Reporte de proyecto de Bases de datos NoSQL en MongoDB.

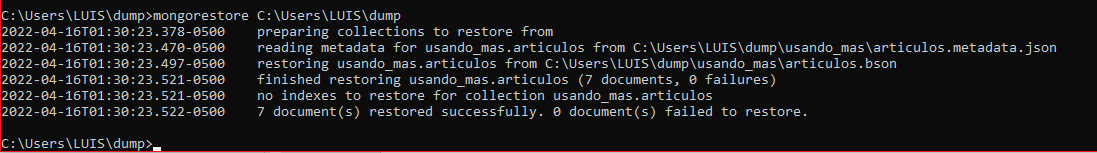
1. Recuperación de la base de datos.

Para recuperar la base de datos nos ubicamos en la carpeta que

Contienen el dump, es decir la descarga del archivo, y escribimos el comando

**Mongorestore C:\Users\LUIS\dump**

Nota: “*he movido el archivo de descarga hasta la carpeta del dump del usuario de la laptop.”*

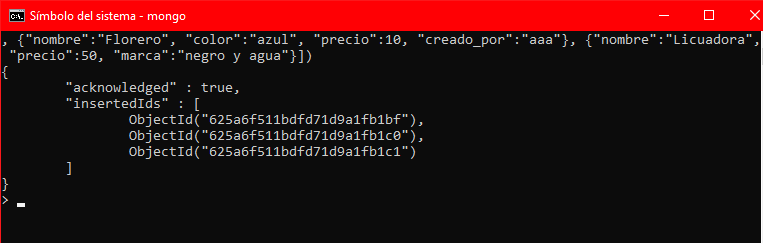


1. Actualizar la base de datos con los productos dados.

Se puede realizar a través del comando:

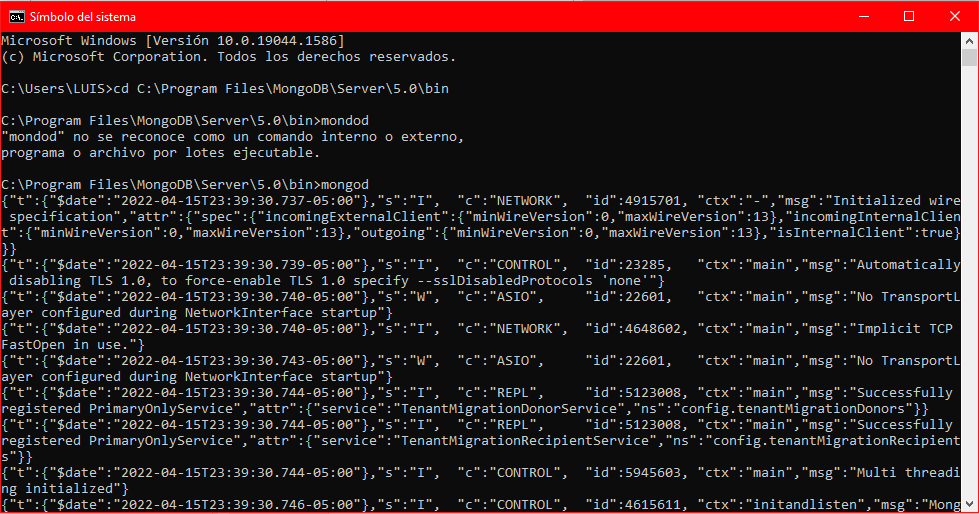
**db.articulos.insertMany([{“nombre” : “Mesa”, “color” : ”gris”, ”material” : ”madera”, “precio” : 100 },{“nombre” : “Florero”, “color” : “azul”, “precio” : “10”, “creado\_por” : “aaa”} , {“nombre” : “Licuadora”, “precio” : “50” , “marca” : “negro y agua”}])**

Nota: *“Las inserciones de documentos también se podían hacer de manera individual o a través de Mongo Compass”.*



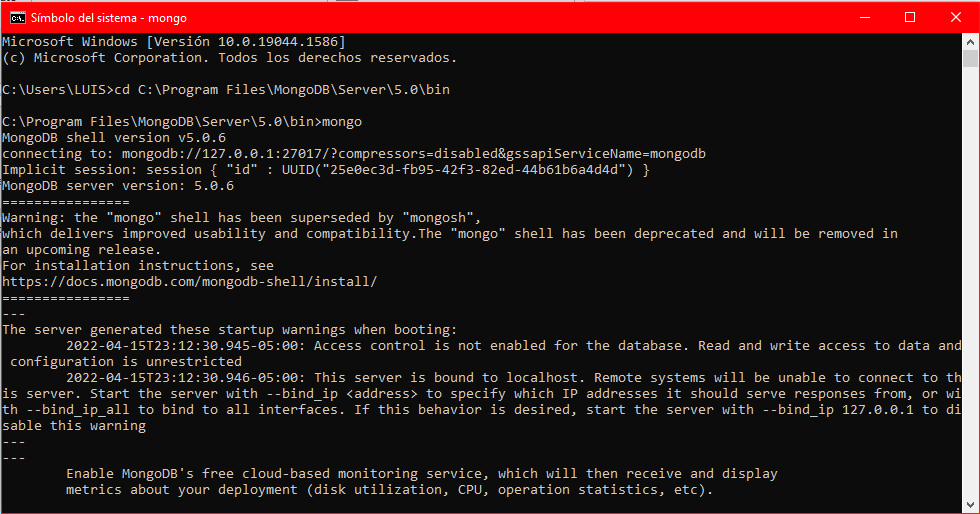
Crear la base de datos “**usando\_mas**” y una colección “**artículos”**

**Forma 1)** Iniciamos nuestro servidor con el comando mongod, después de habernos ingresado has la carpeta bin donde se encuentra los ejecutables de mongo.



El cual nos inicializará el servidor, minimizamos este Shell y abrimos otro Shell que será para el cliente.

Navegamos nuevamente hacia el bin de MongoDB y digitamos mongo.



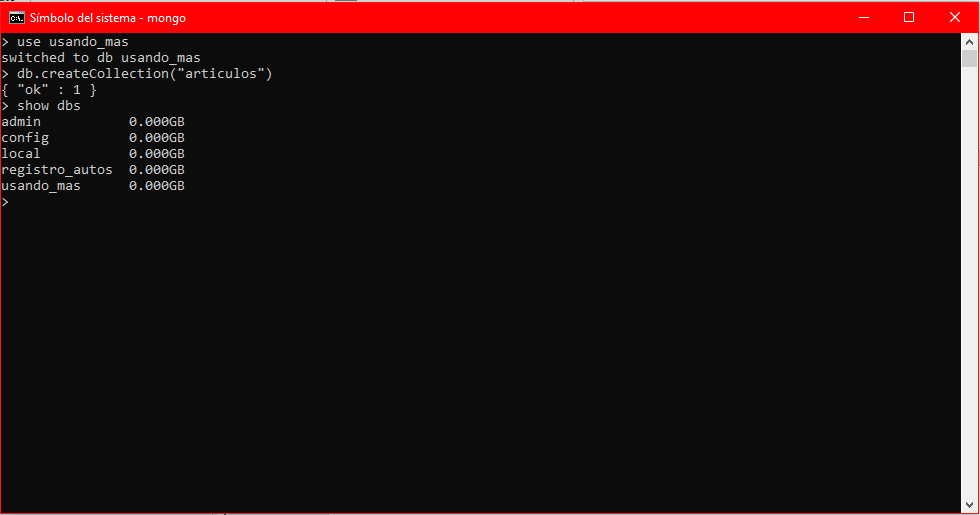
Que nos inicializara el cliente en el servidor a través del Shell.

Seguido de esto, creamos nuestra base de datos haciendo uso del comando:

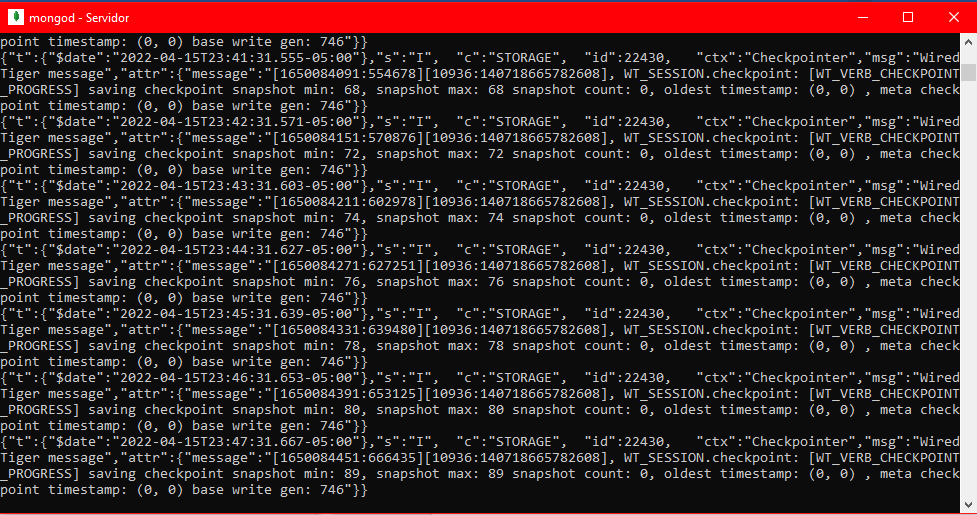
**use usando\_mas**

y luego de esto le asignamos una colección con el comando:

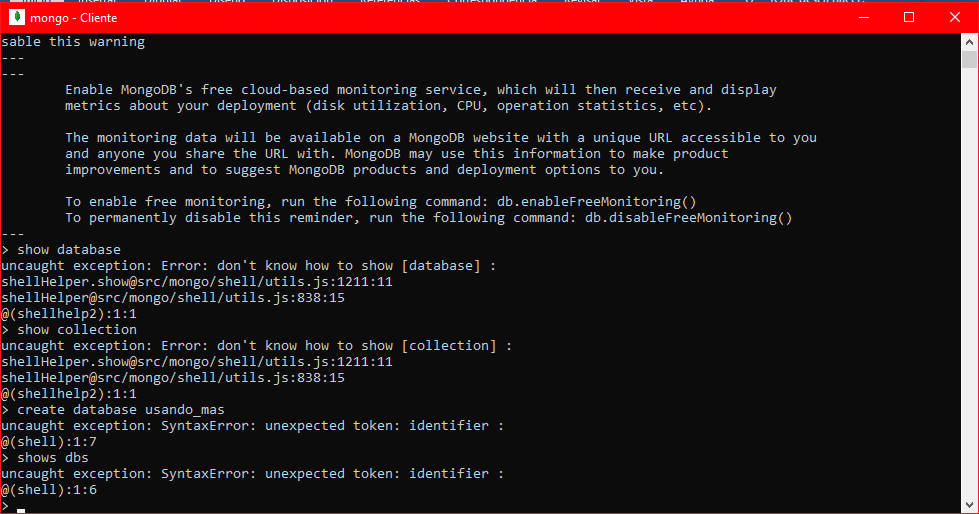
**db.createCollection(“articulos”)**



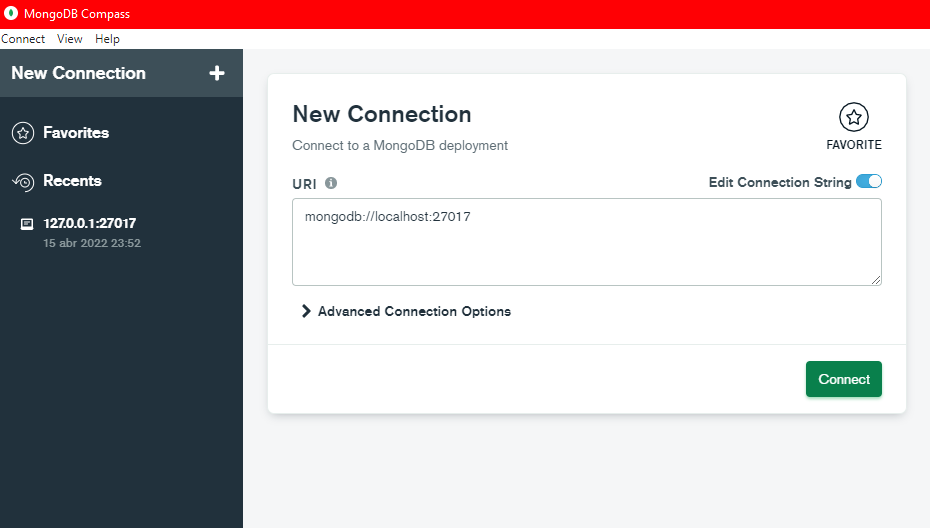
**Forma 2)** Iniciar el servidor de Mongo a través de la aplicación Mongod



Iniciar el cliente

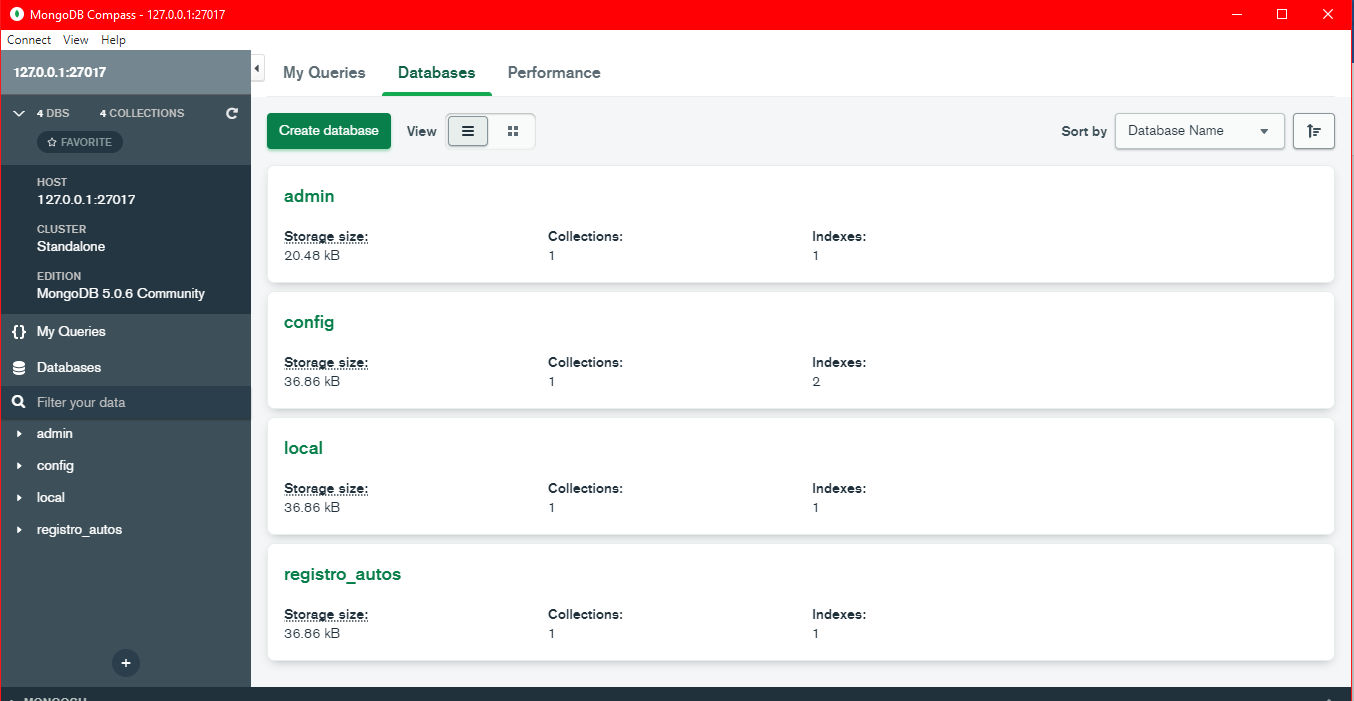


Ir a la aplicación de Mongo Compass y establecer una conexión con el servidor a través de esta interfaz gráfica y damos doble clic en nuestra conexión (**127.0.0.0:2717**).

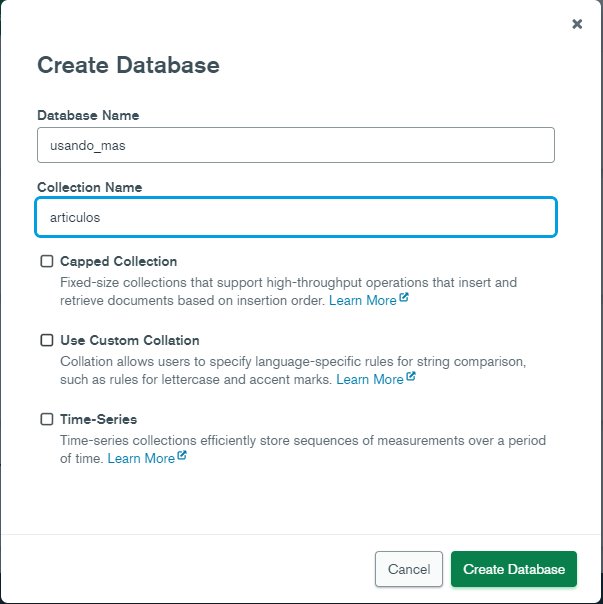


Que se conectara al servidor de Mongo.

Luego de ingresar, damos click en DATABASE donde podremos observar las bases de datos que ya hemos creado.



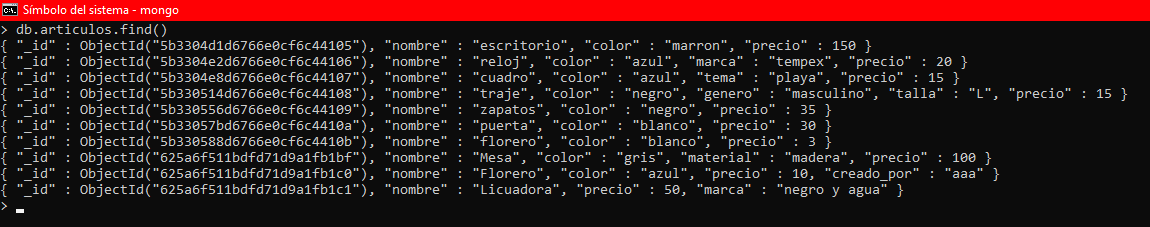
Pulsamos el botón que dice CREATE DATABASE. Donde procedemos a darle nombre a nuestra base de datos y asignamos su primera colección.



Pulsamos el botón Créate Database y ya nos genera nuestra base de datos.

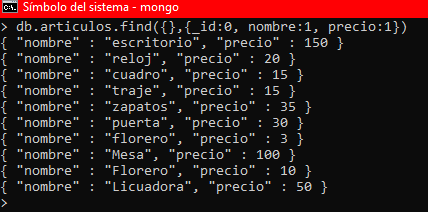
1. Listados que se hará al CEO:
   1. Todos los artículos con todos sus atributos

**db.articulos.find()** y **db.articulos.find().pretty** para un mejor formato



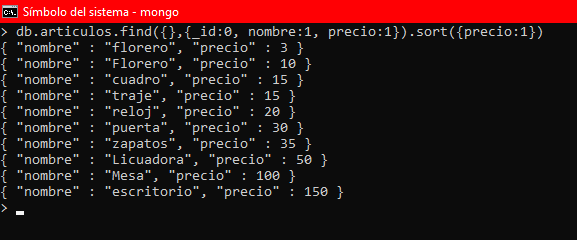
* 1. Todos los artículos con solo nombre y costo

**db.articulos.find({}, {\_id:0, nombre:1,precio:1})**



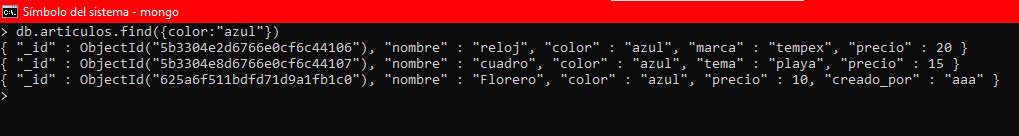
* 1. Todos los artículos con solo nombre y costo del producto ordenados por costo.

**db.articulos.find({},{\_id:0, nombre:1, precio:1}).sort({precio:1})**



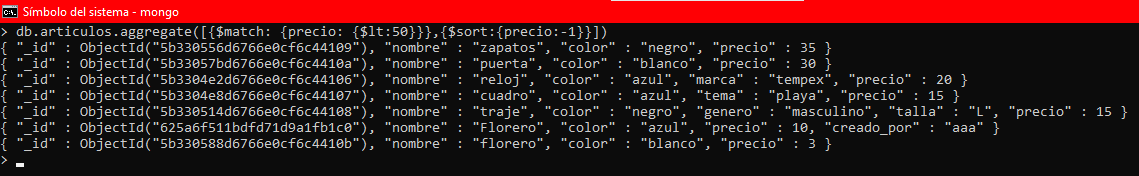
* 1. Todos los productos de color azul.

**db.articulos.find({color: “azul”})**



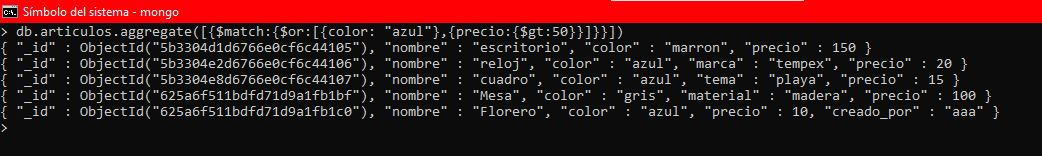
* 1. Todos los productos con precios menor a 50, ordenados descendientemente por precio

**db.articulos.aggregate([{$match: {precio:{$lt:50}}},{$sort: {precio:-1}}])**



* 1. Todos los productos que son azules y el precio es mayor que 50.

**db.articulos.aggregate([{$match: {$or:[{color: “azul”}, {precio:{$gt:50}}]}}])**



1. Listados para el análisis de venta.
   1. Cantidad de productos agrupados por color:

**db.articulos.aggregate([{$group:{\_id: “$color”, cantidad:{$sum:1}}}])**

**resultados de la consulta:**

**{ "\_id" : "blanco", "cantidad" : 2 }**

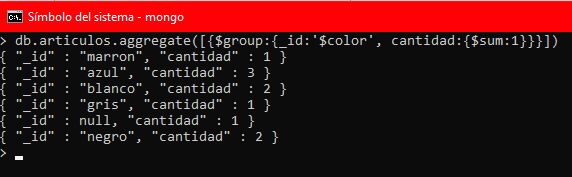
**{ "\_id" : "gris", "cantidad" : 1 }**

**{ "\_id" : "negro", "cantidad" : 2 }**

**{ "\_id" : null, "cantidad" : 1 }**

**{ "\_id" : "marron", "cantidad" : 1 }**

**{ "\_id" : "azul", "cantidad" : 3 }**

****

* 1. Cantidad de productos agrupados por color menores a 80.

**db.articulos.aggregate([{$match:{precio:{$lt:80}}},{$group:{\_id: “$color”, cantidad:{$sum:1}}}])**

**{ "\_id" : "negro", "cantidad" : 2 }**

**{ "\_id" : "azul", "cantidad" : 3 }**

**{ "\_id" : null, "cantidad" : 1 }**

**{ "\_id" : "blanco", "cantidad" : 2 }**

