

IES GONZALO NAZARENO
GESTIÓN DE BASES DE DATOS

PROYECTO MONGODB



Autor: Alfonso Roldán Amador

ÍNDICE

1	Seleccionamos el fichero json.....	1
2	Importamos los documentos a una colección.....	(1-2)
3	Insertar varios documentos.....	(3-4)
4	Eliminar varios documentos.....	5
5	Actualizar varios documentos.....	(5-6)
6	Consultas.....	(6-14)

1. Seleccionamos el fichero json.

En este caso usaré el fichero **vehiculos.json**.
Este fichero cuenta con la siguiente estructura:

```
{
  "id": 1,
  "NombrePropietario": "Stévina",
  "ApellidoPropietario": "Strowger",
  "GeneroPropietario": "Male",
  "Email_Fabricante": "bstrowger0@newsvine.com",
  "Marca": "GMC",
  "Modelo": "Yukon XL 2500",
  "Año": 2005,
  "Color": ["Violeta", "Gris", "Amarillo"],
  "Disponibilidad": true,
  "CV": 139,
  "metadata": {
    "sourceName": "GMC",
    "sourceUrl": "https://www.gmc.com/"
  }
},
{
  "id": 2,
  "NombrePropietario": "Estée",
```

Como podemos ver, el JSON se compone de varios diccionarios con los diferentes vehículos y propietarios, MongoDB interpreta estos diccionarios como documentos independientes, y los agrega a la colección.

2. Importamos los documentos a una colección.

Comenzaremos usando la utilidad **mongoimport**, para importar el fichero **json**. Esta utilidad se ejecuta desde la línea de comandos, (No desde la terminal de mongo).

```
mongoimport --db=proyecto_mongo --collection=vehiculos --file=ruta_fichero.json
```

```
root@debian-alfonso:/home/alfonso/Descargas# mongoimport --db=proyecto_mongo
--collection=vehiculos --file=vehiculos.json
2022-05-19T21:03:15.513+0200    connected to: mongodb://localhost/
2022-05-19T21:03:15.522+0200    1 document(s) imported successfully. 0 docume
nt(s) failed to import.
```

Podemos verificar que se haya creado la base de datos con el comando “**show dbs**”, desde el intérprete de mongo.

```
> show dbs
admin                0.000GB
config               0.000GB
local                0.000GB
proyecto_mongo       0.000GB
> 
```

Podemos eliminar la base de datos seleccionándola con:

```
use proyecto_mongo
```

Y posteriormente ejecutando:

```
db.dropDatabase()
```

Y para ver que base de datos estamos usando ejecutamos:

```
db
```

Si queremos mostrar una colección completa usamos:

```
db.vehiculos.find().pretty()
```

3. Insertar varios documentos.

Al no insertar el campo `_id`, MongoDB lo generará automáticamente.

Insertaremos documentos utilizando los dos métodos de inserción vistos:

Con `insertOne`:

- Realiza la inserción de un nuevo vehículo.

```
> db.vehiculos.insertOne({
...   "id": 57,
...   "NombrePropietario": "Edmée",
...   "ApellidoPropietario": "Lister",
...   "GeneroPropietario": "Male",
...   "Email_Fabricante": "xlisterli@elpais.com",
...   "Marca": "Subaru",
...   "Modelo": "Legacy",
...   "Año": 1996,
...   "Color": ["Blue"],
...   "Disponibilidad": false,
...   "CV": 93,
...   "metadata": {
...     "sourceName": "Subaru",
...     "sourceUrl" : "https://www.subaru.es/"
...   }
... })
{
  "acknowledged" : true,
  "insertedId" : ObjectId("6286d5d5a21c2ac69d334c93")
}
```

Con insertMany:

- Realiza la inserción de varios vehículos.

```
> db.vehiculos.insertMany([
...   {
...     "id": 77,
...     "NombrePropietario": "Cloé",
...     "ApellidoPropietario": "Edmonds",
...     "GeneroPropietario": "Female",
...     "Email_Fabricante": "gedmonds24@nsw.gov.au",
...     "Marca": "Nissan",
...     "Modelo": "Maxima",
...     "Año": 2009,
...     "Color": ["Violet"],
...     "Disponibilidad": false,
...     "CV": 133,
...     "metadata":{
...       "sourceName": "Nissan",
...       "sourceUrl" : "https://www.nissan.es/"
...     }
...   },
...   {
...     "id": 96,
...     "NombrePropietario": "Léandre",
...     "ApellidoPropietario": "Grimster",
...     "GeneroPropietario": "Male",
...     "Email_Fabricante": "mgrimster2n@xing.com",
...     "Marca": "Mercedes-Benz",
...     "Modelo": "CLK-Class",
...     "Año": 2003,
...     "Color": ["Orange"],
...     "Disponibilidad": false,
...     "CV": 143,
...     "metadata":{
...       "sourceName": "Mercedes Benz",
...       "sourceUrl" : "https://www.mercedes-benz.es"
...     }
...   }
... ])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("628769305b6a0e4a8f869a3f"),
    ObjectId("628769305b6a0e4a8f869a40")
  ]
}
```

4. Elimina varios documentos utilizando los dos métodos de eliminación de MongoDB.

Con deleteOne:

- Elimina el vehículo con modelo "LX".

```
> db.vehiculos.deleteOne({"Modelo" : "LX"})  
{ "acknowledged" : true, "deletedCount" : 1 }
```

Con deleteMany:

- Elimina los vehículos de la marca "Suzuki".

```
> db.vehiculos.deleteMany({"Marca": "Suzuki"})  
{ "acknowledged" : true, "deletedCount" : 4 }
```

5. Actualiza varios documentos utilizando los tres métodos de eliminación de MongoDB.

Con updateOne

- Actualiza el vehículo que cuente con 85 CV y cambia la disponibilidad a true.

```
> db.vehiculos.updateOne({"CV": 85},{ $set: {"Disponibilidad": true}})  
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
```

Con updateMany

- Realiza una actualización en la que todos los vehículos pertenezcan del un año superior a 2010, tengan 120 CV.

```
> db.vehiculos.updateMany({  
...   Año: { $gt: 2010}  
... }, {  
...   $set: {  
...     "CV": 120  
...   }  
... })  
{ "acknowledged" : true, "matchedCount" : 4, "modifiedCount" : 4 }
```

Con replaceOne

- Reemplaza el modelo por "CTS-V" en el vehículo que tenga "Cadillac" en el nombre de la fuente de sus metadatos.

```
> db.vehiculos.replaceOne({"metadata.sourceName":"Cadillac"}, {"Modelo":"CTS-V"})  
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
```

6. Consultas

CONSULTAS SIMPLES

- Muestra los vehículos que están disponibles.

```
db.vehiculos.find({"Disponibilidad": true}).pretty()
```

```
> db.vehiculos.find({"Disponibilidad": true}).pretty()  
{  
  "_id" : ObjectId("62877f036d2888a2cd6577f8"),  
  "id" : 1,  
  "NombrePropietario" : "Stévina",  
  "ApellidoPropietario" : "Strowger",  
  "GeneroPropietario" : "Male",  
  "Email_Fabricante" : "bstrowger0@newsvine.com",  
  "Marca" : "GMC",  
  "Modelo" : "Yukon XL 2500",  
  "Año" : 2005,  
  "Color" : [  
    "Violeta",  
  ],  
}
```


- Muestra los vehículos de la marca "Nissan".

```
db.vehiculos.find({"Marca": "Nissan"}).pretty()
```

```
> db.vehiculos.find({"Marca": "Nissan"}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd6577fc"),
  "id" : 10,
  "NombrePropietario" : "Nadège",
  "ApellidoPropietario" : "Gatfield",
  "GeneroPropietario" : "Male",
  "Email_Fabricante" : "hgatfield9@tinyurl.com",
  "Marca" : "Nissan",
  "Modelo" : "Versa",
  "Año" : 2011,
  "Color" : [
    "Lima",
    "Gris"
  ],
  "Disponibilidad" : false,
  "CV" : 120,
  "metadata" : {
    "sourceName" : "Nissan",
    "sourceUrl" : "https://www.nissan.es/"
  }
}
{
  "_id" : ObjectId("62877f1f8b3d4c410d4dcbec"),
  "id" : 77,
  "NombrePropietario" : "Cloé",
  "ApellidoPropietario" : "Edmonds",
  "GeneroPropietario" : "Female",
  "Email_Fabricante" : "gedmonds24@nsw.gov.au",
  "Marca" : "Nissan",
  "Modelo" : "Maxima",
  "Año" : 2009,
  "Color" : [
    "Violet"
  ],
  "Disponibilidad" : false,
  "CV" : 133,
  "metadata" : {
    "sourceName" : "Nissan",
    "sourceUrl" : "https://www.nissan.es/"
  }
}
```

- Muestra vehiculos en los que los emails de fabricantes terminen por .com.

```
db.vehiculos.find({"Email_Fabricante": /\.com$/}).pretty()
```

```
> db.vehiculos.find({"Email_Fabricante": /\.com$/}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd6577f8"),
  "id" : 1,
  "NombrePropietario" : "Stévina",
  "ApellidoPropietario" : "Strowger",
  "GeneroPropietario" : "Male",
  "Email_Fabricante" : "bstrowger0@newsvine.com",
  "Marca" : "GMC",
  "Modelo" : "Yukon XL 2500",
  "Año" : 2005,
  "Color" : [
    "Violeta",
    "Gris",
    "Amarillo"
  ],
  "Disponibilidad" : true,
  "CV" : 139,
  "metadata" : {
    "sourceName" : "GMC",
    "sourceUrl" : "https://www.gmc.com/"
  }
}
{
  "_id" : ObjectId("62877f036d2888a2cd6577fc"),
  "id" : 10,
  "NombrePropietario" : "Nadège",
  "ApellidoPropietario" : "Gatfield",
  "GeneroPropietario" : "Male",
  "Email_Fabricante" : "hgatfield9@tinyurl.com",
```

- Muestra los vehículos con menos de 100 CV.

```
db.vehiculos.find({"CV":{$lt:100}}).pretty()
```

```
> db.vehiculos.find({"CV":{$lt:100}}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd6577fb"),
  "id" : 9,
  "NombrePropietario" : "Céline",
  "ApellidoPropietario" : "Pellew",
  "GeneroPropietario" : "Female",
  "Email_Fabricante" : "apellew8@google.pl",
  "Marca" : "GMC",
  "Modelo" : "1500 Club Coupe",
  "Año" : 1995,
  "Color" : [
    "Negro",
    "Gris"
  ],
  "Disponibilidad" : true,
  "CV" : 88,
  "metadata" : {
    "sourceName" : "GMC",
    "sourceUrl" : "https://www.gmc.com/"
  }
}
{
  "_id" : ObjectId("62877f036d2888a2cd6577fe"),
  "id" : 11,
```

- Muestra el número de vehículos de la marca GMC.

```
db.vehiculos.find({"Marca": "GMC"}).count()
```

```
> db.vehiculos.find({"Marca": "GMC"}).count()
3
```

CONSULTAS CON ARRAYS

- Muestra todos los vehículos que tengan el color "Negro" ordenados por id.

```
db.vehiculos.find({"Color":{"$in":["Negro"]}}).sort({"id":1}).pretty()
```

```
> db.vehiculos.find({"Color":{"$in":["Negro"]}}).sort({"id":1}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd6577f7"),
  "id" : 2,
  "NombrePropietario" : "Estée",
  "ApellidoPropietario" : "Ratray",
  "GeneroPropietario" : "Male",
  "Email_Fabricante" : "cratray1@so-net.ne.jp",
  "Marca" : "Lexus",
  "Modelo" : "RX",
  "Año" : 2012,
  "Color" : [
    "Rosa",
    "Negro"
  ],
  "Disponibilidad" : false,
  "CV" : 120,
  "metadata" : {
    "sourceName" : "Lexus",
    "sourceUrl" : "https://www.lexusauto.es/"
  }
}
{
  "_id" : ObjectId("62877f036d2888a2cd6577f9"),
  "id" : 6,
  "NombrePropietario" : "Mélanie",
  "ApellidoPropietario" : "Hedon",
  "GeneroPropietario" : "Male",
  "Email_Fabricante" : "fhedon5@goo.gl",
  "Marca" : "Saturn",
  "Modelo" : "S-Series",
  "Año" : 1993,
  "Color" : [
    "Verde",
    "Azul",
    "Naranja",
    "Negro"
  ],
}
```

- Muestra los vehículos que tengan exactamente los colores "Blanco", "Plata" y "Gris".

```
db.vehiculos.find({"Color": ["Blanco", "Plata", "Gris"]}).pretty()
```

```
> db.vehiculos.find({"Color": ["Blanco", "Plata", "Gris"]}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd657802"),
  "id" : 14,
  "NombrePropietario" : "Méghane",
  "ApellidoPropietario" : "Goade",
  "GeneroPropietario" : "Female",
  "Email_Fabricante" : "kgoaded@de.vu",
  "Marca" : "Pontiac",
  "Modelo" : "Bonneville",
  "Año" : 1992,
  "Color" : [
    "Blanco",
    "Plata",
    "Gris"
  ],
  "Disponibilidad" : true,
  "CV" : 103,
  "metadata" : {
    "sourceName" : "Wikipedia",
    "sourceUrl" : "https://es.wikipedia.org/wiki/Pontiac"
  }
}
```

- Muestra los vehículos que tengan los colores "Negro" y "Gris". Pueden tener adicionalmente más colores, pero esos 2 son obligatorios.

```
db.vehiculos.find({"Color":{"$all: ["Negro","Gris"]}}).pretty()
```

```
> db.vehiculos.find({"Color":{"$all: ["Negro","Gris"]}}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd6577fb"),
  "id" : 9,
  "NombrePropietario" : "Céline",
  "ApellidoPropietario" : "Pellew",
  "GeneroPropietario" : "Female",
  "Email_Fabricante" : "apellew8@google.pl",
  "Marca" : "GMC",
  "Modelo" : "1500 Club Coupe",
  "Año" : 1995,
  "Color" : [
    "Negro",
    "Gris"
  ],
  "Disponibilidad" : true,
  "CV" : 88,
  "metadata" : {
    "sourceName" : "GMC",
    "sourceUrl" : "https://www.gmc.com/"
  }
}

{
  "_id" : ObjectId("62877f036d2888a2cd6577fd"),
  "id" : 8,
  "NombrePropietario" : "Léana",
  "ApellidoPropietario" : "Aubrey",
  "GeneroPropietario" : "Female",
  "Email_Fabricante" : "daubrey7@hostgator.com",
  "Marca" : "Mercury",
  "Modelo" : "Sable",
  "Año" : 2008,
  "Color" : [
    "Negro",
    "Verde",
    "Gris"
  ],
  "Disponibilidad" : false,
  "CV" : 119,
  "metadata" : {
    "sourceName" : "Wikipedia",
    "sourceUrl" : "https://es.wikipedia.org/wiki/Mercury_(autom%C3%B3viles)"
  }
}
```

CONSULTAS CON DOCUMENTOS EMBEBIDOS

- Muestra el nombre de los propietarios de los vehículos que tienen "Wikipedia" en el nombre de la fuente de sus metadatos.

```
db.vehiculos.find({"metadata.sourceName": "Wikipedia"}, {"NombrePropietario": 1}).pretty()
```

```
> db.vehiculos.find({"metadata.sourceName": "Wikipedia"}, {"NombrePropietario": 1}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd6577f9"),
  "NombrePropietario" : "Mélanie"
}
{
  "_id" : ObjectId("62877f036d2888a2cd6577fd"),
  "NombrePropietario" : "Léana"
}
{
  "_id" : ObjectId("62877f036d2888a2cd6577ff"),
  "NombrePropietario" : "Régine"
}
{
  "_id" : ObjectId("62877f036d2888a2cd657802"),
  "NombrePropietario" : "Méghane"
}
{ "_id" : ObjectId("62877f036d2888a2cd657806"), "NombrePropietario" : "Jú" }
> □
```

- Muestra el modelo y disponibilidad de los vehículos de la marca "GMC".

```
db.vehiculos.find({"Marca": "GMC"}, {"Modelo": 1, "Disponibilidad": 1}).pretty()
```

```
> db.vehiculos.find({"Marca": "GMC"}, {"Modelo": 1, "Disponibilidad": 1}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd6577f8"),
  "Modelo" : "Yukon XL 2500",
  "Disponibilidad" : true
}
{
  "_id" : ObjectId("62877f036d2888a2cd6577fb"),
  "Modelo" : "1500 Club Coupe",
  "Disponibilidad" : true
}
{
  "_id" : ObjectId("62877f036d2888a2cd65780a"),
  "Modelo" : "Sierra 2500",
  "Disponibilidad" : true
}
> □
```

- Muestra la marca, modelo y CV de los vehículos de un año mayor a 2010.

```
db.vehiculos.find({"Año": {$gt:2010}},{"Marca":1,"Modelo":1,"CV":1}).pretty()
```

```
> db.vehiculos.find({"Año": {$gt:2010}},{"Marca":1,"Modelo":1,"CV":1}).pretty()
{
  "_id" : ObjectId("62877f036d2888a2cd6577f7"),
  "Marca" : "Lexus",
  "Modelo" : "RX",
  "CV" : 120
}
{
  "_id" : ObjectId("62877f036d2888a2cd6577fc"),
  "Marca" : "Nissan",
  "Modelo" : "Versa",
  "CV" : 120
}
> □
```

CONSULTAS DE AGRUPACIÓN

- Agrupa cada modelo por marca, súmalos y devuelve cuantos modelos tiene cada marca.

```
db.vehiculos.aggregate([{$group:{_id: "$Marca", "Nº de modelos":{$sum: 1}}}]).pretty()
```

```
> db.vehiculos.aggregate([{$group:{_id: "$Marca", "Nº de modelos":{$sum: 1}}}]).pretty()
{ "_id" : "Mercury", "Nº de modelos" : 1 }
{ "_id" : "Infiniti", "Nº de modelos" : 1 }
{ "_id" : null, "Nº de modelos" : 1 }
{ "_id" : "Plymouth", "Nº de modelos" : 1 }
{ "_id" : "Saturn", "Nº de modelos" : 1 }
{ "_id" : "Nissan", "Nº de modelos" : 2 }
{ "_id" : "Isuzu", "Nº de modelos" : 1 }
{ "_id" : "GMC", "Nº de modelos" : 3 }
{ "_id" : "Oldsmobile", "Nº de modelos" : 1 }
{ "_id" : "Audi", "Nº de modelos" : 1 }
{ "_id" : "Mazda", "Nº de modelos" : 1 }
{ "_id" : "Mercedes-Benz", "Nº de modelos" : 2 }
{ "_id" : "MINI", "Nº de modelos" : 1 }
{ "_id" : "Pontiac", "Nº de modelos" : 1 }
{ "_id" : "Subaru", "Nº de modelos" : 1 }
{ "_id" : "Lexus", "Nº de modelos" : 1 }
> □
```