EDUCACIÓN IT

Bootcamp de Full Stack

Bienvenidos a la clase N°58

- Fundamentos de BD
- Instalación de MongoDB - Atlas
- Instalación de MongoDB Shell
- Operadores de MongoDB:
  - Consulta
  - Inserción
  - Actualización
  - Eliminación
- Índices

IT

BACKEND

EDUCACIÓN IT

Backend

Fundamentos de BD

¿Qué es una Base de Datos?

Una base de datos es un conjunto organizado de datos que se almacenan de manera estructurada y que pueden ser fácilmente accedidos, gestionados y actualizados.

¿Qué es un Motor de Bases de Datos?

Un motor de bases de datos es un componente de software diseñado para gestionar y manipular grandes cantidades de datos de manera eficiente y segura. Funciona como una capa intermedia entre las aplicaciones y los datos almacenados, facilitando la creación, almacenamiento, acceso y manipulación de la información de una base de datos.

- Relacionales (SQL): Estos motores almacenan datos en tablas con filas y columnas, y utilizan el lenguaje SQL (Structured Query Language) para consultar y manipular los datos. Por ejemplo: MySQL, MS SQL Server, etc.
- No relacionales (NoSQL): Estos motores están diseñados para manejar grandes volúmenes de datos no estructurados o semiestructurados y se almacenan en documentos. Por ejemplo: MongoDB, Cassandra, etc.

EDUCACIÓN IT

Backend

Fundamentos de BD

¿Qué es un MongoDB?

MongoDB es un sistema de gestión de bases de datos (DBMS) NoSQL, que se clasifica como una base de datos orientada a documentos. Fue desarrollado por MongoDB Inc. y está diseñado para ser escalable y flexible, permitiendo el manejo de grandes volúmenes de datos no estructurados.

Base de Datos Orientada a Documentos

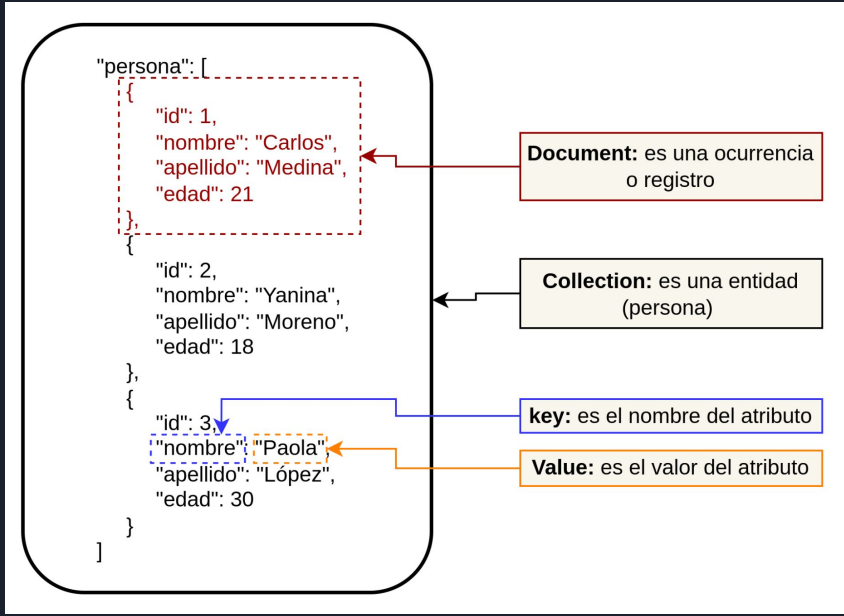
En lugar de almacenar datos en tablas como sucede en las bases de datos relacionales, MongoDB utiliza un formato de almacenamiento flexible llamado BSON (Binary JSON) que organiza la información en documentos JSON (JavaScript Object Notation).

EDUCACIÓN IT

Backend

Fundamentos de BD

Collection & Document MongoDB (NoSQL)



MongoDB es una de las tecnologías más solicitadas en Stacks orientados al lenguaje JavaScript.

EDUCACIÓN IT

Backend

Instalación de MongoDB - Atlas

¿Qué es un Cloud Cluster?

Un clúster en la nube (cloud cluster) se refiere a un conjunto de recursos informáticos interconectados que trabajan juntos para ejecutar aplicaciones o procesos específicos en un entorno de computación en la nube.

¿Qué es un Cloud Cluster?

MongoDB Atlas es un servicio de base de datos en la nube proporcionado por MongoDB Inc. El mismo, simplifica la implementación, administración y escalado de bases de datos de MongoDB en un entorno de la nube.
MongoDB Atlas cuenta con una interfaz de usuario web intuitiva que facilita la gestión y supervisión de tus clústeres de bases de datos.

Consigna:

- Crear una base de datos en https://account.mongodb.com/account/register
- Utilizar la Guía paso a paso de configuración de Atlas - MongoDB
- Crear una base de datos y una colección

EDUCACIÓN IT

Backend

Instalación de MongoDB Shell

¿Qué es MongoDB Shell?

MongoDB Shell es una interfaz de línea de comandos (CLI) que permite interactuar con una base de datos MongoDB. Se utiliza comúnmente para administrar las bases de datos y realizar tareas de desarrollo y depuración. Proporciona una forma rápida y eficiente para su interacción.

Consigna:

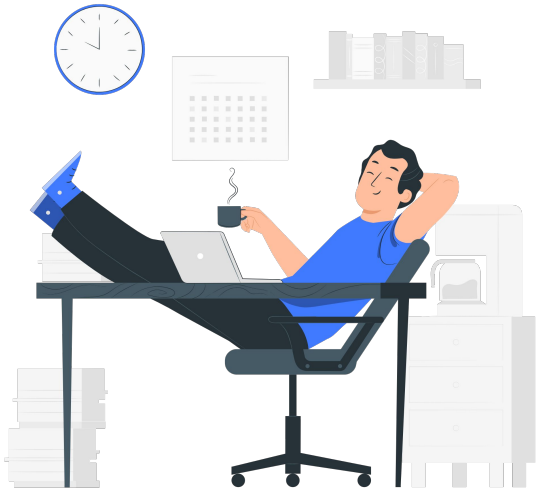
- Instalar MongoDB Shell desde https://www.mongodb.com/try/download/shell
- Comprobar su correcta instalación por medio de la terminal: mongosh --version
- Obtener cadena de conexión:

  mongodb+srv://cluster0.jcxgtwi.mongodb.net/prueba" --apiVersion 1 --username sergioBD2024 --password 4c2cY2zUguTaDcY5
- Operadores básicos:
  - show dbs
  - use <nameOfDatabase>
  - show collections
  - db.createCollection("personas")
  - db.personas.drop()
  - Exit

EDUCACIÓN IT

BREAK

Descansemos 15 minutos



EDUCACIÓN IT

Backend

Operadores de MongoDB

Documentación de MongoDB: https://www.mongodb.com/docs/manual/reference/operator/query-logical/

Operadores de consulta:

- Obtener un coche que coincida con la marca Ford:
  ```
  db.coches.findOne({ marca: "Ford" })
  ```
- Obtener coches que coincidan con la marca Fiat (por defecto, retorna 20 documentos):
  ```
  db.coches.find({ marca: "Fiat" })
  ```
- Obtener coches ordenados por marca:
  ```
  db.coches.find().sort("marca")
  ```
- Obtener el segundo documento que coincida con la marca Fiat:
  ```
  db.coches.find({ marca: "Fiat" }).skip(1).limit(1)
  ```

Operadores lógicos en consultas:

- Obtener coches que coincidan con la marca Fiat y año 2009 (and):
  ```
  db.coches.find({ $and: [ { marca: "Fiat" }, { anio: 2009 } ] })
  ```
- Obtener coches que coincidan con la marca Ford o el año 2019 (or):
  ```
  db.coches.find({ $or: [ { marca: "Ford" }, { anio: 2019 } ] })
  ```

EDUCACIÓN IT

Backend

Operadores de MongoDB

Documentación de MongoDB: https://www.mongodb.com/docs/manual/reference/operator/query-comparison/

Operadores de comparación en consultas:

- Obtener coches que no coincidan con el año 2019 (not equal):
  `db.coches.find( { anio: { $ne: 2019 } })`
- Obtener coches que sean mayor al año 2019 (greater than):
  `db.coches.find( { anio: { $gt: 2019 } })`
- Obtener coches que sean mayor o igual al año 2019 (greater than equal):
  `db.coches.find( { anio: { $gte: 2019 } })`
- Obtener coches que sean mayor al año 2019 (less than):
  `db.coches.find( { anio: { $lt: 2013 } })`
- Obtener coches que sean mayor o igual al año 2019 (less than equal):
  `db.coches.find( { anio: { $lte: 2013 } })`
- Obtener coches que coincidan con el año 2009, 2013 y 2024 (in):
  `db.coches.find( { anio: { $in: [ 2009, 2013, 2024 ] } })`
- Obtener coches que No coincidan con el año 2009, 2013 y 2024 (not in):
  `db.coches.find( { anio: { $nin: [ 2009, 2013, 2024 ] } })`

EDUCACIÓN IT

Backend

Operadores de MongoDB

Documentación de MongoDB: https://www.mongodb.com/docs/manual/tutorial/insert-documents/

Operadores de inserción:

- Insertar un coche:

```
db.coches.insertOne({  id: 9,  marca: "Fiat", modelo: "Toro", anio: 2024 })
```

- Insertar coches:

```
db.coches.insertMany([
 { id: 10,  marca: "Fiat", modelo: "Toro", anio: 2023 },
 { id: 11, marca: "Fiat", modelo: "Toro", anio: 2022 }
])
```

EDUCACIÓN IT

Backend

Operadores de MongoDB

Documentación de MongoDB: https://www.mongodb.com/docs/manual/tutorial/update-documents/

Operadores de modificación:

- Modificar el coche que tiene el id 9:

```
db.coches.updateOne({ id: 9 }, { $set: { modelo: "Qubo", anio: 2024 } })
```

- Modificar todos los coches en donde el año sea mayor o igual al 2020:

```
db.coches.updateMany({ anio: { $gte: 2020 } }, { $set: { marca: "Fiat", modelo: "Qubo", anio: 2020 } })
```

EDUCACIÓN IT

Backend

Operadores de MongoDB

Documentación de MongoDB: https://www.mongodb.com/docs/manual/tutorial/remove-documents/

Operadores de eliminación:

- Eliminar el coche con id 9:
  ```
  db.coches.deleteOne({ id: 9 })
  ```
- Eliminar todos los coches en donde el año sea menor al 2020:
  ```
  db.coches.deleteMany({ anio: { $lt: 2020 } })
  ```

Operador de contador de documentos:

- Contar documentos de la colección coches:
  ```
  db.coches.countDocuments()
  ```
- Contar documentos que coincidan con la marca Fiat de la colección coches:
  ```
  db.coches.find({ marca: "Fiat" }).count()
  ```

EDUCACIÓN IT

Backend

Índices

Documentación de MongoDB: https://www.mongodb.com/docs/manual/core/indexes/index-properties/

Operadores de creación de índices:

- Crear un índice ascendente para el campo marca de coches:
  ```
  db.coches.createIndex({ "marca": 1 }, { name: "idx_marca" })
  ```
- Crear un índice descendente para el campo modelo de coches:
  ```
  db.coches.createIndex({ "modelo": -1 }, { name: "idx_modelo" })
  ```
- Crear un índice ascendente de **tipo único** para el campo **id** de coches:
  ```
  db.coches.createIndex({ "id": 1 }, { name: "idx_id", unique: true })
  ```

Operadores de eliminación de índices:

- Eliminar el índice **idx_modelo** del campo modelo de coches:
  ```
  db.coches.dropIndex("idx_modelo")
  ```

Operador de uso de índices:

- Emplear el índice **idx_marca** para buscar coches por marca:
  ```
  db.coches.find({ "marca": 1 }).hint("idx_marca")
  ```

EDUCACIÓN IT

CIERRE DE CLASE

Continuaremos en la próxima clase