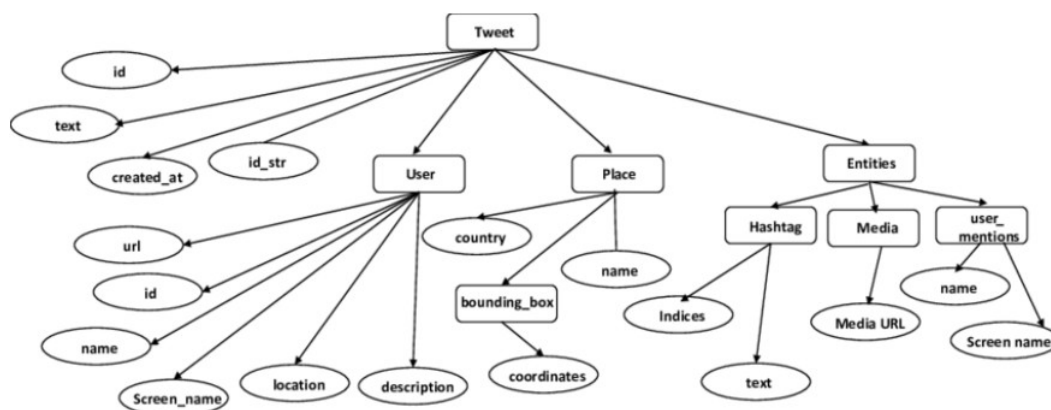


Fig 3. Esquema de claves del documento Tweet

Se detalla una tabla con los campos, tipo de dato y descripción funcional.

Campo	Tipo de dato	Descripción
Id	int64	Identificador único del tweet
Text	string	Texto del tweet
created_At	string	Fecha de creación del tweet UTC
id_str	string	Id en formato string
User	subdocumento	Contiene un subdocumento con la estructura del tipo User
User.url	string	Url al perfil del usuario
User.id	int64	Identificador único de usuario
User.name	string	Nombre de usuario
User.screen_name	string	Nombre de usuario secundario
User.location	string	Nombre de ubicación
User.description	string	Texto de descripción que introduce usuario
Place	Subdocumento	Contiene información de una ubicación
Place.country	string	Nombre de país de ubicación
Place.name	string	Nombre de un lugar
Place.bounding_box	Subdocumento	Información para delimitar un área en un mapa
Entities	Subdocumento	Subdocumento que contiene las distintas entidades de un tweet en subdocumentos
Entities.hashtag	Subdocumento	Subdocumento que contiene el hashtag de un tweet
Entities.hashtag.indices	Array of int	Localiza la posición del carácter especial de hashtag
Entities.hashtag.text	string	Texto del hashtag
Entities.media	Subdocumento	Subdocumento que contiene imágenes o videos del tweet
Entities.media.url	string	Url del host del contenido media
Entities.user_mention	Subdocumento	Subdocumento que contiene la información de una mención en un tweet
Entities.user_mention.name	string	Nombre del usuario al que se menciona
Entities.user_mention.screen_name	string	Nombre secundario del usuario al que se menciona

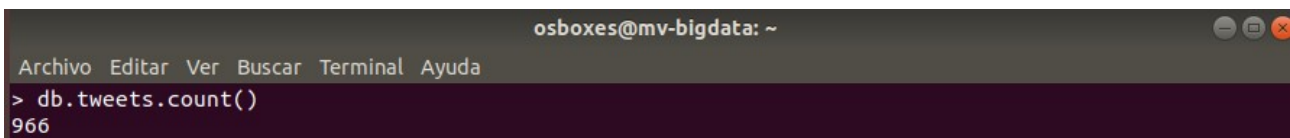
Se observan varios detalles relevantes:

- Los campos de un objeto Tweet pueden contener registros relacionales (Id) o dimensiones y medidas propias del objeto, pero tambien pueden contener objetos de otra naturaleza (como User, Coordinates, etc). Se hace necesaria una estructura de objetos flexible.
- Los datos a almacenar en el objeto pueden variar a lo largo del tiempo, porque se incluyan nuevos campos o que se queden obsoletos. Por lo que un esquema flexible sería deseable.
- Los objetos de varios tweets se relacionan entre sí (por ejemplo, menciones entre usuarios) por lo que es necesario poder relacionarlos mediante operaciones de join.

En conclusión, dados los requerimientos anteriores, una base de datos documental con características ACID sería lo más apropiado para persistir y gestionar estos datos.

3. (0,5 puntos) ¿Cuántos Tweets tiene la colección?

Se hace un recuento del número de registros mediante la instrucción `count()`.

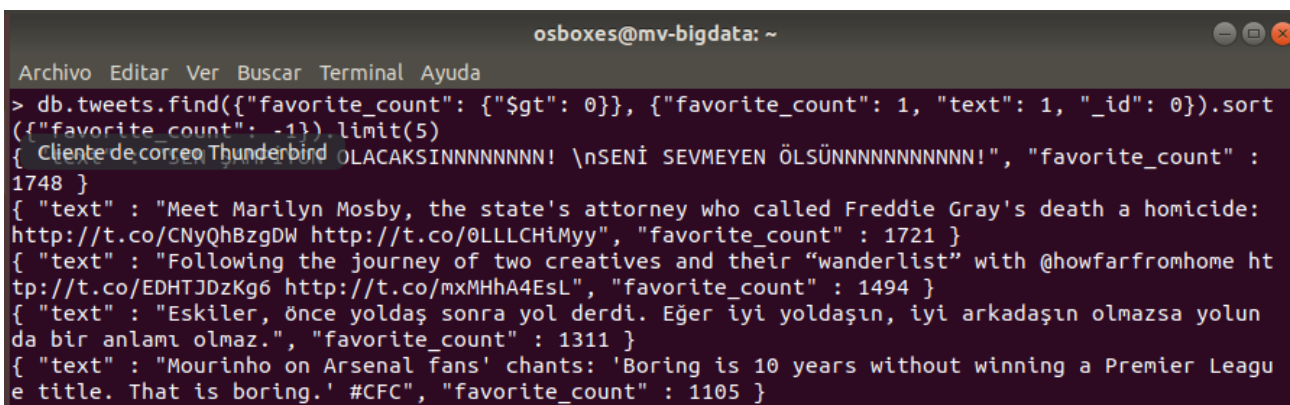


```
osboxes@mv-bigdata: ~
Archivo Editar Ver Buscar Terminal Ayuda
> db.tweets.count()
966
```

Fig 4. Resultado consulta ejercicio 3

4. (1 punto) En esta consulta, vamos a trabajar con el campo "favorite_count", que indica cuántas veces dicho tweet ha gustado a otros usuarios de Twitter. Listar todos los Tweets cuyo campo sea mayor que 0, devolviendo este campo además del campo "text" (texto del Tweet) ordenando la consulta del Tweet que más ha gustado al que menos ha gustado.

En este caso se usa el operador `$gt` para filtrar por el valor de `favorite_count` mayor a 0. se proyectan los campos `favorite_count`, `text` y se elimina `_id` por sencillez. Se ordenan de mayor a menor pasando como parametro -1 al operador `$sort`



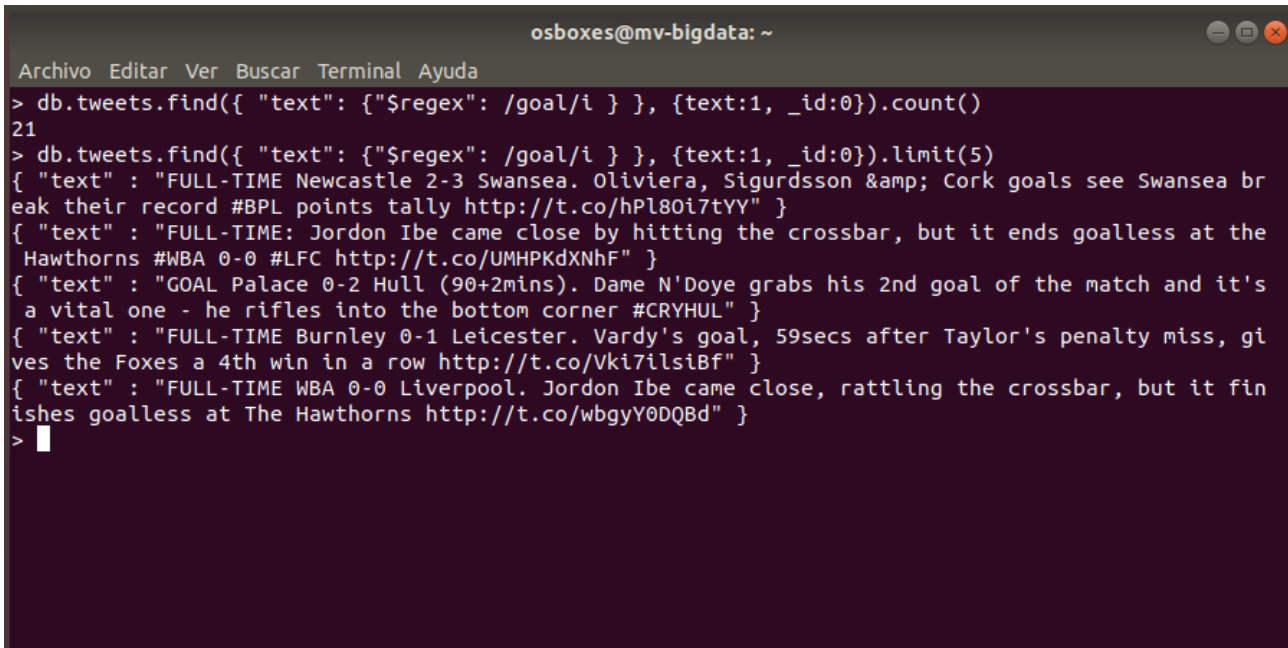
```
osboxes@mv-bigdata: ~
Archivo Editar Ver Buscar Terminal Ayuda
> db.tweets.find({"favorite_count": {"$gt": 0}}, {"favorite_count": 1, "text": 1, "_id": 0}).sort(
{"favorite_count": -1}).limit(5)
{ "Cliente de correo Thunderbird OLACAKSINNNNNNNN! \nSENİ SEVMEYEN ÖLSÜNNNNNNNNNN!", "favorite_count" :
1748 }
{ "text" : "Meet Marilyn Mosby, the state's attorney who called Freddie Gray's death a homicide:
http://t.co/CNyQhBzgDW http://t.co/0LLLCHiMy", "favorite_count" : 1721 }
{ "text" : "Following the journey of two creatives and their "wanderlist" with @howfarfromhome ht
tp://t.co/EDHTJDzKg6 http://t.co/mxMHhA4EsL", "favorite_count" : 1494 }
{ "text" : "Eskiler, önce yoldaş sonra yol derdi. Eğer iyi yoldaşın, iyi arkadaşın olmazsa yolun
da bir anlamı olmaz.", "favorite_count" : 1311 }
{ "text" : "Mourinho on Arsenal fans' chants: 'Boring is 10 years without winning a Premier Leagu
e title. That is boring.' #CFC", "favorite_count" : 1105 }
```

Fig 4. Resultado consulta ejercicio 4

5. (1,5 puntos) Lista todos los Tweets cuyo campo "text" contiene los caracteres "goal"

(sin distinguir entre mayúsculas/minúsculas). En el resultado, no muestres aquellos campos que no aporten valor. Justifica tu respuesta.

Se aplica la regla regex “/goal/i” para encontrar todos los strings que contengan el substring “goal” sin distinguir mayúsculas y minúsculas. Se proyecta solo el campo text eliminando el _id por sencillez. Se pone primero el recuento total y despues una muestra del resultado mediante limit().

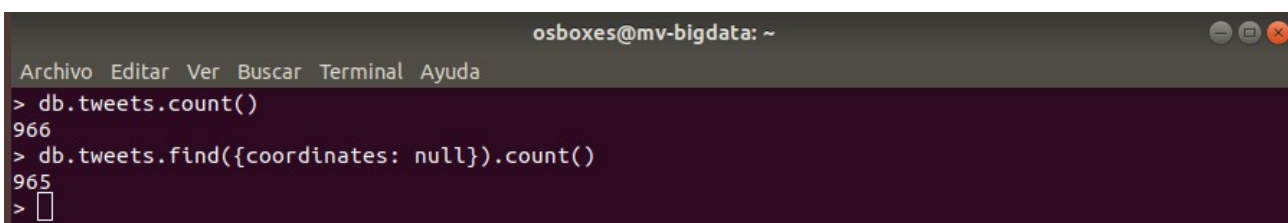


```
osboxes@mv-bigdata: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
> db.tweets.find({ "text": {"$regex": /goal/i } }, {text:1, _id:0}).count()  
21  
> db.tweets.find({ "text": {"$regex": /goal/i } }, {text:1, _id:0}).limit(5)  
{ "text" : "FULL-TIME Newcastle 2-3 Swansea. Oliviera, Sigurdsson & Cork goals see Swansea br  
eak their record #BPL points tally http://t.co/hPl80i7tYY" }  
{ "text" : "FULL-TIME: Jordon Ibe came close by hitting the crossbar, but it ends goalless at the  
Hawthorns #WBA 0-0 #LFC http://t.co/UMHPKdXNhF" }  
{ "text" : "GOAL Palace 0-2 Hull (90+2mins). Dame N'Doye grabs his 2nd goal of the match and it's  
a vital one - he rifles into the bottom corner #CRYHUL" }  
{ "text" : "FULL-TIME Burnley 0-1 Leicester. Vardy's goal, 59secs after Taylor's penalty miss, gi  
ves the Foxes a 4th win in a row http://t.co/vki7ilsibf" }  
{ "text" : "FULL-TIME WBA 0-0 Liverpool. Jordon Ibe came close, rattling the crossbar, but it fin  
ishes goalless at The Hawthorns http://t.co/wbgY0DQBd" }  
> █
```

Fig 5. Resultado consulta ejercicio 5

6. (1 punto) ¿Existe algún Tweet que tenga el campo “coordinates” vacío? Si existen, lístalos.

Se filtran los tweets por aquellos documentos de la colección que tengan el valor de coordinates a null, que es vacío en MongoDB. Se ve que son todos menos un documento (tweet).



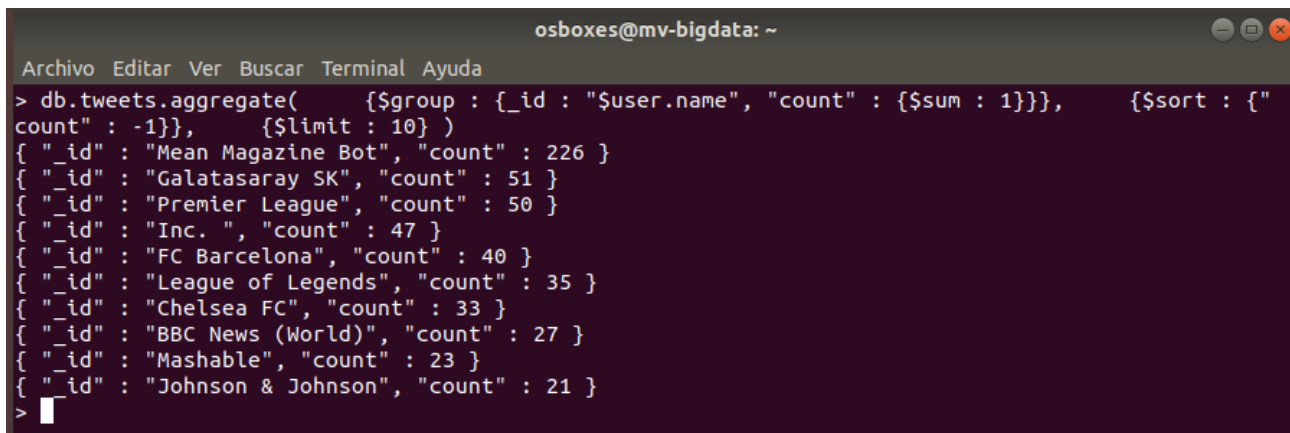
```
osboxes@mv-bigdata: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
> db.tweets.count()  
966  
> db.tweets.find({coordinates: null}).count()  
965  
> █
```

Fig 6. Resultado consulta ejercicio 6

7. (1 punto) ¿Cuál es el usuario que más Tweets ha publicado? (campo “user.name”).

Se realiza una operación de agregación. Se agrupa por nombre de usuario (campo name del subdocumento user) y se hace un count. Se ordenan los resultados ascendentemente con -1 sobre el campo de recuento count. Se muestran solo los 10 primeros resultados.

El usuario con más tweets es Mean Magazine Bot.



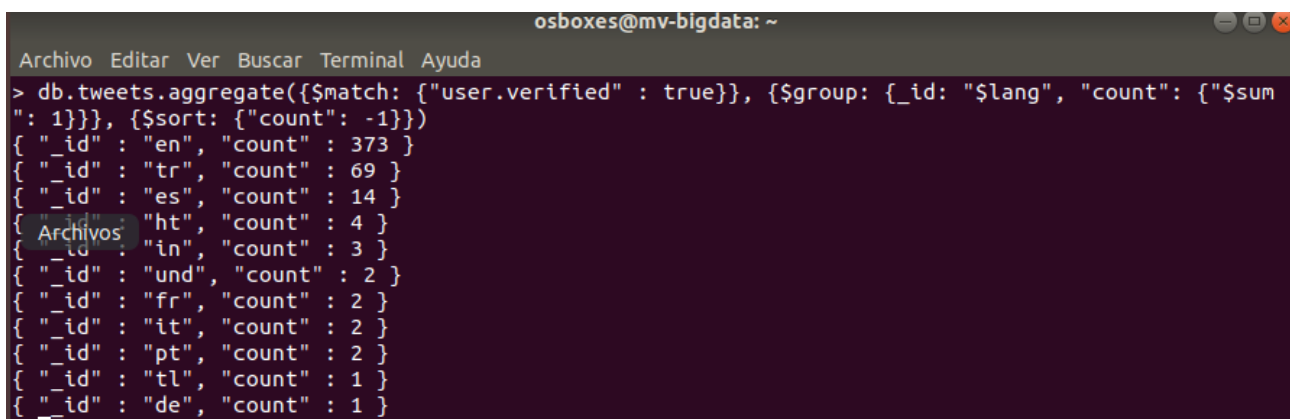
```
osboxes@mv-bigdata: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
> db.tweets.aggregate({$group: {_id: "$user.name", "count": {$sum: 1}}}, {$sort: {"count": -1}}, {$limit: 10})  
{ "_id": "Mean Magazine Bot", "count": 226 }  
{ "_id": "Galatasaray SK", "count": 51 }  
{ "_id": "Premier League", "count": 50 }  
{ "_id": "Inc. ", "count": 47 }  
{ "_id": "FC Barcelona", "count": 40 }  
{ "_id": "League of Legends", "count": 35 }  
{ "_id": "Chelsea FC", "count": 33 }  
{ "_id": "BBC News (World)", "count": 27 }  
{ "_id": "Mashable", "count": 23 }  
{ "_id": "Johnson & Johnson", "count": 21 }  
>
```

Fig 7. Resultado consulta ejercicio 7

8. (1 punto) Propón una consulta que tenga cierta complejidad y que sea interesante de realizar en base a los datos proporcionados. Justifica el porqué y explica la misma.

Se hace un recuento del idioma de los tweets para los usuarios verificados. Sirve para saber cual es el idioma dominante en la colección de tweets recabada. Se hace una consulta similar a la anterior pero añadiendo un match, para filtrar antes de agregar por usuarios verificados.

Se ve que el idioma dominante es el inglés (en) seguido del turco (tr).



```
osboxes@mv-bigdata: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
> db.tweets.aggregate({$match: {"user.verified": true}}, {$group: {_id: "$lang", "count": {$sum: 1}}}, {$sort: {"count": -1}})  
{ "_id": "en", "count": 373 }  
{ "_id": "tr", "count": 69 }  
{ "_id": "es", "count": 14 }  
{ "_id": "ht", "count": 4 }  
{ "_id": "in", "count": 3 }  
{ "_id": "und", "count": 2 }  
{ "_id": "fr", "count": 2 }  
{ "_id": "it", "count": 2 }  
{ "_id": "pt", "count": 2 }  
{ "_id": "tl", "count": 1 }  
{ "_id": "de", "count": 1 }
```

Fig 8. Resultado consulta ejercicio 7

Conclusiones

Se ha visto cómo utilizar un cliente de MongoDB a través de la consola mongo dentro del servidor. Se ha entendido la utilidad y ventajas de usar una base de datos orientada a documentos, en particular, para los tweets de usuarios (documentos), un dato cambiante a lo largo del tiempo y que necesita una estructura flexible a largo plazo.