

ACÀMICA

Agenda

Introducción

Daily

Programo: Mongo DB - CRUD

Buenas prácticas

Break

Programamos: Mongo DB

Programan: Endpoints con Mongo DB

Cierre



MongoDB

Hoy nos introducimos en el mundo de las bases de datos no relacionales a través de MongoDB.

Daily



Daily



Sincronizando...

Bitácora



¿Cómo te ha ido?
¿Obstáculos?
¿Cómo seguimos?

Challenge



¿Cómo te ha ido?
¿Obstáculos?
¿Cómo seguimos?

Bases de datos



Relacional (SQL)

VS

No Relacional (No SQL)

Se basan en la organización de la información en trozos pequeños, que se relacionan entre ellos mediante la relación de identificadores.



ORACLE®



Las bases de datos no relacionales tienen un esquema dinámico, no se requiere la estructura de los datos para su manipulación.



MongoDB



MongoDB

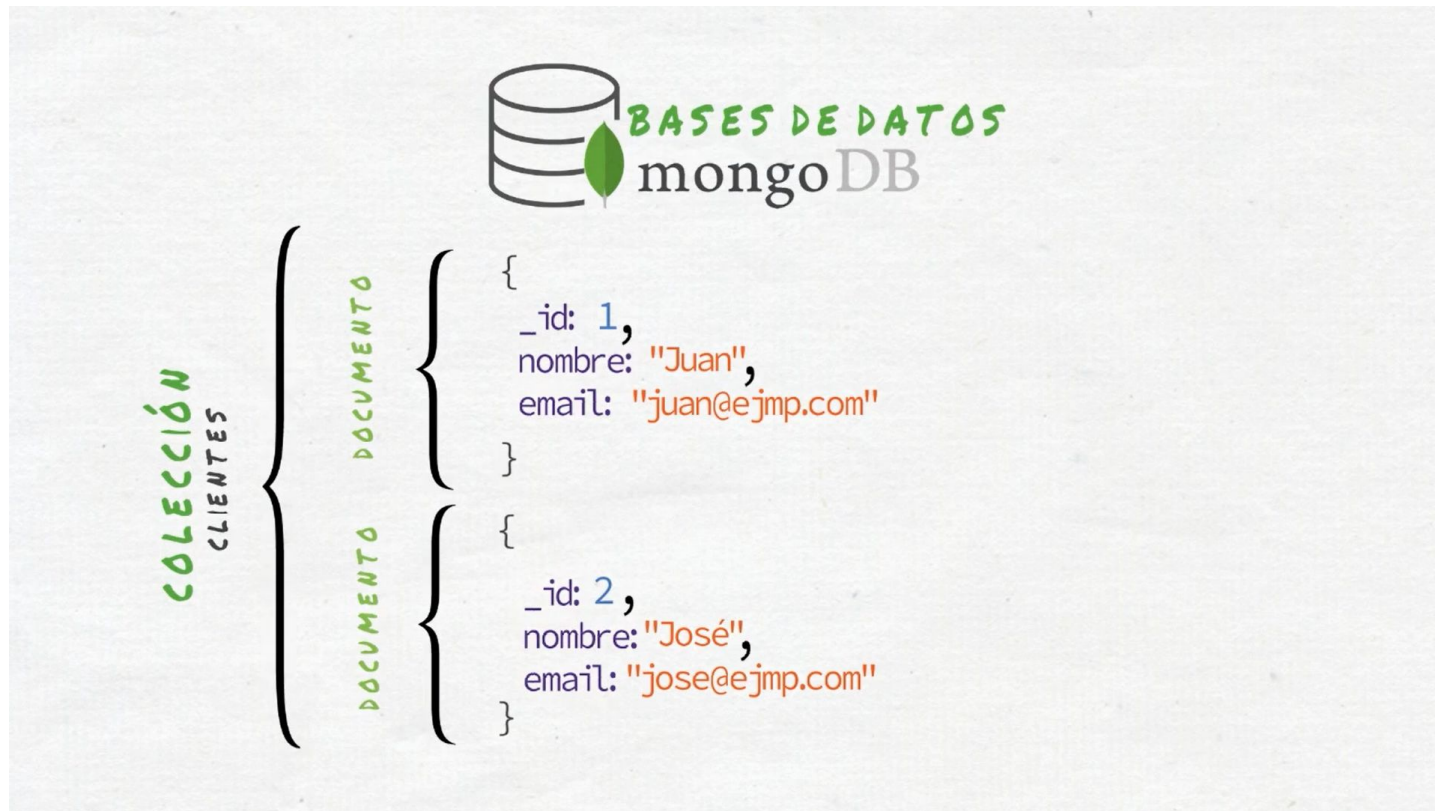
es un sistema para la gestión de bases de datos no relacional. Utiliza un modelo de documentos basado en JSON.

¿Cómo se compone una base de datos MongoDB?

MongoDB utiliza **colecciones** para agrupar información y dentro de ellas almacena **documentos**.



Composición



Mongo

Entre sus principales ventajas se encuentran

- Escalabilidad
- Flexibilidad
- Bajo consumo de recursos



Programo

mentores/as



MongoDB

Veamos en vivo cómo trabajar con **MongoDB**



Buenas prácticas



Credenciales

Almacena los datos de conexión en variables de entorno.

Los accesos variarán si estás utilizando una conexión local o una conexión a producción.

No olvides agregar tus credenciales de archivos al `.gitignore` para que no queden en el repositorio.

A close-up photograph of a white ceramic cup filled with a latte. The surface of the milk is decorated with intricate latte art, featuring a central heart shape surrounded by concentric, wavy lines. The cup is placed on a matching white saucer. In the background, a white napkin and a silver fork are visible, though they are out of focus. The overall lighting is soft and even, highlighting the textures of the coffee and the smooth surface of the cup.

¡BREAK!



Programamos

todos/as



Programamos

Programemos la siguiente estructura:

1. Crear un script que inserte en Mongo el menú de un restaurante

`{plato: 'valor', precio: 'valor', tipo_de_plato: 'valor'}`

2. Crear un segundo script que retorne todos los platos del restaurante.

¡Ambos script deben compartir
el mismo archivo de conexión a la base de datos!

Programan

estudiantes



Restaurant

1. Añade express a tu proyecto de **Programamos** y genera un Endpoint para cada una de las operaciones.



Actividad

Restaurant

1. Añade express a tu proyecto de **Programamos** y genera un Endpoint para cada una de las operaciones.
2. Añade un Endpoint que actualice un plato en tu base de datos.



Restaurant

1. Añade express a tu proyecto de **Programamos** y genera un Endpoint para cada una de las operaciones.
2. Añade un Endpoint que actualice un plato en tu base de datos.
3. Añade un Endpoint para eliminar un plato.



Restaurant

1. Añade express a tu proyecto de **Programamos** y genera un Endpoint para cada una de las operaciones.
2. Añade un Endpoint que actualice un plato en tu base de datos.
3. Añade un Endpoint para eliminar un plato.
4. Genera un nuevo Endpoint para buscar un plato en específico.



Restaurant

1. Crea un middleware para la siguiente operación:
“verificar que un plato exista”.
2. Cuando lo tengas funcionando, aplica este middleware a las operaciones de selección, actualización y borrado.



Para la próxima

- 1) Termina el ejercicio del encuentro de hoy.
- 2) Lee la bitácora 47 y carga las dudas que tengas al Trello.
- 3) Resuelve el challenge.

En el encuentro que viene uno/a de ustedes será seleccionado para mostrar el ejercicio de hoy y otro/a mostrará cómo resolvió el challenge de la bitácora. De esta manera, ¡aprendemos todos/as de (y con) todos/as, así que vengan preparados/as!

ACÀMICA