



Índice

Índice	1
1. Introducción	2
2. Qué es una base de datos documental	2
5. Qué es MongoDB	3
6. Componentes principales de MongoDB	3
7. Instalación de MongoDB en Windows	3
Paso 1: Descargar el instalador	3
Paso 2: Ejecutar el instalador	3
Paso 3: Instalar MongoDB Compass	4
Paso 4: Verificar la instalación	4
8. Primeros pasos con MongoDB Compass	4
9. Crear una base de datos y colección	4
10. Insertar, consultar, actualizar y eliminar documentos en MongoDB (detallado)	5
10.1 Insertar documentos	5
10.2 Consultar documentos (Filtros básicos y avanzados)	5
10.3 Operadores de filtrado importantes	6
10.4 Actualizar documentos	7
10.5 Eliminar documentos	8
10.6 Ordenar, limitar y contar resultados	8
10.7 Ejemplo completo: combinación de filtros y orden	9

1. Introducción

En el mundo actual, los datos son el motor de casi todas las aplicaciones. Existen diferentes tipos de bases de datos según cómo almacenan y gestionan la información. Las más tradicionales son las **bases de datos relacionales (SQL)**, pero con la llegada de los grandes volúmenes de datos y los modelos más flexibles, surgieron las **bases de datos NoSQL**.

Entre las bases NoSQL, uno de los modelos más populares es el **modelo documental**, y la herramienta más utilizada en este ámbito es **MongoDB**.

2. Qué es una base de datos documental

Una base de datos documental almacena información en **documentos estructurados** (normalmente en formato **JSON** o **BSON**).

Cada documento representa una entidad completa (por ejemplo, un cliente, un pedido o un producto) y puede contener estructuras anidadas.

En lugar de tablas, filas y columnas, las bases documentales usan:

- **Base de datos** → conjunto de colecciones.
- **Colección** → conjunto de documentos.
- **Documento** → unidad básica de información (similar a una fila, pero flexible).

3. Diferencias con las bases de datos relacionales

Característica	Relacional (SQL)	Documental (MongoDB)
Estructura	Tablas y filas fijas	Colecciones de documentos
Esquema	Fijo	Flexible

UT 6 - Introducción a bases de datos documentales con MongoDB

Profesor: *Luis Javier López López*

Consultas	SQL	Propio de MongoDB (JSON)
Relaciones	Join entre tablas	Documentos embebidos o referencias
Escalado	Vertical	Horizontal (sharding)

5. Qué es MongoDB

MongoDB es una **base de datos documental NoSQL** de código abierto.

Está diseñada para almacenar datos en formato **BSON** (una versión binaria de JSON) y ofrece:

- Alta disponibilidad mediante *replica sets*.
- Escalabilidad horizontal mediante *sharding*.
- Acceso desde interfaces gráficas (como MongoDB Compass) o desde la consola.

6. Componentes principales de MongoDB

- **mongod** → el proceso principal del servidor MongoDB.
- **mongosh** → shell interactivo para ejecutar comandos.
- **MongoDB Compass** → interfaz gráfica para visualizar y gestionar las bases de datos.

7. Instalación de MongoDB en Windows

Paso 1: Descargar el instalador

1. Ve al sitio oficial:
<https://www.mongodb.com/try/download/community>
2. Selecciona:
 - Versión: *MongoDB Community Server*
 - Plataforma: *Windows*
 - Tipo de paquete: *MSI Installer*
3. Descarga el archivo **.msi**.

Paso 2: Ejecutar el instalador

UT 6 - Introducción a bases de datos documentales con MongoDB

Profesor: *Luis Javier López López*

1. Abre el instalador descargado.
2. Selecciona **Complete** cuando se te pregunte el tipo de instalación.
3. Activa la casilla *Install MongoDB as a Service*.
4. Acepta el resto de las opciones por defecto.
5. Espera a que finalice la instalación.

Paso 3: Instalar MongoDB Compass

En las versiones recientes del instalador puedes marcar la casilla para instalar **MongoDB Compass** directamente.

Si no lo hiciste durante la instalación, puedes descargarlo desde:

<https://www.mongodb.com/products/compass>

Paso 4: Verificar la instalación

1. Abre una ventana de comandos (cmd o PowerShell).

Escribe: `mongosh`

2. Si aparece el prompt de MongoDB (>), la instalación fue correcta.

Escribe: `show dbs`

3. Verás las bases de datos por defecto (`admin`, `config`, `local`).

8. Primeros pasos con MongoDB Compass

1. Abre **MongoDB Compass**.

En la ventana de conexión, escribe: `mongodb://localhost:27017`

2. (Esta es la conexión al servidor local).
3. Pulsa **Connect**.
4. Verás la lista de bases de datos locales.
5. Puedes crear una nueva base de datos haciendo clic en **Create Database**.

9. Crear una base de datos y colección

1. En Compass, pulsa **Create Database**.
2. Nombra la base de datos, por ejemplo: `escuela`.
3. Nombra la primera colección, por ejemplo: `alumnos`.
4. Pulsa **Create Database**.

10. Insertar, consultar, actualizar y eliminar documentos en MongoDB (detallado)

10.1 Insertar documentos

Operación	Comando	Descripción
Insertar un solo documento	<code>db.alumnos.insertOne({ "nombre": "Laura", "edad": 22, "curso": "BD I" })</code>	Inserta un documento en la colección <code>alumnos</code> .
Insertar varios documentos	<code>db.alumnos.insertMany([{ "nombre": "Carlos", "edad": 23, "curso": "BD I" }, { "nombre": "Ana", "edad": 21, "curso": "BD II" }])</code>	Inserta varios documentos a la vez.

10.2 Consultar documentos (Filtros básicos y avanzados)

Operación	Comando	Descripción
Consultar todos los documentos	<code>db.alumnos.find()</code>	Muestra todos los documentos de la colección.
Consultar documentos por igualdad	<code>db.alumnos.find({ "curso": "BD I" })</code>	Devuelve alumnos cuyo curso es "BD I".

UT 6 - Introducción a bases de datos documentales con MongoDB

Profesor: Luis Javier López López

Consultar documentos por condición	<code>db.alumnos.find({ "edad": { "\$gt": 21 } })</code>	Alumnos con edad mayor a 21 años.
Consultar con AND	<code>db.alumnos.find({ "curso": "BD I", "edad": { "\$gt": 21 } })</code>	Alumnos de BD I con edad mayor a 21 años.
Consultar con OR	<code>db.alumnos.find({ "\$or": [{ "curso": "BD I" }, { "edad": 21 }] })</code>	Alumnos de BD I o con edad 21.
Consultar con NOT	<code>db.alumnos.find({ "curso": { "\$ne": "BD I" } })</code>	Alumnos que no estén en BD I.
Consultar con rangos	<code>db.alumnos.find({ "edad": { "\$gte": 21, "\$lte": 23 } })</code>	Alumnos entre 21 y 23 años inclusive.
Consultar campos específicos	<code>db.alumnos.find({}, { "nombre": 1, "curso": 1, "_id": 0 })</code>	Devuelve solo los campos nombre y curso , ocultando _id .

10.3 Operadores de filtrado importantes

Operador	Ejemplo	Descripción
\$eq	<code>{ "edad": { "\$eq": 22 } }</code>	Igual a 22
\$ne	<code>{ "curso": { "\$ne": "BD I" } }</code>	Diferente de "BD I"

UT 6 - Introducción a bases de datos documentales con MongoDB

Profesor: **Luis Javier López López**

\$gt	<code>{ "edad": { "\$gt": 21 } }</code>	Mayor que 21
\$gte	<code>{ "edad": { "\$gte": 21 } }</code>	Mayor o igual que 21
\$lt	<code>{ "edad": { "\$lt": 23 } }</code>	Menor que 23
\$lte	<code>{ "edad": { "\$lte": 23 } }</code>	Menor o igual que 23
\$in	<code>{ "curso": { "\$in": ["BD I", "BD II"] } }</code>	Valor dentro de un array
\$nin	<code>{ "curso": { "\$nin": ["BD III"] } }</code>	Valor NO dentro de un array
\$exists	<code>{ "telefono": { "\$exists": true } }</code>	Documentos que tienen el campo telefono
\$regex	<code>{ "nombre": { "\$regex": "^A" } }</code>	Nombres que empiezan con "A"

10.4 Actualizar documentos

Operación	Comando	Descripción
Actualizar un campo	<code>db.alumnos.updateOne({ "nombre": "Ana" }, { "\$set": { "curso": "BD Avanzadas" } })</code>	Cambia el curso de Ana.
Actualizar varios documentos	<code>db.alumnos.updateMany({ "curso": "BD I" }, { "\$set": { "nivel": "Principiante" } })</code>	Agrega campo nivel a todos los alumnos de BD I.

UT 6 - Introducción a bases de datos documentales con MongoDB

Profesor: Luis Javier López López

Incrementar valor numérico	<code>db.alumnos.updateOne({ "nombre": "Laura" }, { "\$inc": { "edad": 1 } })</code>	Suma 1 a la edad de Laura.
Renombrar campo	<code>db.alumnos.updateOne({ "nombre": "Carlos" }, { "\$rename": { "curso": "materia" } })</code>	Cambia el nombre del campo <code>curso</code> a <code>materia</code> .

10.5 Eliminar documentos

Operación	Comando	Descripción
Eliminar un documento	<code>db.alumnos.deleteOne({ "nombre": "Carlos" })</code>	Borra el primer documento que coincida.
Eliminar varios documentos	<code>db.alumnos.deleteMany({ "curso": "BD II" })</code>	Borra todos los documentos del curso BD II.
Eliminar toda la colección	<code>db.alumnos.drop()</code>	Borra toda la colección <code>alumnos</code> .

10.6 Ordenar, limitar y contar resultados

Operación	Comando	Descripción
Ordenar ascendente	<code>db.alumnos.find().sort({ "edad": 1 })</code>	Ordena por edad de menor a mayor
Ordenar descendente	<code>db.alumnos.find().sort({ "edad": -1 })</code>	Ordena por edad de mayor a menor



UT 6 - Introducción a bases de datos documentales con MongoDB

Profesor: Luis Javier López López

Limitar resultados	<code>db.alumnos.find().limit(2)</code>	Muestra solo 2 documentos
Saltar documentos	<code>db.alumnos.find().skip(1)</code>	Omite el primer documento
Contar documentos	<code>db.alumnos.countDocuments({ "curso": "BD I" })</code>	Cuenta los alumnos de BD I

10.7 Ejemplo completo: combinación de filtros y orden

```
db.alumnos.find(  
  { "curso": "BD I", "edad": { "$gte": 21 } }, // filtro  
  { "nombre": 1, "edad": 1, "_id": 0 }        // campos a mostrar  
)  
.sort({ "edad": -1 }) // orden descendente  
.limit(5)             // máximo 5 resultados
```