

# Rubrica 4: Sigue el tutorial MongoDB

SO:



Docker:



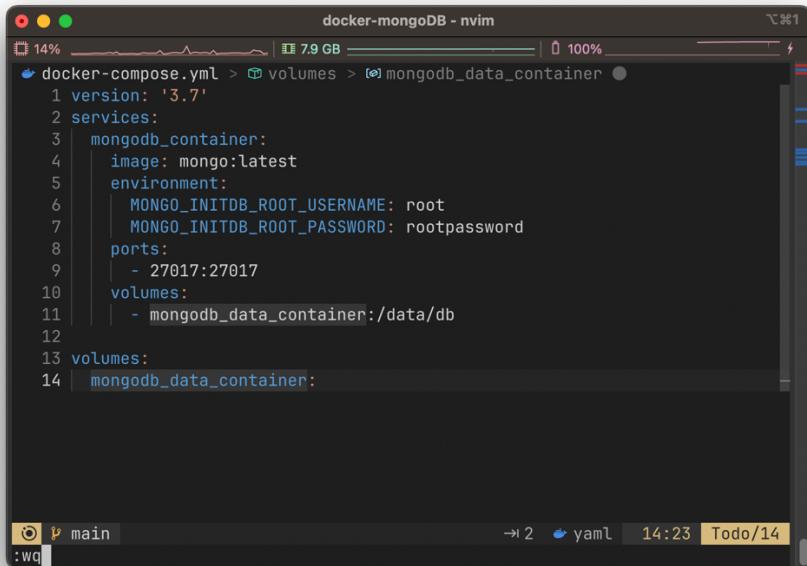
1. Creamos un directorio para almacenar el archivo de configuración:

A screenshot of a terminal window on a Mac OS X system. The terminal session shows the following commands being run:

```
alberto@MBP-de-Alberto:~/gitSalesianos/BBDD2223/docker-mongoDB
$ cd gitSalesianos/BBDD2223/
$ mkdir docker-mongoDB
$ cd docker-mongoDB/
$ vim docker-compose.yml
```

The terminal window has a dark theme and shows the current working directory as `alberto@MBP-de-Alberto:~/gitSalesianos/BBDD2223/docker-mongoDB`.

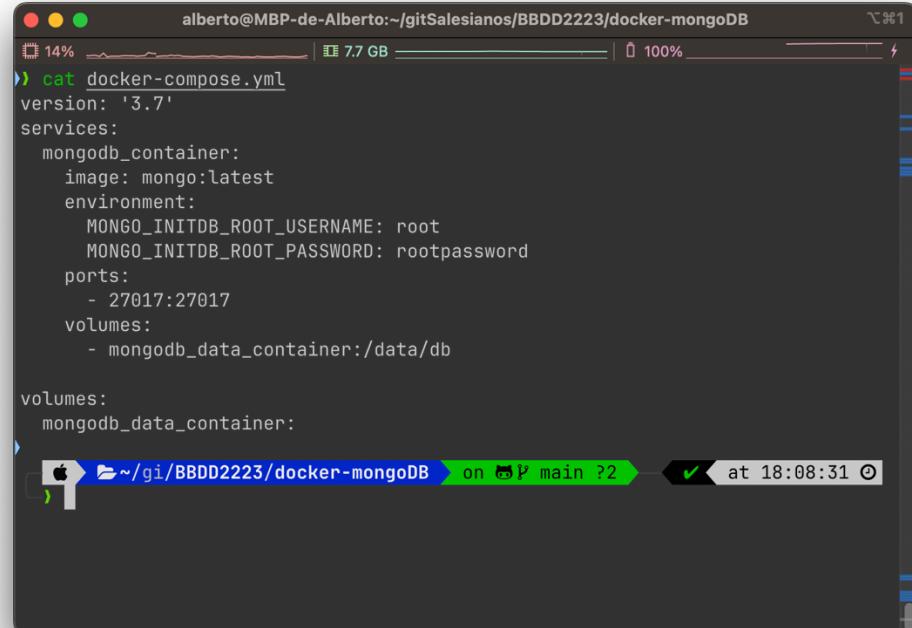
## 2. Creamos el archivo de configuración de Docker Compose:



```
docker-mongoDB - nvim
dockers-compose.yml > volumes > mongodb_data_container
1 version: '3.7'
2 services:
3   mongodb_container:
4     image: mongo:latest
5     environment:
6       MONGO_INITDB_ROOT_USERNAME: root
7       MONGO_INITDB_ROOT_PASSWORD: rootpassword
8     ports:
9       - 27017:27017
10    volumes:
11      - mongodb_data_container:/data/db
12
13 volumes:
14   mongodb_data_container:
```

Este archivo indica que la imagen es la última de mongoDB, el usuario es 'root', el password 'rootpassword', mapeamos el puerto 27017 del contenedor con el de la máquina y le asignamos un volumen que se guardará en /data/db y se llamará 'mongodb\_data\_container'.

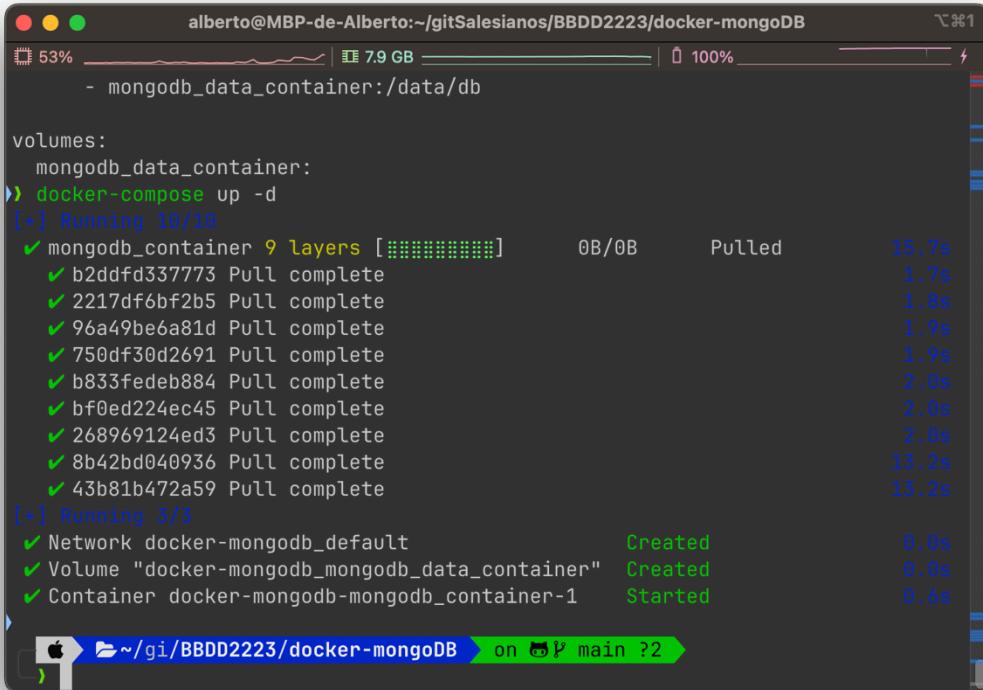
Nos aseguramos que se ha creado correctamente con el comando 'cat':



```
alberto@MBP-de-Alberto:~/gitSalesianos/BBDD2223/docker-mongoDB
cat docker-compose.yml
version: '3.7'
services:
  mongodb_container:
    image: mongo:latest
    environment:
      MONGO_INITDB_ROOT_USERNAME: root
      MONGO_INITDB_ROOT_PASSWORD: rootpassword
    ports:
      - 27017:27017
    volumes:
      - mongodb_data_container:/data/db

volumes:
  mongodb_data_container:
```

### 3. Levantamos el contenedor:

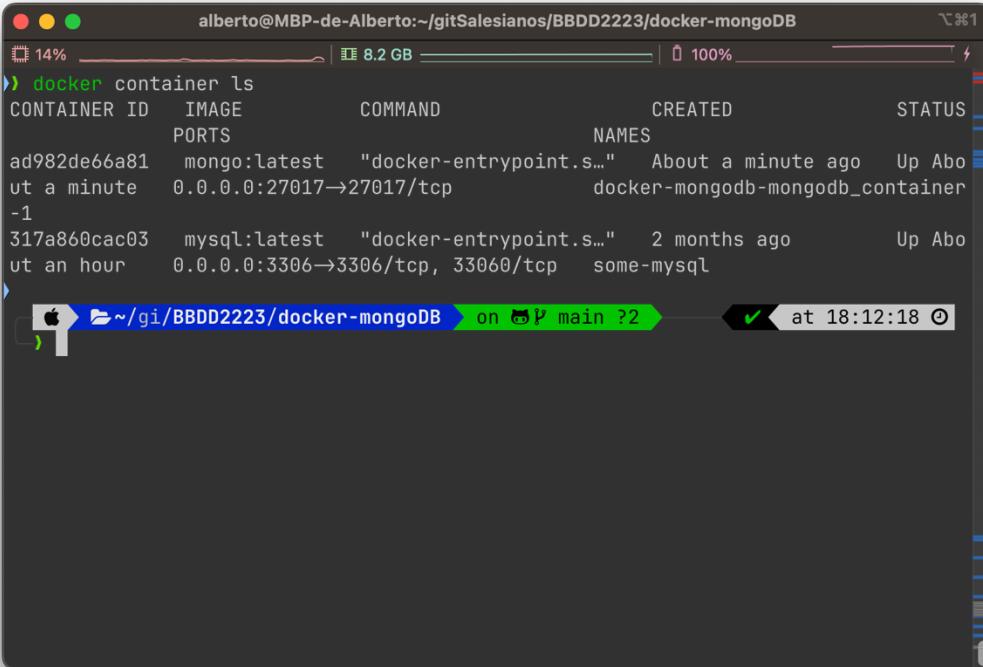


```
alberto@MBP-de-Alberto:~/gitSalesianos/BBDD2223/docker-mongoDB
mongodbs_data_container:/data/db

volumes:
  mongodb_data_container:
    > docker-compose up -d
[+] Running 10/10
  ✓ mongodb_container 9 layers [██████████]      0B/0B      Pulled      15.7s
    ✓ b2ddfd337773 Pull complete                1.7s
    ✓ 2217df6bf2b5 Pull complete                1.8s
    ✓ 96a49be6a81d Pull complete                1.9s
    ✓ 750df30d2691 Pull complete                1.9s
    ✓ b833fedeb884 Pull complete                2.0s
    ✓ bf0ed224ec45 Pull complete                2.0s
    ✓ 268969124ed3 Pull complete                2.0s
    ✓ 8b42bd040936 Pull complete                13.2s
    ✓ 43b81b472a59 Pull complete                13.2s
[+] Running 3/3
  ✓ Network docker-mongodb_default           Created      0.0s
  ✓ Volume "docker-mongodb_mongodb_data_container" Created      0.0s
  ✓ Container docker-mongodb-mongodb_container-1 Started     0.6s

> ~/gi/BBDD2223/docker-mongoDB on MacBook-Pro main ?2
```

### 4. Comprobamos que se ha levantado:



```
alberto@MBP-de-Alberto:~/gitSalesianos/BBDD2223/docker-mongoDB
CONTAINER ID   IMAGE      COMMAND      CREATED      STATUS      NAMES
ad982de66a81   mongo:latest "docker-entrypoint.s..."  About a minute ago   Up Abo
ut a minute   0.0.0.0:27017→27017/tcp      docker-mongodb-mongodb_container
-1
317a860cac03   mysql:latest "docker-entrypoint.s..."  2 months ago       Up Abo
ut an hour    0.0.0.0:3306→3306/tcp, 33060/tcp  some-mysql

> ~/gi/BBDD2223/docker-mongoDB on MacBook-Pro main ?2 → ✓ at 18:12:18 ⏺
```

A screenshot of the Docker Desktop application interface. At the top, there are icons for Docker Desktop, Upgrade plan, a search bar with placeholder text "Search for local and re...", and user account information. Below the header is a sidebar with icons for Containers, Images, Networks, Volumes, Services, and a plus sign for creating new resources. The main area is titled "Containers" with a "Give feedback" link. It displays a table of running containers:

	Name	Image	Status	Port(s)	Last started
hopeful_chebyshev	655b9d507f4d	trafelex/php	Exited (255)	80:8080	20 days ago
some-mysql	317a860cac03	mysql:latest	Running	3306:3306	1 hour ago
docker-mongodb		-	Running (1/1)		2 minutes ago
mongodb_container-1	ad982de66a81	mongo:latest	Running	27017:27017	2 minutes ago

At the bottom of the container list, it says "Showing 4 items". The status bar at the bottom shows system resources: RAM 3.60 GB, CPU 0.60%, Disk 51.22 GB avail. of 58.37 GB, and a note that it is Not connected to Hub. The version is v4.18.0.

## 5. Nos conectamos a la base de datos con DataGrip

A screenshot of the DataGrip application. The left sidebar shows "Data Sources" selected, with a list of "Project Data Sources" including "@localhost [2]". The main panel is titled "Data Sources and Drivers" and shows the configuration for the "@localhost [2]" data source. The "General" tab is selected. The configuration fields are:

- Name: @localhost [2]
- Comment: (empty)
- Connection type: default
- Driver: MongoDB
- Host: localhost
- Port: 27017
- Authentication: User & Password
- User: root
- Password: (redacted)
- Save: Forever
- Database: (empty)
- URL: mongodb://localhost:27017

A success message box is displayed at the bottom left, stating:

**Succeeded** Copy  
DBMS: Mongo DB (ver. 6.0.5)  
Case sensitivity: plain=mixed, delimited=mixed  
Driver: MongoDB JDBC Driver (ver. 1.18, JDBC4.2)  
Ping: 152 ms

Buttons at the bottom right include "Cancel", "Apply", and "OK".

The screenshot shows the MySQL Workbench interface. On the left, the Database Explorer pane lists databases: @localhost [16], @localhost [2] (selected), admin, config, and local. In the center, the 'console\_2' tab is active, displaying a terminal session:

```
console_2 [localhost [2]] x
show dbs
use mydatabase
show collections
```

The Services pane at the bottom shows a terminal session for MongoDB:

```
[2023-05-03 18:37:37] Connected
test> show dbs
[2023-05-03 18:37:37] 3 rows retrieved starting from 1 in 121 ms

test> use mydatabase
[2023-05-03 18:37:56] 1 row retrieved starting from 1 in 100 ms

mydatabase> show collections
[2023-05-03 18:38:12] 0 rows retrieved in 100 ms (execution: 95 ms,
fetching: 5 ms)
```

The status bar at the bottom indicates: @localhost [2]: DBMS: Mongo DB (ver. 6.0.5) // Case sensitivity: plain=mixed, delimited=mixed // ... (2 minutes ago) LF UTF-8 4 spaces.

## 6. También lo hacemos con mongo Shell

The screenshot shows a macOS terminal window titled 'alberto@MBP-de-Alberto:~'. The user runs several Homebrew commands to install MongoDB:

```
brew tap mongodb/brew
Tapping mongodb/brew
Cloning into '/opt/homebrew/Library/Taps/mongodb/homebrew-brew'...
remote: Enumerating objects: 1184, done.
remote: Counting objects: 100% (469/469), done.
remote: Compressing objects: 100% (144/144), done.
remote: Total 1184 (delta 363), reused 388 (delta 324), pack-reused 715
Receiving objects: 100% (1184/1184), 255.06 KiB | 1.82 MiB/s, done.
Resolving deltas: 100% (658/658), done.
Tapped 18 formulae (36 files, 332.2KB).
brew update
Already up-to-date.
brew install mongodb-community@6.0
```

The terminal shows the command completed successfully: took 4s at 18:39:48.

```
alberto@MBP-de-Alberto:~
```

9% 6.2 GB 100%

```
Installing mongodb/brew/mongodb-community
→ Caveats
To restart mongodb/brew/mongodb-community after an upgrade:
  brew services restart mongodb/brew/mongodb-community
Or, if you don't want/need a background service you can just run:
  /opt/homebrew/opt/mongodb-community/bin/mongod --config /opt/homebrew/etc/mongod.conf
→ Summary
🍺 /opt/homebrew/Cellar/mongodb-community/6.0.5: 11 files, 200.5MB, built in 2 seconds
→ Running `brew cleanup mongodb-community`...
Disable this behaviour by setting HOMEBREW_NO_INSTALL_CLEANUP.
Hide these hints with HOMEBREW_NO_ENV_HINTS (see `man brew`).
→ Caveats
→ mongodb-community
To restart mongodb/brew/mongodb-community after an upgrade:
  brew services restart mongodb/brew/mongodb-community
Or, if you don't want/need a background service you can just run:
  /opt/homebrew/opt/mongodb-community/bin/mongod --config /opt/homebrew/etc/mongod.conf
```

took 37s at 18:40:58

```
alberto@MBP-de-Alberto:~/gitSalesianos/BBDD2223/docker-mongoDB
```

10% 7.5 GB 100%

```
~/gi/BBDD2223/docker-mongoDB on main ?2
mongosh admin -u root -p rootpassword
```

```
mongosh mongodb://<credentials>@127.0.0.1:27017/admin?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.8.2
Current Mongosh Log ID: 64529087e0cfe5ed9012d24d
Connecting to:      mongodb://<credentials>@127.0.0.1:27017/admin?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.8.2
Using MongoDB:     6.0.5
Using Mongosh:    1.8.2

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

-----
The server generated these startup warnings when booting
2023-05-03T16:29:16.586+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2023-05-03T16:29:17.362+00:00: vm.max_map_count is too low

-----
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

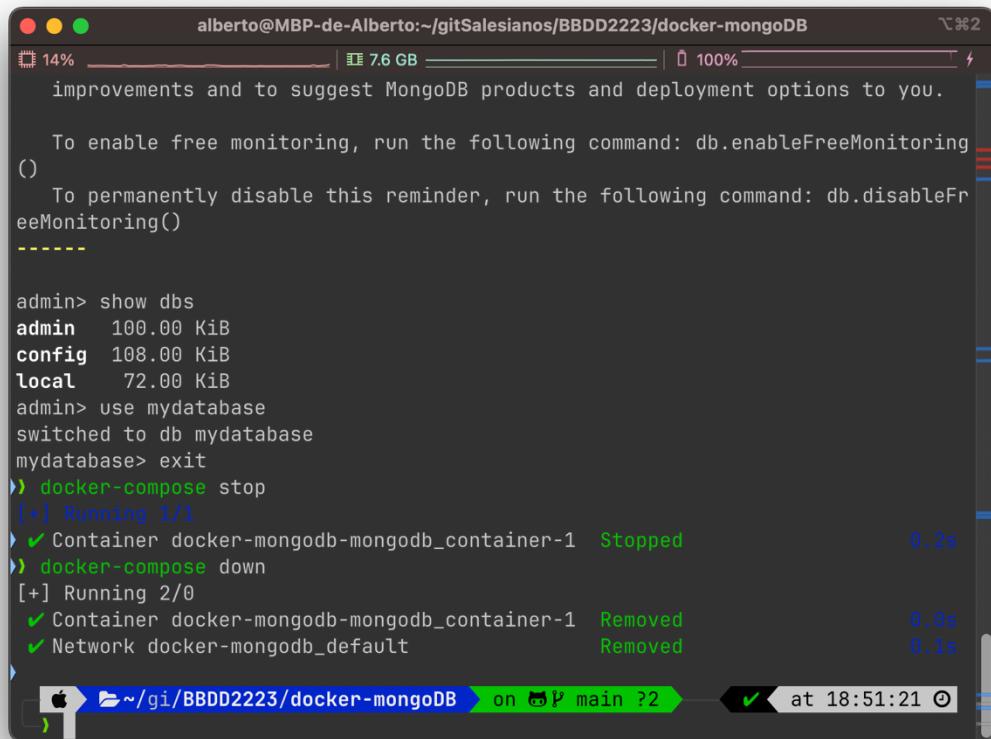
The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()

admin>
```

```
mongosh mongodb://<credentials>@127.0.0.1:27017/admin?directConnection=true&serverSe...
-----  
-----  
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()  
-----  
admin> show dbs  
admin 100.00 KiB  
config 108.00 KiB  
local 72.00 KiB  
admin> use mydatabase  
switched to db mydatabase  
mydatabase>
```

Con el comando 'exit' salimos de la base de datos y con los siguientes comandos: primero apagamos/cerramos el contenedor y con el segundo lo eliminamos (pero no el volumen)



The screenshot shows a terminal window on a Mac OS X desktop. The title bar reads "alberto@MBP-de-Alberto:~/gitSalesianos/BBDD2223/docker-mongoDB". The window displays the following text:

```
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()

-----
admin> show dbs
admin 100.00 KiB
config 108.00 KiB
local 72.00 KiB
admin> use mydatabase
switched to db mydatabase
mydatabase> exit
> docker-compose stop
[+] Running 1/1
-> ✓ Container docker-mongodb-mongodb_container-1 Stopped 0.2s
-> docker-compose down
[+] Running 2/0
✓ Container docker-mongodb-mongodb_container-1 Removed 0.0s
✓ Network docker-mongodb_default Removed 0.1s
->
```

The terminal prompt at the bottom is "at 18:51:21".