



# Bases de datos NoSQL

202  
4

# CONTENIDO

1. Comprender qué son las operaciones CRUD.
2. Aprender a crear y seleccionar bases de datos en MongoDB.
3. Introducción a las colecciones en MongoDB.

# ¿Qué es MongoDB?



{ name: mongo, type: DB }

- MongoDB almacena datos en documentos flexibles similares a JSON, por lo que los campos pueden variar entre documentos y la estructura de datos puede cambiarse con el tiempo}
- El modelo de documento se asigna a los objetos en el código de su aplicación para facilitar el trabajo con los datos
- Las consultas ad hoc, la indexación y la agregación en tiempo real ofrecen maneras potentes de acceder a los datos y analizarlos
- MongoDB es una base de datos distribuida en su núcleo, por lo que la alta disponibilidad, la escalabilidad horizontal y la distribución geográfica están integradas y son fáciles de usar
- MongoDB es de uso gratuito. Las versiones lanzadas antes del 16 de octubre de 2018 se publican bajo licencia AGPL.

# Colecciones de datos y Documentos

## ¿Cómo funciona?

MongoDB

SQL

**Base de Datos**

**Base de Datos**

**Colecciones**

**Tablas**

**Documentos**

**Filas (registros)**

**Campos**

**Columnas**

# •Ejercicios

## Ejercicio 1: Creación de Base de Datos y Colecciones

Objetivo: Familiarizarse con la creación de bases de datos y colecciones en MongoDB.

### Instrucciones:

Crea una base de datos llamada Biblioteca.

Dentro de la base de datos Biblioteca, crea dos colecciones: Libros y Visitantes.

Asegúrate de que la base de datos y las colecciones se hayan creado correctamente usando MongoDB Compass.

# •Ejercicios

## Ejercicio 2: Inserción de Documentos

Objetivo: Aprender a insertar documentos en una colección.

### Instrucciones:

Inserta tres documentos en la colección Libros. Cada documento debe tener los siguientes campos:

titulo: El título del libro.

autor: El nombre del autor del libro.

anio: El año de publicación del libro.

disponible: Un valor booleano que indica si el libro está disponible para préstamo.

Inserta dos documentos en la colección Visitantes. Cada documento debe tener los siguientes campos:

nombre: El nombre del visitante.

email: El correo electrónico del visitante.

fechaRegistro: La fecha en que el visitante se registró en la biblioteca.



## •Ejercicios

### Ejercicio 3: Actualización de Documentos

Objetivo: Practicar la actualización de documentos en una colección.

#### Instrucciones:

Actualiza un libro en la colección Libros para cambiar su estado de disponible a false.  
Añade un nuevo campo a todos los documentos en la colección Visitantes llamado miembroActivo, estableciendo su valor en true.

## •Ejercicios

### Ejercicio 4: Eliminación de Documentos

Objetivo: Aprender a eliminar documentos de una colección.

#### Instrucciones:

Elimina un libro de la colección Libros que tenga más de 20 años de antigüedad.

Elimina un visitante de la colección Visitantes que no haya visitado la biblioteca en el último año.



## •Ejercicios

### Ejercicio 5: Consultas Simples y Uso de Filtros

Objetivo: Realizar consultas utilizando filtros condicionales.

#### Instrucciones:

Realiza una consulta para encontrar todos los libros que estén disponibles para préstamo.  
Utiliza un filtro para buscar todos los visitantes que se hayan registrado en la biblioteca en los últimos tres meses.

## •Ejercicios

### Ejercicio 6: Agregar un Nuevo Campo a Todos los Documentos

- ◆ Se ha decidido registrar el número de copias disponibles de cada libro en la colección Libros.

 Objetivo: Agregar un campo copiasDisponibles con un valor inicial de 5 a todos los documentos en la colección Libros.

# •Ejercicios

## Ejercicio 7: Agregar Comentarios a los Libros

- ♦ Ahora queremos agregar comentarios de usuarios a los libros en la colección Libros.

 Objetivo: Usar el operador \$push para agregar un nuevo comentario.

## •Ejercicios

### Ejercicio 8: Contar el Número de Libros Disponibles

- ♦ Se necesita saber cuántos libros están disponibles para préstamo.

 Objetivo: Contar los documentos donde disponible: true.

MUCHAS GRACIAS!



inacap.cl