

Bootcamp Full Stack Engineer

Fase 3 - Desarrollo Back End

Módulo 35

MongoDB

MongoDB

Una de las bases de datos NoSQL más conocidas hoy en día es **MongoDB**, una alternativa muy distinta a las bases de datos relacionales que, sin embargo, es una excelente opción para **el almacenamiento de datos de aplicaciones**.

Al aproximarnos por primera vez a Mongo, encontramos algunas diferencias que pueden desorientarnos:

- **No veremos tablas.**
- **No contamos con registros.**
- **No tenemos SQL.**

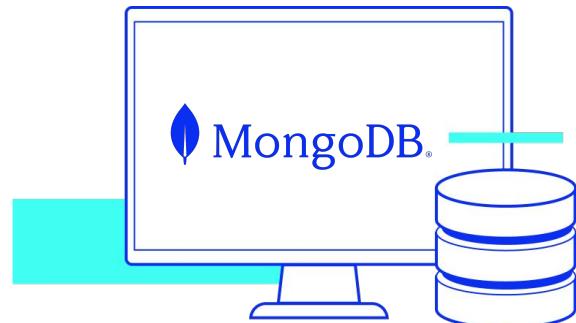
A continuación, veremos más **sobre su funcionamiento, qué podemos hacer con ella y cómo podemos usarla**.

Web oficial:
<https://www.mongodb.com/es>



Características de MongoDB

- Acceso sofisticado y eficiente a los datos.
- Integridad de los datos.
- Gestión e integración de la empresa.
- Flexibilidad.
- Escalabilidad y rendimiento.
- Siempre disponible.
- Despliegues globales.



Sobre MongoDB

MongoDB es **una base de datos orientada a documentos** que, en vez de guardar los datos en registros lo hace, precisamente, en documentos. Éstos se almacenan en **BSON**, una representación binaria de JSON.

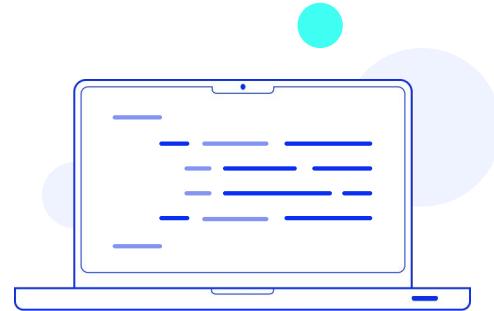
A diferencia de las bases de datos relacionales, con Mongo **no debemos seguir un esquema**. Los documentos de una misma **colección** (similar a una tabla en una base de datos relacional) simplemente pueden tener distintos esquemas.

Para entenderlo, veamos un ejemplo en la siguiente slide.



Supongamos que tenemos una colección llamada *Personas*. Allí, podríamos almacenar un documento de la siguiente manera:

```
{  
  nombre: "Pedro",  
  apellidos: "Martínez Campo",  
  edad: 22,  
  aficiones: ["fútbol", "tenis", "ciclismo"],  
  amigos: [  
    {  
      nombre: "María",  
      edad: 22,  
    },  
    {  
      nombre: "Luis",  
      edad: 28,  
    },  
  ],  
}
```



En el ejemplo anterior vimos un típico ejemplo de un **documento JSON**. Tiene **strings, arrays, subdocumentos y números**.

En esa misma colección, podríamos guardar un documento como el que sigue:

```
{  
  nombre: "Luis",  
  estudios: "Administración y Dirección de Empresas"  
,  
  amigos:12  
}
```

A diferencia del ejemplo en la diapositiva anterior, este bloque de código no sigue el mismo esquema. Cuenta con **menos campos, campos nuevos y de distinto tipo**. En una base de datos relacional, esto sería algo indispensable, sin embargo, para MongoDB, esto es absolutamente válido.



Características de MongoDB

- **Velocidad:** el rasgo distintivo de MongoDB es **su velocidad**, la cual, gracias a su **sistema de consulta de contenidos**, se encuentra en excelente equilibrio entre **rendimiento y funcionalidad**.
- **Consultas ad hoc:** con Mongo, podemos hacer **todo tipo de consultas**, como consultas de rangos, búsquedas de expresiones regulares y de campos. Además, estas pueden devolver tanto un campo específico del documento como una función JavaScript definida por el usuario.
- **Indexación:** el concepto de **índices** en Mongo es similar al de bases de datos relacionales. La diferencia más importante es que **cualquier campo documentado puede ser indexado y añadir múltiples índices secundarios**.
- **Replicación:** MongoDB soporta el tipo de replicación **primario-secundario**. Así, mientras podemos realizar consultas con el primario, el secundario actúa como réplica de datos en solo lectura a modo de copia de seguridad. La particularidad es que los nodos secundarios **permiten elegir un nuevo primario en caso de que el primario actual no responda**.

- **Balanceo de carga:** Mongo puede ejecutarse **simultáneamente en múltiples servidores**. Esto ofrece un balance de carga o servicio de replicación de datos que permite mantener el sistema en funcionamiento **en caso de que falle el hardware**.
- **Almacenamiento de archivos:** Mongo cuenta con una funcionalidad llamada **GridFS** que permite la manipulación de **archivos y contenido**.
- **Ejecución de JavaScript del lado del servidor:** con Mongo, podemos hacer consultas con JavaScript y hacer que sean enviadas directamente **a la base de datos para ser ejecutadas**.



Ventajas y desventajas de MongoDB

Ventajas

- Permite **validar documentos**.
- Cuenta con **motores de almacenamiento integrado**.
- Tiene un menor tiempo de recuperación ante fallos.

Desventajas

- No es una solución adecuada para aplicaciones con **transacciones complejas**.
- No tiene un reemplazo para las soluciones de herencia.
- Aún es **una tecnología joven**.

¿Cómo funciona MongoDB?

MongoDB está escrito en C++ y las consultas se hacen **pasando objetos JSON como parámetro**. Esto último es algo bastante comprensible dado que los propios documentos se almacenan en **BSON**. Por ejemplo:

```
db.Clientes.find({nombre: "Pedro"});
```

La consulta anterior buscará todos los clientes cuyo nombre sea “Pedro”.

- De serie, MongoDB viene con una **consola** desde la que podemos ejecutar los comandos. Esta está construida sobre **JavaScript**, por lo que las consultas se realizan con ese lenguaje. Esto implica que, además de las funciones de MongoDB, podemos utilizar varias de las funciones de JavaScript.

En la consola también podemos definir **variables, funciones o utilizar bucles**.

- Si queremos usar un lenguaje de programación de nuestra elección, existen *drivers* para muchos de ellos: hay **drivers para C#, Java, Node.js, PHP, Python, Ruby, C, C + +, Perl o Scala**. Pese a ser soportados por MongoDB, no todos están en el mismo estado de madurez. Por ejemplo, el de C es una versión *alpha*.

Si deseamos usar un lenguaje específico, lo mejor es buscar los *drivers* disponibles para corroborar que sean adecuados para un entorno de producción.



¿Dónde podemos usar MongoDB?

Puede usar MongoDB cualquier aplicación que necesite almacenar **datos semi estructurados**. Esto es lo que sucede con muchas aplicaciones CRUD o de varios desarrollos web actuales.

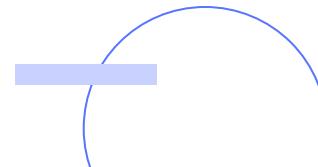
Es importante recordar que, pese a que las colecciones de MongoDB no necesitan definir un esquema, **es central que diseñemos nuestra aplicación para seguir uno**. Tendremos que decidir si necesitamos **normalizar los datos, desnormalizarlos o utilizar una aproximación híbrida** y estas decisiones pueden afectar el rendimiento de nuestra aplicación.

En definitiva, es recomendable que **el esquema lo definan las consultas que vayamos a realizar con más frecuencia**.

MongoDB resulta de gran utilidad en entornos que requieren **escalabilidad**. Con sus opciones de **replicación y sharding**, podemos conseguir un sistema que **escala horizontalmente** sin presentar inconvenientes.

¿Dónde no usar MongoDB?

- En Mongo **no existen las transacciones**, pese a que nuestra aplicación pueda simularlas, MongoDB no cuenta con esta posibilidad. Solo garantiza operaciones atómicas **a nivel del documento**, por lo que si para nuestro desarrollo las transacciones resultan centrales, debemos elegir otro sistema.
- Tampoco contamos con **JOINS**, por lo que, si precisamos consultar datos relacionados en dos o más colecciones, **deberemos hacer más de una consulta**. Lo recomendado es que, si podemos estructurar nuestros datos en tablas y necesitamos las relaciones, lo ideal es elegir un RDBMS clásico.



- **Consultas de agregación:** en MongoDB encontraremos un framework para hacer consultas de agregación llamado ***Aggregation Framework***, aunque también es posible usar *Map Reduce*. Sin embargo, estos métodos **no alcanzan la potencia de un sistema relacional** por lo que, si deseamos explotar informes más complejos, es recomendable elegir otro sistema.



¿Cómo se instala MongoDB?

Debemos **descargar los binarios para nuestro sistema operativo**, existen versiones para Windows, Linux y MacOs.

En el link a continuación, podrás descargar el MSI e instalarlo: [MongoDB Community Download](#).



The screenshot shows a web browser window with the URL mongodb.com/try/download/community. The page is titled "MongoDB Community Server". It features a brief introduction to the Community version of MongoDB, highlighting its flexible document data model and support for ad-hoc queries, secondary indexing, and real-time aggregations. Below this, there's a note about MongoDB Atlas and its managed services. A green "Try Free" button is visible at the top right. On the right side, there's a sidebar titled "Available Downloads" with dropdown menus for "Version" (set to "5.0.2 (current)"), "Platform" (set to "Windows"), and "Package" (set to "msi"). A large green "Download" button is prominently displayed. Below the sidebar, links to "Current releases & packages", "Development releases", "Archived releases", "Changelog", and "Release Notes" are listed. At the bottom left, a file download progress bar indicates "mongodb-windows.msi" is being downloaded at 99.2/274 MB, with 34 z. remaining. A "Mostrar todo" (Show all) button is at the bottom right.

Veremos que podemos instalar MongoDB como **servicio** o para arrancar en forma **manual** desde una consola. Asimismo, podemos seleccionar **MongoDBCompass** para instalar el cliente GUI de MongoDB propuesto por el instalador. La instalación por defecto se hará en esta carpeta e incluirá los siguientes archivos ejecutables por consola:

C:\Program Files\MongoDB\Server\5.0\bin				
Nombre	Fecha de modifica...	Tipo	Tamaño	
InstallCompass.ps1	3/8/2021 20:22	Script de Window...	2 KB	
mongo.exe	3/8/2021 21:51	Aplicación	21.646 KB	
mongod.cfg	3/8/2021 20:22	Archivo CFG	1 KB	
mongod.exe	3/8/2021 21:51	Aplicación	45.745 KB	
mongod.pdb	3/8/2021 21:51	Program Debug D...	516.908 KB	
mongos.exe	3/8/2021 21:11	Aplicación	29.047 KB	
mongos.pdb	3/8/2021 21:11	Program Debug D...	304.668 KB	

Una vez concluido el proceso, podremos arrancar el servicio de MongoDB con un solo comando ejecutado en la consola:

```
mongod --dbpath "path/de/la/base/de/datos"
```

Con este comando arrancamos el servicio **mongod**, que empezará a escuchar peticiones por el puerto **27017**.

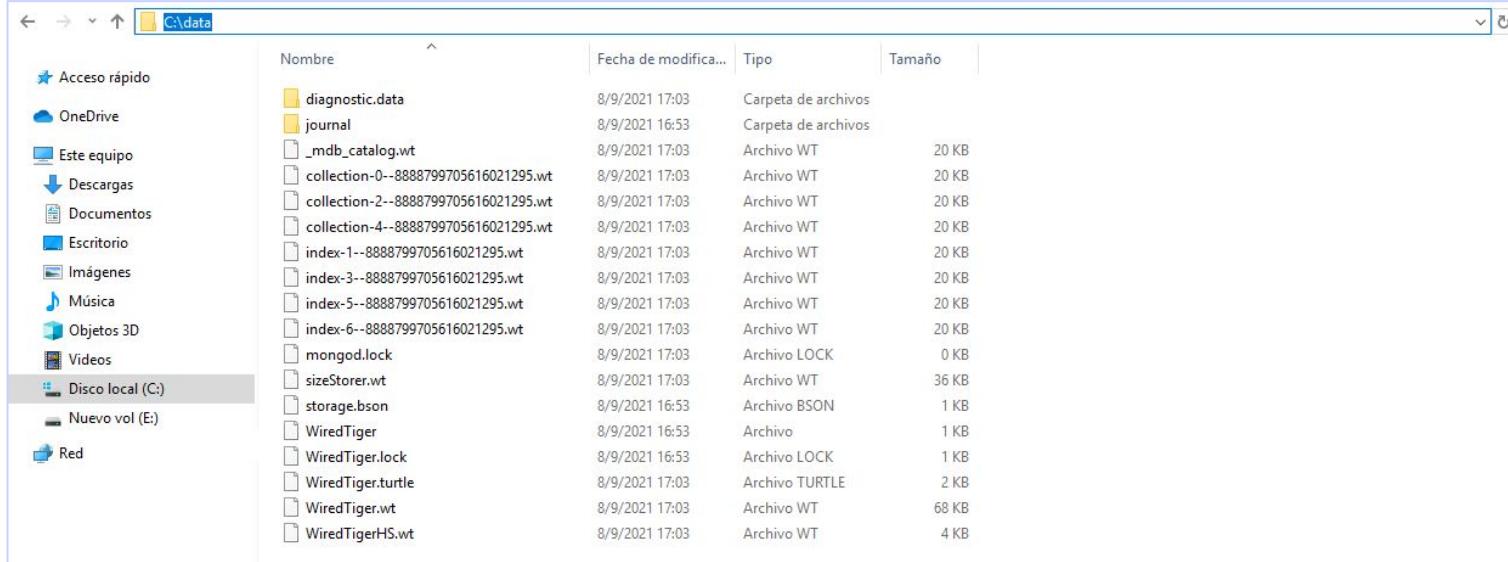


```
C:\cmder mini
λ mongod --dbpath data
{"t":{"$date":"2021-09-12T10:24:00.692-03:00"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"main","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2021-09-12T10:24:01.500-03:00"},"s":"I", "c":"NETWORK", "id":4915701, "ctx":"main","msg":"Initialized wire specification","attr":{"spec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":13}, "incomingInternalClient":{"minWireVersion":0,"maxWireVersion":13}, "outgoing":{"minWireVersion":0,"maxWireVersion":13}}}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.513-03:00"},"s":"W", "c":"ASIO", "id":22601, "ctx":"main","msg":"No TransportLayer configured during NetworkInterface startup"}
 {"t":{"$date":"2021-09-12T10:24:01.516-03:00"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"main","msg":"Implicit TCP FastOpen in use."}
 {"t":{"$date":"2021-09-12T10:24:01.558-03:00"},"s":"W", "c":"ASIO", "id":22601, "ctx":"main","msg":"No TransportLayer configured during NetworkInterface startup"}
 {"t":{"$date":"2021-09-12T10:24:01.562-03:00"},"s":"I", "c":"REPL", "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":{"TenantMigrationDonorService","ns":"config.tenantMigrationDonors"}}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.563-03:00"},"s":"I", "c":"REPL", "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":{"TenantMigrationRecipientService","ns":"config.tenantMigrationRecipients"}}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.565-03:00"},"s":"I", "c":"CONTROL", "id":4615611, "ctx":"initandlisten","msg":"MongoDB starting","attr":{"pid":18400,"port":27017,"dbPath":data,"architecture":64-bit,"host":Instructor-EducacionIT}}
 {"t":{"$date":"2021-09-12T10:24:01.565-03:00"},"s":"I", "c":"CONTROL", "id":23398, "ctx":"initandlisten","msg":"Target operating system minimum version","attr":{"targetMinOS":{"Windows 7/Windows Server 2008 R2"}}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.566-03:00"},"s":"I", "c":"CONTROL", "id":23403, "ctx":"initandlisten","msg":"Build Info","attr":{"buildInfo":{"version":5.0.2,"gitVersion":"6d9ec525e78465decadcf99cce953d380fedc8","modules":[],"allocator":"tcmalloc","environment":{"distmod":"windows","distarch":"x86_64","target_arch":"x86_64"}}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.566-03:00"},"s":"I", "c":"CONTROL", "id":51765, "ctx":"initandlisten","msg":"Operating System","attr":{"os":{"name":"Microsoft Windows 10","version":"10.0 (build 17763)"}}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.566-03:00"},"s":"I", "c":"CONTROL", "id":21951, "ctx":"initandlisten","msg":"Options set by command line","attr":{"options":{"storage":{"dbPath":data}}}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.589-03:00"},"s":"E", "c":"CONTROL", "id":20557, "ctx":"initandlisten","msg":"DBException in initAndListen, terminating","attr":{"error":{"NonExistentPath: Data directory data not found. Create the missing directory or specify another path using (1) the --dbpath command line option, or (2) by adding the 'storage.dbPath' option in the configuration file."}}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.589-03:00"},"s":"I", "c":"REPL", "id":4784900, "ctx":"initandlisten","msg":"Stepping down the ReplicationCoordinator for shutdown","attr":{"waitForReplicaSet":true,"waitTimeMillis":15000}}
 {"t":{"$date":"2021-09-12T10:24:01.597-03:00"},"s":"I", "c":"COMMAND", "id":4784901, "ctx":"initandlisten","msg":"Shutting down the MirrorMaestro"}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.598-03:00"},"s":"I", "c":"SHARDING", "id":4784902, "ctx":"initandlisten","msg":"Shutting down the WaitForMajorityService"}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.598-03:00"},"s":"I", "c":"NETWORK", "id":20562, "ctx":"initandlisten","msg":"Shutting down: going to close listening sockets"}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.598-03:00"},"s":"I", "c":"NETWORK", "id":4784905, "ctx":"initandlisten","msg":"Shutting down the global connection pool"}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.598-03:00"},"s":"I", "c":"CONTROL", "id":4784906, "ctx":"initandlisten","msg":"Shutting down the FlowControlTicketholder"}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.598-03:00"},"s":"I", "c":"FLOWCONTROL", "id":20520, "ctx":"initandlisten","msg":"Stopping further Flow Control ticket acquisitions."}, "isInternalClient":true}}
 {"t":{"$date":"2021-09-12T10:24:01.599-03:00"},"s":"I", "c":"NETWORK", "id":4784918, "ctx":"initandlisten","msg":"Shutting down the ReplicaSetMonitor"}, "isInternalClient":true}}
```

Es importante indicar el parámetro **--dbpath**, con la ruta donde se almacenarán los ficheros de nuestra base de datos. De no incluir este último, se necesitará tener una carpeta **/data/db** en la base del disco en el cual ejecutamos el comando (esta es la carpeta por default).

En el ejemplo, se creó la carpeta data en la base del disco **C:** y ahí se guardará la estructura de la base una vez ejecutado el comando **mongod**, como se muestra en la diapositiva siguiente.





	Nombre	Fecha de modifica...	Tipo	Tamaño
	diagnostic.data	8/9/2021 17:03	Carpeta de archivos	
	journal	8/9/2021 16:53	Carpeta de archivos	
	_mdb_catalog.wt	8/9/2021 17:03	Archivo WT	20 KB
	collection-0-8888799705616021295.wt	8/9/2021 17:03	Archivo WT	20 KB
	collection-2-8888799705616021295.wt	8/9/2021 17:03	Archivo WT	20 KB
	collection-4-8888799705616021295.wt	8/9/2021 17:03	Archivo WT	20 KB
	index-1-8888799705616021295.wt	8/9/2021 17:03	Archivo WT	20 KB
	index-3-8888799705616021295.wt	8/9/2021 17:03	Archivo WT	20 KB
	index-5-8888799705616021295.wt	8/9/2021 17:03	Archivo WT	20 KB
	index-6-8888799705616021295.wt	8/9/2021 17:03	Archivo WT	20 KB
	mongod.lock	8/9/2021 17:03	Archivo LOCK	0 KB
	sizeStorer.wt	8/9/2021 17:03	Archivo WT	36 KB
	storage.bson	8/9/2021 16:53	Archivo BSON	1 KB
	WiredTiger	8/9/2021 16:53	Archivo	1 KB
	WiredTiger.lock	8/9/2021 16:53	Archivo LOCK	1 KB
	WiredTiger.turtle	8/9/2021 17:03	Archivo TURTLE	2 KB
	WiredTiger.wt	8/9/2021 17:03	Archivo WT	68 KB
	WiredTigerHS.wt	8/9/2021 17:03	Archivo WT	4 KB

¿Cómo puedo usar la consola?

En caso de que ya tengamos el servidor lanzado en nuestra consola, alcanza con lanzar desde la consola el comando que sigue: **mongo localhost**.

Tras hacerlo, entraremos en la consola donde podremos realizar consultas.

En el ejemplo de la pantalla que sigue, ejecutamos el servidor mongod en una consola dentro de una ventana izquierda, mientras que a la derecha tenemos conectado el cliente mongo Shell:





The image shows a Cmder terminal window with two sessions running side-by-side.

Left Session (mongod):

```
C:\λ mongod --dbpath data
{"t": {"$date": "2021-09-08T16:53:55.130-03:00"}, "s": "I", "c": "NETWORK", "id": 4915701, "ctx": "main", "msg": "Initialized wire specification", "ttr": {"spec": {"incomingExternalClient": {"minWireVersion": 0, "maxWireVersion": 13}, "outgoing": {"minWireVersion": 0, "maxWireVersion": 13}, "isInternalClient": true}}}
{"t": {"$date": "2021-09-08T16:53:55.155-03:00"}, "s": "I", "c": "CONTROL", "id": 23285, "ctx": "main", "msg": "Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t": {"$date": "2021-09-08T16:53:55.966-03:00"}, "s": "W", "c": "ASIO", "id": 22601, "ctx": "main", "msg": "No TransportLayer configured during NetworkInterface startup"}
{"t": {"$date": "2021-09-08T16:53:55.968-03:00"}, "s": "I", "c": "NETWORK", "id": 4648602, "ctx": "main", "msg": "Implicit TCP FastOpen in use."}
{"t": {"$date": "2021-09-08T16:53:56.009-03:00"}, "s": "W", "c": "ASIO", "id": 22601, "ctx": "main", "msg": "No TransportLayer configured during NetworkInterface startup"}
{"t": {"$date": "2021-09-08T16:53:56.013-03:00"}, "s": "I", "c": "REPL", "id": 5123008, "ctx": "main", "msg": "Successfully registered PrimaryOnlyService", "attr": {"service": "TenantMigrationDonorService", "ns": "config.tenantMigrationDonors"} }
{"t": {"$date": "2021-09-08T16:53:56.015-03:00"}, "s": "I", "c": "REPL", "id": 5123008, "ctx": "main", "msg": "Successfully registered PrimaryOnlyService", "attr": {"service": "TenantMigrationRec
```

Right Session (mongo):

```
C:\λ mongo
MongoDB shell version v5.0.2
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("669e12d2-dac6-4e13-a387-af341c328655") }
MongoDB server version: 5.0.2
=====
Warning: the "mongo" shell has been superseded by "mongosh",
which delivers improved usability and compatibility. The "mongo" shell has been deprecated and will be removed in
an upcoming release.
We recommend you begin using "mongosh".
For installation instructions, see
https://docs.mongodb.com/mongodb-shell/install/
=====

The server generated these startup warnings when booting:
2021-09-08T16:53:56.141-03:00: Access control is not enabled for the database. Read and write access to data an
d configuration is unrestricted
2021-09-08T16:53:56.142-03:00: This server is bound to localhost. Remote systems will be unable to connect to t
his server. Start the server with --bind_ip <address> to specify which IP addresses it should serve responses from, or
with --bind_ip_all to bind to all interfaces. If this behavior is desired, start the server with --bind_ip 127.0.0.1 to
disable this warning
=====

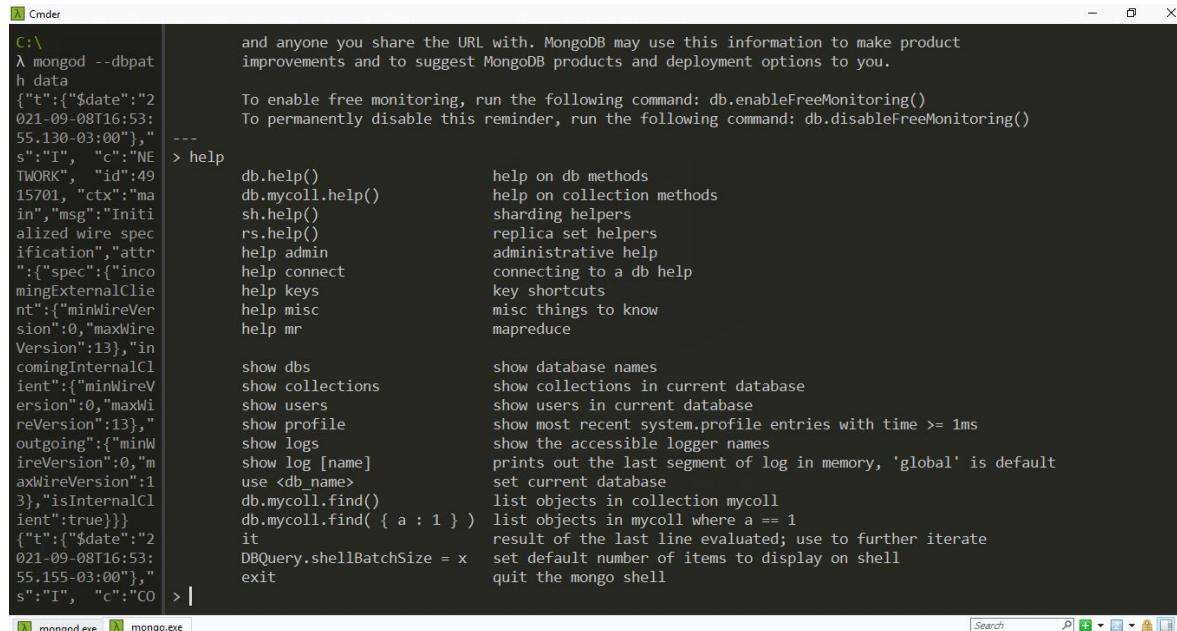
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

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()
> |
```

At the bottom of the terminal window, there is a toolbar with icons for file operations, search, and other utilities.

Si tipeamos `help`, se nos presentará un listado con los comandos más habituales junto a una descripción:



The screenshot shows a terminal window titled "Cmder" with a dark theme. The command entered is "mongod --dbpath h data". The output is a help menu for the MongoDB shell. It includes a reminder about sharing the URL with MongoDB for product improvements, and instructions for enabling and disabling free monitoring. The main part of the output lists various commands with their descriptions:

Command	Description
db.help()	help on db methods
db.mycoll.help()	help on collection methods
sh.help()	sharding helpers
rs.help()	replica set helpers
help admin	help admin
help connect	administrative help
help keys	connecting to a db help
help misc	key shortcuts
help mr	misc things to know
mapreduce	mapreduce
show dbs	show database names
show collections	show collections in current database
show users	show users in current database
show profile	show most recent system.profile entries with time >= 1ms
show logs	show the accessible logger names
show log [name]	prints out the last segment of log in memory, 'global' is default
use <db_name>	set current database
db.mycoll.find()	list objects in collection mycoll
db.mycoll.find({ a : 1 })	list objects in mycoll where a == 1
it	result of the last line evaluated; use to further iterate
DBQuery.shellBatchSize = x	set default number of items to display on shell
exit	quit the mongo shell

Razones técnicas para usar MongoDB

- MongoDB está en **permanente evolución** y en cada versión veremos nuevas mejoras.
- Desde el lado del desarrollo, el motivo principal para usar MongoDB es **su velocidad** y capacidad para agregar **registros con diferentes campos en una misma colección** de manera mucho más **flexible** de cómo haríamos en una base de datos relacional.
- **Su escalabilidad vertical y horizontal:** que una base de datos sea escalable significa que se va **adaptando, incrementando su tamaño en función de la información que guarda y los usuarios que la utilizarán**.

La **escalabilidad vertical** refiere al uso de **memoria y CPU**, es decir, la capacidad que tiene MongoDB para crear **nuevos nodos**. Este es un sistema distribuido en el que se pueden añadir nuevos nodos como si fueran replicaciones del propio MongoDB, un atributo que lleva a que la aplicación tenga un **mejor rendimiento**.

- **Flexibilidad (Schemaless):** al ser una base de datos que **no sigue un esquema**, MongoDB ofrece mucha flexibilidad. Este sistema difiere de las bases de datos relacionales donde se regulan determinados aspectos como de qué manera se almacena la información y qué tipos de datos se almacenan, entre otras cosas.
- **Transacciones:** pese a que las bases de datos NoSQL no suelen ofrecer transacciones como sí lo hacen las bases de datos relacionales, desde la versión **4.2**, MongoDB **sí las soporta**.
- **Proyectos Big Data:** para este tipo de proyectos las bases de datos NoSQL son **un excelente complemento**. Esto se debe a que son diseñadas para datos con estructuras que o bien no se conocen o son bastante dispares entre sí y los proyectos de Big Data suelen manejar este tipo de información.
- **Documentación oficial de calidad:** esto no es un detalle menor ya que podrás usar estos recursos para mejorar las operaciones y tareas que desarrolles.

Razones adicionales para usar MongoDB

- **Es de código abierto:** este tipo de software está **en permanente evolución y mejora** gracias a una comunidad.
- **Buen complemento para desarrollar en lenguajes basados en JavaScript:** si ya desarollas con Node.js, MongoDB es **un gran complemento** para esos proyectos.
- **Es fácil de aprender:** en poco tiempo podrás **alcanzar un nivel adecuado para desarrollar aplicaciones en tus proyectos.**
- **Es gratuita:** podrás incorporarlo a tus proyectos comerciales **sin pagar licencias.**
- Al estar escrito en C ++ **MongoDB** es multi-plataforma, por lo que podrás instalarlo fácilmente en distintos sistemas operativos.

Nota: Para instalar MongoDB en Ubuntu sencillamente debes correr: `sudo apt install -y mongodb` desde el terminal.

**¡Sigamos
trabajando!**