

Taller Mongo.

Base de Datos Masivas

Juan David Moreno Pico

ID:852417

Docente. Alex Matallana

Ingeniería de Sistemas.

Corporación Universitaria Uniminuto.

Zipaquirá

2025

Contenido

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

Objetivos

El objetivo principal es comprender las diferencias entre MongoDB y MySQL, destacando las ventajas de MongoDB como base de datos NoSQL en términos de escalabilidad, flexibilidad y rendimiento. También se busca analizar su estructura de almacenamiento basada en documentos BSON dentro de colecciones, diferenciándola del modelo relacional de SQL. Además, se pretende explicar el formato JSON y BSON, su aplicación en MongoDB y cómo este último mejora la eficiencia en el almacenamiento y procesamiento de datos.

Otro objetivo clave es aprender a realizar operaciones CRUD en MongoDB mediante comandos básicos para insertar, consultar, actualizar y eliminar datos. Asimismo, se busca entender cómo se pueden relacionar datos sin utilizar joins, explorando técnicas como referencias manuales, documentos embebidos y agregaciones con \$lookup. Finalmente, se busca configurar MongoDB en Docker y conocer herramientas gráficas como MongoDB Compass, Studio 3T y Robo 3T, facilitando la administración y visualización de bases de datos.

Desarrollo de Ejercicio.

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

Específicamente una base de datos orientada a documentos. Se diferencia de una base de datos relacional como MySQL en que almacena datos en documentos JSON o BSON en lugar de tablas con filas y columnas. No requiere un esquema fijo, lo que permite mayor flexibilidad. Además, MongoDB escala horizontalmente mediante sharding, mientras que MySQL suele escalar verticalmente.

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

Una colección en MongoDB es un grupo de documentos, similar a una tabla en SQL. Sin embargo, a diferencia de una tabla, los documentos dentro de una colección no necesitan tener la misma estructura. Esto permite que los documentos en una misma colección tengan diferentes campos y tipos de datos, lo que no es posible en una tabla SQL, donde todas las filas deben seguir un esquema definido.

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

MongoDB almacena la información en documentos que utilizan un formato llamado BSON (Binary JSON). BSON es una representación binaria de JSON, lo que permite un almacenamiento más eficiente y un acceso más rápido a los datos. Cada

documento es una estructura de datos que contiene pares de clave-valor, similar a un objeto JSON.

Explica la diferencia entre JSON y BSON en MongoDB.

JSON (JavaScript Object Notation) es un formato de texto ligero para el intercambio de datos, fácil de leer para los humanos. BSON (Binary JSON) es una extensión binaria de JSON, diseñada para ser más eficiente en términos de almacenamiento y procesamiento. BSON incluye tipos de datos adicionales, como fechas y binarios, que no están disponibles en JSON.

Estructura de los archivos json

Los archivos JSON tienen una estructura de pares clave-valor, donde las claves son cadenas de texto y los valores pueden ser cadenas, números, arrays, objetos, booleanos o null. Por ejemplo:

```
{
  "nombre": "Juan",
  "edad": 30,
  "ciudad": "Madrid",
  "hobbies": ["leer", "correr", "viajar"]
}
```

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

Escalabilidad: MongoDB está diseñado para escalar horizontalmente, lo que significa que puede distribuir datos a través de múltiples servidores (sharding) para manejar grandes volúmenes de datos y tráfico.

Flexibilidad: MongoDB no requiere un esquema fijo, lo que permite que los documentos en una colección tengan diferentes estructuras. Esto facilita la evolución de la aplicación y la adaptación a cambios en los requisitos de datos.

Comandos para realizar CRUD en Mongo

Insertar:

```
db.usuarios.insertOne({ nombre: "Juan", edad: 30 });  
db.usuarios.insertMany([ { nombre: "Ana" }, { nombre: "Pedro" } ]);
```

Leer/consultar:

```
db.usuarios.find();  
db.usuarios.find({ edad: { $gt: 25 } });
```

Actualizar:

```
db.usuarios.updateOne({ nombre: "Juan" }, { $set: { edad: 31 } });  
db.usuarios.updateMany({}, { $set: { activo: true } });
```

Eliminar:

```
db.usuarios.deleteOne({ nombre: "Pedro" });  
db.usuarios.deleteMany({ edad: { $lt: 18 } });
```

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

En MongoDB, los datos se pueden relacionar mediante referencias o embedding (incrustación).

Referencias: Se almacena un identificador único (ObjectId) en un documento que referencia a otro documento en otra colección.

```
{  
  "nombre": "Juan",  
  "curso_id": ObjectId("603d3b3f9e1a7c0015f6aef3")  
}
```

Embedding: Se incrustan documentos directamente dentro de otros documentos.

Esto es útil cuando la relación es uno a uno o uno a pocos.

```
{
  "nombre": "Juan",
  "curso": { "nombre": "Matemáticas", "profesor": "Sr. García" }
}
```

Descargar imagen de mongo en Docker

Para descargar la imagen oficial de MongoDB en Docker, puedes usar el siguiente comando:

```
docker pull mongo
```

Luego, puedes ejecutar un contenedor con:

```
docker run -d --name mi-mongo -p 27017:27017 mongo
```

Herramientas similares a Workbench para visualizar los datos de mongo.

- MongoDB Compass (Oficial de MongoDB)
- Robo 3T (RoboMongo)
- NoSQLBooster for MongoDB
- Studio 3T
- Humongous (Extensión de VS Code)

CONCLUSIONES

MongoDB se presenta como una alternativa eficiente a las bases de datos relacionales, ofreciendo mayor flexibilidad y escalabilidad gracias a su modelo basado en documentos BSON. Su capacidad para manejar grandes volúmenes de datos sin requerir una estructura rígida lo convierte en una opción ideal para aplicaciones modernas que demandan agilidad y rendimiento. Además, su integración con herramientas como MongoDB Compass y Studio 3T facilita la gestión y visualización de datos.

A lo largo de este trabajo, se exploraron las principales características de MongoDB, su estructura de almacenamiento, operaciones CRUD y formas de relacionar datos sin joins. También se abordó su implementación en Docker y la comparación con MySQL. En conclusión, MongoDB es una base de datos versátil que, dependiendo del caso de uso, puede ser una mejor opción que las bases de datos relacionales, especialmente en entornos donde la escalabilidad y la flexibilidad son prioritarias.

NETGRAFÍA

1. <https://www.mongodb.com/docs/>
<https://www.mongodb.com/docs/manual/crud/>
2. <https://www.mongodb.com/docs/manual/reference/bson-types/>
3. https://hub.docker.com/_/mongo
4. <https://www.mongodb.com/docs/manual/aggregation/>
5. <https://www.mongodb.com/products/compass>
6. <https://studio3t.com/>