

Manual Básico de MongoDB

1. Crear una Base de Datos

Para crear una base de datos en MongoDB, puedes usar el comando `use`. Si la base de datos no existe, se creará automáticamente.

```
use escuela
```

Aquí hemos creado una base de datos llamada `escuela`.

2. Crear una Colección

Una colección en MongoDB es similar a una tabla en bases de datos relacionales. Puedes crear una colección simplemente insertando un documento en ella.

- Crear una colección para `estudiantes`:

```
db.createCollection("estudiantes")
```

Aunque MongoDB creará automáticamente la colección al insertar un documento, usar `createCollection` te da más control sobre su configuración.

3. Insertar Documentos

Los documentos se insertan en una colección usando `insertOne()` o `insertMany()`.

- **Insertar un solo estudiante:**

```
db.estudiantes.insertOne({
  "nombre": "Juan Pérez",
  "edad": 20,
  "carrera": "Ingeniería de Sistemas"
})
```

Esto insertará un documento con los datos del estudiante `Juan Pérez` en la colección `estudiantes`.

- **Insertar varios estudiantes a la vez:**

```
db.estudiantes.insertMany([
  { "nombre": "Ana Gómez", "edad": 22, "carrera": "Matemáticas" },
  { "nombre": "Carlos López", "edad": 21, "carrera": "Física" }
])
```

Con `insertMany()`, se pueden insertar varios documentos a la vez.

4. Crear Colecciones de Materias y Matrículas

- Crear una colección `materias` e insertar algunos documentos:

```
db.materias.insertMany([
  { "nombre": "Matemáticas", "codigo": "MAT101" },
  { "nombre": "Física", "codigo": "FIS101" },
  { "nombre": "Programación", "codigo": "PRO101" }
])
```

- Crear una colección `matriculas` para registrar las materias en las que cada estudiante se matricula:

```
db.matriculas.insertMany([
  { "estudiante": "Juan Pérez", "materia": "Programación", "semestre": 1 },
  { "estudiante": "Ana Gómez", "materia": "Matemáticas", "semestre": 2 }
])
```

5. Realizar Operaciones CRUD

- **Read (Leer):** Leer todos los estudiantes:

```
db.estudiantes.find()
```

Leer un estudiante específico:

```
db.estudiantes.find({ "nombre": "Juan Pérez" })
```

- **Update (Actualizar):** Actualizar la edad de `Juan Pérez`:

```
db.estudiantes.updateOne(
  { "nombre": "Juan Pérez" },
  { $set: { "edad": 21 } }
)
```

- **Delete (Eliminar):** Eliminar un estudiante:

```
db.estudiantes.deleteOne({ "nombre": "Carlos López" })
```

- **Insertar Notas:** Crear una colección de **notas** y agregar notas para los estudiantes:

```
db.notas.insertMany([
  { "estudiante": "Juan Pérez", "materia": "Programación", "nota": 4.5 },
  { "estudiante": "Ana Gómez", "materia": "Matemáticas", "nota": 3.8 }
])
```

6. Consultas Combinadas

Puedes hacer consultas para combinar datos de diferentes colecciones usando **lookup**. Ejemplo: Ver las materias matriculadas por un estudiante:

```
db.matriculas.aggregate([
  {
    $lookup: {
      from: "materias",
      localField: "materia",
      foreignField: "nombre",
      as: "detalles_materia"
    }
  }
])
```

Resumen

Este manual cubre los conceptos básicos de:

1. Crear una base de datos (**use**).
2. Crear colecciones (**createCollection**).
3. Insertar documentos (**insertOne**, **insertMany**).
4. Operaciones CRUD (**find**, **updateOne**, **deleteOne**).
5. Ejemplo práctico con **estudiantes**, **materias**, **matriculas** y **notas**.

Espero que esta guía te sea útil para empezar a trabajar con MongoDB utilizando el ejemplo de estudiantes y sus datos relacionados.