

## **Taller Mongo**

Jorge Esteban Herrera Jimenez – 833060

Corporación Universitaria Minuto de Dios

Bases de Datos Masivas

Ing. De Sistemas

William Alexander Matallana Porras

2025

## Contenido

Introducción :	3
1. ¿Qué tipo de base de datos es MongoDB y en qué se diferencia de una base de datos relacional como MySQL?	4
2. ¿Qué es una colección en MongoDB y en qué se diferencia de una tabla en SQL?	4
3. ¿Cómo se almacena la información en MongoDB y qué formato utiliza?	5
4. Explica la diferencia entre JSON y BSON en MongoDB.	5
5. Estructura de los archivos json:	6
6. ¿Qué ventajas tiene MongoDB sobre una base de datos relacional en términos de escalabilidad y flexibilidad?	8
7. Comandos para realizar CRUD en Mongo:	9
8. Cómo se pueden relacionar datos en Mongo sin usar joins como en sql.	10
9. Descargar imagen de mongo en Docker	11
10. Herramientas similares a Workbench para visualizar los datos de mongo.	12

## Introducción :

MongoDB es una base de datos NoSQL que es diferente a otras bases de datos relacionales como puede ser MYSQL, por su modelo que es basado en documento, en vez de tablas. Esto es lo que le permite almacenar datos de manera mas flexible en formato BSON, mejorando la optimización de rendimiento y la escalabilidad. En este trabajo se realizará un análisis de sus principales características, en donde también se incluirán la diferencia entre colecciones y tablas, la estructura de los archivos JSON, y las ventajas que ofrece a diferencia de las bases de datos relacionales.

Así mismo, se explicarán los comandos básicos para poder realizar operaciones CRUD, las formas en las que se relacionan los datos sin unas joins, y el proceso de la instalación de MongoDB mediante Docker. También revisaremos las herramientas similares a Workbench para visualizar datos y facilitar la gestión de bases de datos en MongoDB.

## 1. ¿Qué tipo de base de datos es MongoDB y en qué se diferencia de una base de datos relacional como MySQL?

MongoDB es una base de datos NoSQL que va orientada a documentos, la cual almacena datos en formato de documentos JSON, permitiendo una estructura mas flexible y sin esquemas predefinidos. Esto lo que hace es facilitar la gestión de datos no estructurados o semi-estructurados y su escalabilidad horizontal.

Por el contrario, una base de datos relacional como MYSQL, es una base de datos que organiza la información en tablas filas y columnas, siguiendo un esquemas más rígido y predefinido . Esta estructura puede garantizar la integridad referencial y es ideal para las aplicaciones que requieren transacciones complejas y consistencia estricta.

Mientras que MSQL hace uso de SQL(Structured Query Language) para la gestión y consulta de los datos, MongoDB emplea una sintaxis basada en JavaScript para sus operaciones.

## 2. ¿Qué es una colección en MongoDB y en qué se diferencia de una tabla en SQL?

Una colección en MongoDB e un conjunto de documentos que se agrupan debido a ciertas características comunes. Cada documento que esta dentro de una colección es una estructura de datos flexible que es capaz de contener diversos tipos de información, como textos, números o incluso otros documentos anidados. Esta flexibilidad le permite a los documentos que están dentro de una misma colección puedan tener diferentes capos y estructuras, adaptándose a las necesidades especificas de los datos que se manejen.

Por otra parte, las tablas en SQL son una estructura que es capaz de organizar los datos en filas

y columnas, siguiendo un esquema ya definido y rígido. En cada fila se representa un registro, y en cada columna un atributo específico del dato. Todos los registros en una tabla se deben ajustar al mismo esquema, lo que puede garantizar la consistencia y la integridad de los datos, pero esto puede limitar la flexibilidad para poder almacenar información que no se ajuste al modelo ya establecido.

La principal diferencia que existe entre una colección en MongoDB y una tabla en SQL es la flexibilidad del esquema. Las colecciones en MongoDB permiten que cada documento tenga una infraestructura única, mientras que las tablas de SQL requieren que todos los registros sigan una estructura uniforme. Esto le permite a MongoDB una mejor adaptabilidad para poder manejar datos no estructurados o semiestructurados, haciendo más fácil su uso en aplicaciones donde los requisitos de datos pueden ir evolucionando con el tiempo.

### 3. ¿Cómo se almacena la información en MongoDB y qué formato utiliza?

MongoDB almacena su información, haciendo uso de un formato de almacenamiento de documentos llamado BSON(Binary JSON), que es una forma binaria de JSON(JavaScript Object Notation). Este formato le permite representar estructuras de datos complejas, incluyendo arreglos y documentos anidados, de manera más eficiente. También, al ser binario, BSON le facilita una lectura y estructura más rápida a comparación de JSON, optimizando su rendimiento en la gestión de los datos.

### 4. Explica la diferencia entre JSON y BSON en MongoDB.

JSON es un formato de texto ligero y legible por humanos que se usa normalmente para intercambiar datos. Este representa estructuras de datos como objetos y arrays, siendo adoptado por muchas aplicaciones web para la comunicación entre cliente y servidor.

Por otra parte, BSON es una representación binaria de estructuras de datos similares a JSON, fue diseñada para ser eficiente en términos de espacio y velocidad de codificación y decodificación. BSON lo que hace es añadir soporte para los tipos de datos adicionales que no son presentes en JSON, como pueden ser fechas y datos binarios.

Las principales diferencias que hay entre JSON y BSON son:

- **Formato:** JSON es un formato que esta basado en texto y es legible por los humanos, mientras que BSON es una representación binaria que no es directamente legible.
- **Tipos de datos:** BSON admite una gama mas grande de tipos de datos, permitiendo enteros de 32 y 64 bits, fechas y datos binarios, lo que no es posible en JSON.
- **Eficiencia:** Debido a la naturaleza binaria de BSON, permite una codificación y decodificación mas veloz, lo que ayuda a mejorar el rendimiento en operaciones de lectura y escritura , especialmente con grandes volúmenes de datos.

## 5. Estructura de los archivos json:

La estructura de los archivos JSON se basa en 2 componentes principales:

- **Objetos:** Estos son colecciones desordenadas de pares clave/valor. Cada clave es una cadena de texto que debe estar entre comillas dobles, y cada valor puede ser una cadena, numero, booleano, null, objeto o una matriz. Los objetos se delimitan con llaves { } y los pares clave/valor se separan por comas. Por ejemplo:

```
{  
  'nombre': 'Facundo',  
  'edad': 30,  
  'ciudad': 'Madrid'  
}
```

- Arreglos: Son listas ordenadas de valores, que pueden ser de cualquier tipo permitido en JSON, incluyendo otros objetos o matrices. Las matrices se encierran entre corchetes [] y los valores se separan por comas. Por ejemplo:

```
{  
  'nombre': 'Facundo',  
  'edad': 30,  
  'ciudad': 'Madrid'  
},  
{  
  'nombre': 'Saturnina',  
  'edad': 98,  
  'ciudad': 'Barcelona'  
},  
{  
  'nombre': 'Adoración',  
  'edad': 55,  
  'ciudad': 'Sevilla'  
}
```

## 6. ¿Qué ventajas tiene MongoDB sobre una base de datos relacional en términos de escalabilidad y flexibilidad?

MongoDB ofrece ventajas significativas sobre las bases de datos relacionales tradicionales en términos de **escalabilidad** y **flexibilidad**:

### **Flexibilidad:**

- **Esquema dinámico:** A diferencia de las bases de datos relacionales que requieren un esquema fijo, MongoDB permite almacenar documentos con estructuras variadas dentro de una misma colección. Esto facilita la adaptación a cambios en los requisitos de datos sin necesidad de reestructurar la base de datos.
- **Almacenamiento de datos no estructurados:** MongoDB es ideal para gestionar datos no estructurados o semiestructurados, como documentos, registros de actividad o contenido generado por usuarios, sin la rigidez de un esquema predefinido.

### **Escalabilidad:**

- **Escalabilidad horizontal:** MongoDB está diseñado para escalar horizontalmente mediante la adición de más servidores al clúster, distribuyendo la carga de trabajo y mejorando el rendimiento general. Esta capacidad es especialmente útil para aplicaciones que manejan grandes volúmenes de datos o experimentan un crecimiento rápido.
- **Sharding:** Esta técnica permite dividir y distribuir datos en múltiples servidores, garantizando una alta disponibilidad y un acceso rápido a la información, incluso en entornos con cargas de trabajo intensivas.



## 7. Comandos para realizar CRUD en Mongo:

En MongoDB, las operaciones CRUD (**Crear**, **Leer**, **Actualizar** y **Eliminar**) se realizan utilizando métodos específicos que permiten gestionar documentos dentro de las colecciones. A continuación, se detallan los comandos básicos para cada operación:

- Crear (Create):

- **insertOne()**: Inserta un único documento en una colección

```
db.collectionName.insertOne()
```

- **insertMany()**: Inserta múltiples documentos en una colección.

```
db.collectionName.insertMany();
```

- Leer (Read):

- **find()**: Recupera documentos que coinciden con un criterio de búsqueda.

```
db.collectionName.find(query, projection)
```

- **findOne()**: Recupera un único documento que coincide con un criterio de búsqueda.

```
db.collectionName.findOne()
```

- Actualizar (Update):

- **updateOne()**: Actualiza el primer documento que coincide con un criterio de

búsqueda.

```
db.collectionName.updateOne(filter, update, options)
```

- **updateMany()**: Actualiza todos los documentos que coinciden con un criterio de búsqueda.

```
db.collectionName.updateMany(filter, update, options)
```

- Eliminar (Delete):

- **deleteOne()**: Elimina el primer documento que coincide con un criterio de búsqueda.

```
db.collectionName.deleteOne(filter, options)
```

- **deleteMany()**: Elimina todos los documentos que coinciden con un criterio de búsqueda.

```
db.collectionName.deleteMany(filter, options)
```

## 8. Cómo se pueden relacionar datos en Mongo sin usar joins como en sql.

En MongoDB, aunque no existen las operaciones `JOIN` como en las bases de datos relacionales, es posible establecer relaciones entre colecciones utilizando dos enfoques principales: **documentos incrustados** y **referencias manuales**. La elección entre estos métodos depende de las necesidades específicas de la aplicación y de cómo se estructuran los datos.

### 1. Documentos Incrustados:

Este enfoque consiste en almacenar documentos relacionados directamente dentro de un documento principal. Es especialmente útil cuando la relación es de uno a uno o de uno a muchos, y los datos relacionados se consultan frecuentemente juntos.

## 2. Referencias Manuales:

En situaciones donde los datos relacionados son grandes, se actualizan con frecuencia o se comparten entre múltiples documentos, es más eficiente utilizar referencias manuales. Este método implica almacenar el `_id` de un documento en otro documento para establecer la relación.

## 9. Descargar imagen de mongo en Docker

Para descargar mongo en Docker, abrimos el cmd como administrador y ejecutamos el comando: `docker pull mongo`

```
C:\Windows\System32>docker pull mongo
Using default tag: latest
latest: Pulling from library/mongo
7afa02f8c09e: Download complete
d67c4ebf9460: Download complete
342a4f4728ff: Download complete
5a7813e071bf: Download complete
4e7ca17a42bd: Download complete
d5bafd14fbe8: Download complete
734719e891c0: Download complete
0c492c8e8cfd: Download complete
Digest: sha256:7bd28e5eea1c5766a084d5818254046f3ebe3b8f20a65e3a274640189e296667
Status: Downloaded newer image for mongo:latest
docker.io/library/mongo:latest
```

Y se revisa que la imagen se haya descargado correctamente:

```
C:\Windows\System32>docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
mongo	latest	7bd28e5eea1c	7 days ago	1.19GB
mysql	latest	146682692a3a	8 weeks ago	1.09GB

## 10. Herramientas similares a Workbench para visualizar los datos de mongo.

Para gestionar y visualizar datos en MongoDB, existen varias herramientas similares a MySQL Workbench que ofrecen interfaces gráficas intuitivas y funcionalidades avanzadas. A continuación, se presentan algunas opciones destacadas:

- **MongoDBCompass:**

La herramienta oficial de MongoDB proporciona una interfaz gráfica que permite explorar y analizar datos, así como optimizar consultas y gestionar índices. Es especialmente útil para usuarios que buscan una integración directa con las características nativas de MongoDB.

- **DBeaver:**

Este cliente universal de bases de datos es compatible con una amplia variedad de sistemas, incluyendo MongoDB. Ofrece una versión de código abierto y otra comercial, brindando herramientas para administración, desarrollo y análisis de bases de datos.

- **Navicat-for-MongoDB:**

Esta herramienta de gestión de bases de datos ofrece una interfaz amigable y funciones avanzadas para profesionales que trabajan con MongoDB, facilitando la administración de grandes conjuntos de datos y flujos de trabajo complejos.

## Referencias:

1. <https://datascientest.com/es/mongodb-todo-sobre-la-base-de-datos-nosql-orientada-a-documentos#:~:text=ingenier%C3%ADa%20de%20datos.-.MongoDB%20es%20una%20base%20de%20datos%20NoSQL%20orientada%20a%20documentos,almacenar%20como%20colecciones%20y%20documentos>
2. <https://www.ionos.com/es-us/digitalguide/paginas-web/desarrollo-web/mongodb-create-collection/>  
  
<https://www.mongodb.com/es/resources/basics/databases/nosql-explained/nosql-vs-sql>
3. <https://www.purestorage.com/es/knowledge/what-is-mongodb.html#:~:text=MongoDB%20usa%20un%20formato%20de,acomodar%20m%C3%A1s%20tipos%20de%20datos.>
4. <https://www.mongodb.com/es/resources/basics/json-and-bson>
5. <https://www.arsys.es/blog/formato-json-que-es-y-para-que-sirve#:~:text=Los%20objetos%20en%20JSON%20son,s%C3%AD%20por%20comas%20%C2%AB%C2%BB.>
6. [https://www.dongee.com/tutoriales/ventajas-y-desventajas-de-mongodb/?utm\\_source=chatgpt.com](https://www.dongee.com/tutoriales/ventajas-y-desventajas-de-mongodb/?utm_source=chatgpt.com)
7. <https://www.scaler.com/topics/crud-operations-in-mongodb/>
8. <https://medium.com/%40diego.coder/relaciones-en-mongodb-edf2107a94ad>  
<https://studio3t.com/es/knowledge-base/articles/mongodb-aggregation-framework/>

