

# **MÓDULO** Gestión de Bases de Datos

Azahara Aragón Martín

## Query Documents

### Índice

● Preparación del proyecto	2
● Creación inserciones	3
● Creación consultas	5
● Webgrafía	6

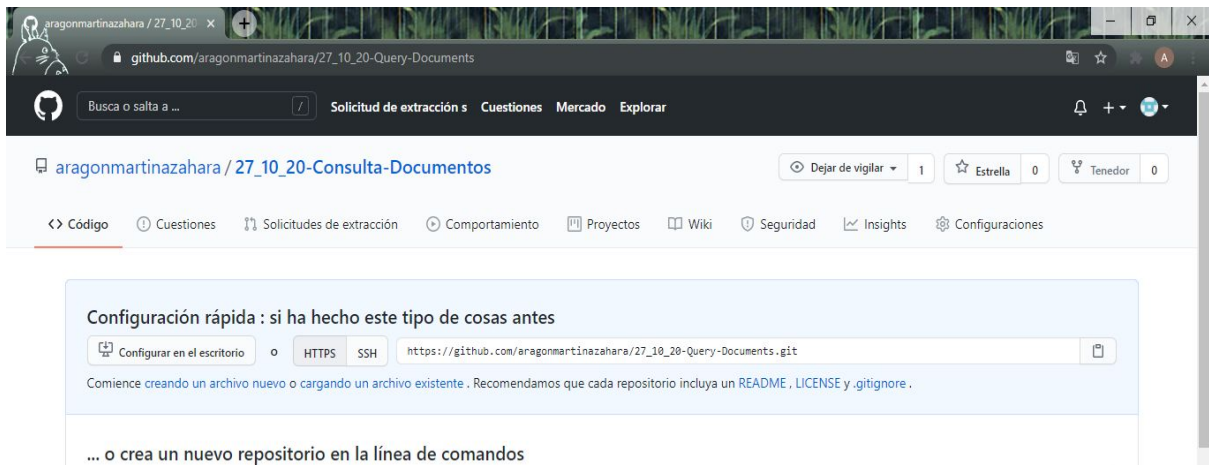
## Preparación del proyecto

Para empezar, lo primero que he hecho ha sido crear la estructura de carpetas `src` y `doc` para poder empezar a crear los archivos necesarios, y crear el archivo `README.md`.



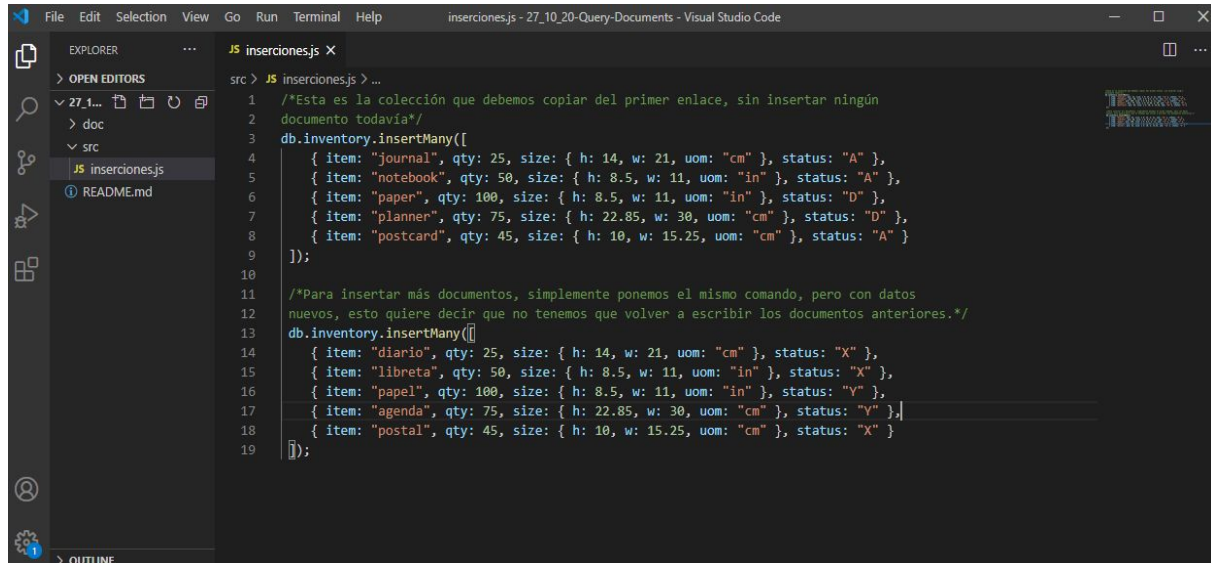
```
File Edit Selection View Go Run Terminal Help • README.md - 27_10_20-Query-Documents - Visual Studio Code
EXPLORER
> OPEN EDITORS 1 UNSAVED
27_10_20-QUERY-DOCUME...
  > doc
  > src
  README.md
README.md
1 # Tercer proyecto con Mongo y Git
2
3 Primero creamos el repositorio correspondiente en **GitHub**
4
5 Crearemos las carpetas *src* y *doc*
6
7 La carpeta *doc* contendrá el documento PDF explicativo de este proyecto
8
9 La carpeta *src* contendrá los archivos "inserciones.js" y "consultas.js"
10
11 Cuando tengamos los documentos correspondientes al proyecto en sus respectivas carpetas, se subirá el
    proyecto a **GitHub**
```

También he creado el repositorio en mi cuenta de **GitHub**, para que una vez hecho el proyecto, simplemente ejecute en el terminal Git y escriba los comandos correspondientes para subirlo.



## Creación inserciones

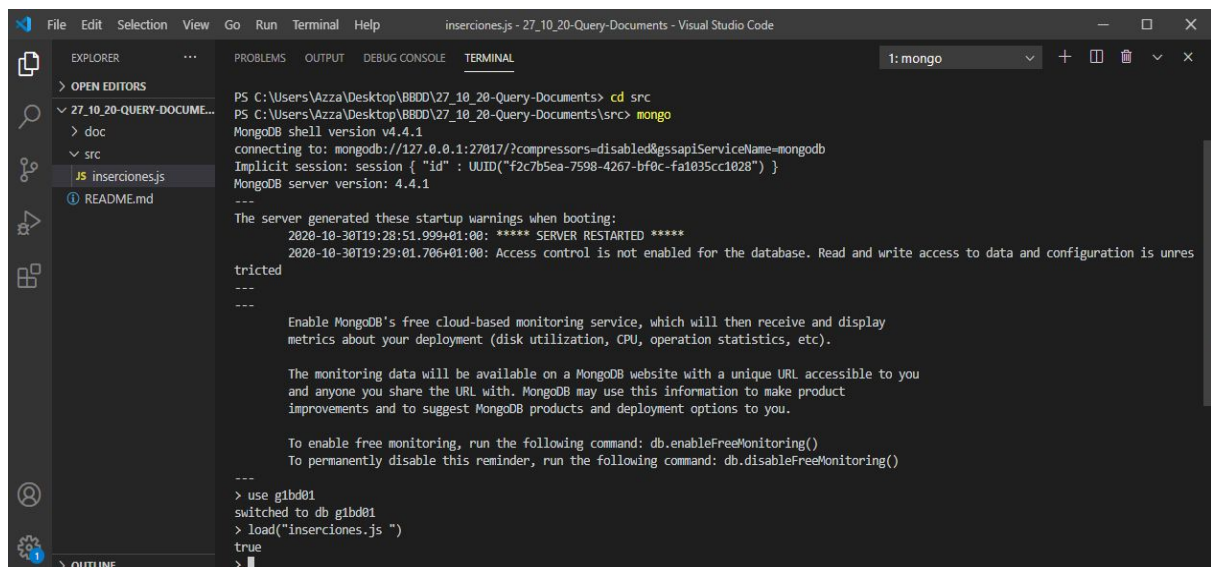
Primero crearemos nuestro archivo .js dentro de la carpeta src de nuestro equipo. Dentro de ese archivo insertamos otros documentos en la colección *inventory*, donde la estructura será similar pero con otros valores en los campos.



```

1  /*Esta es la colección que debemos copiar del primer enlace, sin insertar ningún
2  documento todavía*/
3  db.inventory.insertMany([
4    { item: "journal", qty: 25, size: { h: 14, w: 21, uom: "cm" }, status: "A" },
5    { item: "notebook", qty: 50, size: { h: 8.5, w: 11, uom: "in" }, status: "A" },
6    { item: "paper", qty: 100, size: { h: 8.5, w: 11, uom: "in" }, status: "D" },
7    { item: "planner", qty: 75, size: { h: 22.85, w: 30, uom: "cm" }, status: "D" },
8    { item: "postcard", qty: 45, size: { h: 10, w: 15.25, uom: "cm" }, status: "A" }
9  ]);
10
11  /*Para insertar más documentos, simplemente ponemos el mismo comando, pero con datos
12  nuevos, esto quiere decir que no tenemos que volver a escribir los documentos anteriores.*/
13  db.inventory.insertMany([
14    { item: "diario", qty: 25, size: { h: 14, w: 21, uom: "cm" }, status: "X" },
15    { item: "libreta", qty: 50, size: { h: 8.5, w: 11, uom: "in" }, status: "X" },
16    { item: "papel", qty: 100, size: { h: 8.5, w: 11, uom: "in" }, status: "Y" },
17    { item: "agenda", qty: 75, size: { h: 22.85, w: 30, uom: "cm" }, status: "Y" },
18    { item: "postal", qty: 45, size: { h: 10, w: 15.25, uom: "cm" }, status: "X" }
19  ]);
  
```

Una vez escrito todos los documentos que queremos insertar en la colección *inventory*, guardaremos el archivo, y una vez guardado abrimos un terminal, nos ubicamos en el directorio *src* y ejecutaremos mongo. En el terminal seleccionamos la base de datos correspondiente y cargamos el archivo *inserciones.js* con el comando *load()*.



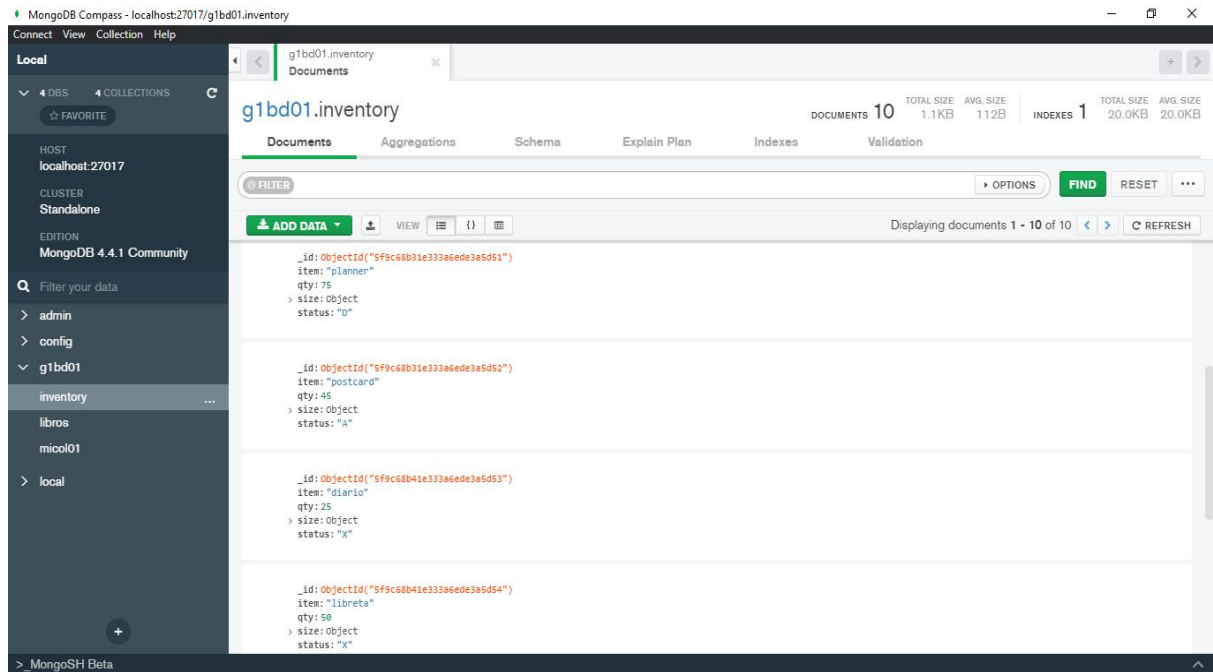
```

PS C:\Users\Azza\Desktop\BDD\27_10_20-Query-Documents> cd src
PS C:\Users\Azza\Desktop\BDD\27_10_20-Query-Documents\src> mongo
MongoDB shell version v4.4.1
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("f2c7b5ea-7598-4267-bf0c-fa1035cc1028") }
MongoDB server version: 4.4.1
---
The server generated these startup warnings when booting:
2020-10-30T19:28:51.999+01:00: ***** SERVER RESTARTED *****
2020-10-30T19:29:01.706+01:00: Access control is not enabled for the database. Read and write access to data and configuration is unres
tricted
---
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()
---
> use g1bd01
switched to db g1bd01
> load("inserciones.js ")
true
>
  
```

Cuando hayamos cargado el archivo inserciones.js podremos ejecutar Mongo Compass para poder comprobar que está todo adecuadamente.



## Creación consultas

Para realizar las consultas lo primero que tenemos que hacer es abrir un terminal donde ejecutemos mongo, y ahí seleccionamos la base de datos para hacer las consultas con query selectors y query documents:

- Consulta \$in y consulta \$gte:

The screenshot shows a Visual Studio Code editor with a file named `consultas.js` in the `src` directory. The code contains two MongoDB queries. The first query uses the `$gte` operator to find items with a quantity greater than or equal to 75. The second query uses the `$in` operator to find items with a status of either 'X' or 'D'. The terminal at the bottom shows the execution of these queries, displaying the resulting JSON documents.

```

1 /*Consulta con el operador query selector $gte*/
2 db.inventory.find( { qty: { $gte : 75 } } )
3 { "_id" : ObjectId("5f9c68b31e333a6ede3a5d50"), "item" : "paper", "qty" : 100,
4   "size" : { "h" : 8.5, "w" : 11, "uom" : "in" }, "status" : "D" }
5 { "_id" : ObjectId("5f9c68b31e333a6ede3a5d51"), "item" : "planner", "qty" : 75,
6   "size" : { "h" : 22.85, "w" : 30, "uom" : "cm" }, "status" : "D" }
7 { "_id" : ObjectId("5f9c68b41e333a6ede3a5d55"), "item" : "papel", "qty" : 100,
8   "size" : { "h" : 8.5, "w" : 11, "uom" : "in" }, "status" : "Y" }
9 { "_id" : ObjectId("5f9c68b41e333a6ede3a5d56"), "item" : "agenda", "qty" : 75,
10  "size" : { "h" : 22.85, "w" : 30, "uom" : "cm" }, "status" : "Y" }
11
12 /*Consulta con el operador query selector $in*/
13 db.inventory.find( { status: { $in: [ "X", "D" ] } } )
14 { "_id" : ObjectId("5f9c68b31e333a6ede3a5d50"), "item" : "paper", "qty" : 100,
15   "size" : { "h" : 8.5, "w" : 11, "uom" : "in" }, "status" : "D" }
16 { "_id" : ObjectId("5f9c68b31e333a6ede3a5d51"), "item" : "planner", "qty" : 75,
17   "size" : { "h" : 22.85, "w" : 30, "uom" : "cm" }, "status" : "D" }
18 { "_id" : ObjectId("5f9c68b41e333a6ede3a5d53"), "item" : "diario", "qty" : 25,
19   "size" : { "h" : 14, "w" : 21, "uom" : "cm" }, "status" : "X" }
20 { "_id" : ObjectId("5f9c68b41e333a6ede3a5d54"), "item" : "libreta", "qty" : 50,
21   "size" : { "h" : 8.5, "w" : 11, "uom" : "in" }, "status" : "X" }

```

- Consulta \$and implícita, con los operadores query selector \$in y \$gte:

The screenshot shows a Visual Studio Code editor with a file named `consultas.js` in the `src` directory. The code contains an implicit `$and` query that finds items with a quantity greater than or equal to 75 and a status of either 'X' or 'D'. The terminal at the bottom shows the execution of this query, displaying the resulting JSON documents.

```

24 /*Consulta con el operador $and*/
25 db.inventory.find( { qty: { $gte : 75 }, status: { $in: [ "X","D" ] } } )
26 { "_id" : ObjectId("5f9c68b31e333a6ede3a5d50"), "item" : "paper", "qty" : 100,
27   "size" : { "h" : 8.5, "w" : 11, "uom" : "in" }, "status" : "D" }
28 { "_id" : ObjectId("5f9c68b31e333a6ede3a5d51"), "item" : "planner", "qty" : 75,
29   "size" : { "h" : 22.85, "w" : 30, "uom" : "cm" }, "status" : "D" }
30

```

## Webgrafía

- <https://docs.mongodb.com/manual/tutorial/query-documents/>
- <https://docs.mongodb.com/manual/reference/operator/query/>
- [https://www.youtube.com/watch?v=6WIDQSAE2VE&feature=youtu.be&ab\\_channel=AdolfoSaltoS%C3%A1nchezdelCorral](https://www.youtube.com/watch?v=6WIDQSAE2VE&feature=youtu.be&ab_channel=AdolfoSaltoS%C3%A1nchezdelCorral)