**M3S19: MongoDB: Documentos, Queries, Atlas**

**¡Hola, espero te encuentres muy bien!, nos encontramos en la semana 19 de nuestro Bootcamp.**

**En esta semana damos inicio a un nuevo tema y es el que tiene que ver con MongoDB, es muy importante que presten atención para que puedas continuar sin inconvenientes.**

**A continuación se muestra un pequeño índice con el contenido que estaremos abordando esta semana. Iniciemos… 😃**

**ÍNDICE**

* MongoDB
  + [Instalación](https://github.com/U-Camp/BOOT-M3-SEM19#instalaci%C3%B3n)
  + [Documentos](https://github.com/U-Camp/BOOT-M3-SEM19#documentos)
  + [Filtro de documentos con "queries"](https://github.com/U-Camp/BOOT-M3-SEM19#filtro-de-documentos-con-queries)

**MongoDB**

Según sus creadores: Es una base de datos distribuida, basada en documentos y de uso general que ha sido diseñada para desarrolladores de aplicaciones modernas y para la era de la nube. Es una base de datos documental, lo que significa que almacena datos en forma de documentos tipo JSON. Se cree que esta es la forma más natural de concebir los datos; frente al tradicional modelo de filas y columnas, esta es mucho más expresiva y potente. Si deseas conocer sus características y profundizar un poco más sobre MongoDB puedes ingresar a esta URL: <https://www.mongodb.com/es>

Te dejo por aquí unas infografías que resumen un poco los aspectos más importantes de MongoDB puedes consultarlas y descargarlas.

[Infografía 1](https://github.com/U-Camp/BOOT-M3-SEM19/blob/main/infografias/MongoDB1.pdf); [Infografía 2](https://github.com/U-Camp/BOOT-M3-SEM19/blob/main/infografias/MongoDB2.pdf); [Infografía 3](https://github.com/U-Camp/BOOT-M3-SEM19/blob/main/infografias/MongoDB3.pdf); [Infografía 4](https://github.com/U-Camp/BOOT-M3-SEM19/blob/main/infografias/MongoDB4.pdf);

Ahora bien, si ya tienes más claro lo que es MongoDB, iniciemos con la instalación.

**Instalación**

Necesitaremos instalar dos programas.

* **MongoDB Community Edition.** Nuestra base de datos que nos permitirá almacenar y persistir nuestros datos a lo largo del tiempo. Podrá ser modificado a través del Backend.

<https://docs.mongodb.com/manual/administration/install-community/>

* **MongoDB Compass.** Es un programa que nos permitirá gestionar nuestra base de datos desde una interfaz mucho más amable. <https://www.mongodb.com/es/products/compass>

**Documentos**

Dentro de una base de datos, encontraremos una estructura la cual estará dividida en 2 secciones:

* **Colecciones.** Se refieren a la categoría de nuestro dato. Por ejemplo CARROS, MASCOTAS, CELULARES.
* **Documentos.** Se refiere a una unidad de dato real. Puede ser un iPhone rojo con 32GB o Perro llamado Lassie. Estos se estructuran a través de propiedades y valores, muy similar a los objetos en JavaScript.

Una vez instalados tus dos programas, abre MongoDB Compass y vamos conectarnos, dando clic en "CONNECT".

Te deberá salir una pantalla de acceso a las bases de datos. Darás clic izquierdo a crear base de datos y la llamarás ucamp. Tu primer colección la llamaremos estudiantes.

Deberás llegar a una pantalla similar a esta:

Generemos ahora un documento para profundizar en cómo está establecida.

Demos clic a "ADD DATA", luego INSERT DOCUMENTO y posteriormente, vamos a agregar los siguientes valores para crear nuestro primer documento.

| **Propiedad** | **Valor** | **Tipo de dato** |
| --- | --- | --- |
| id (*Creado automáticamente*) | "Sandra" | String |
| name | "Sandra" | String |
| country | "Colombia" | String |
| previousExperience | false | Boolean |
| gradePointAverage | 96 | Int32 |

Si todo sucedió correctamente, agregamos nuestro primer dato en MongoDB.

Ahora bien, vamos a agregar dos estudiantes más. Utiliza la estructura anterior para guiarte.

name: "Raúl"

country: "México"

previousExperience: false

gradePointAverage: 94

name: "Lorena"

country: "Perú"

previousExperience: true

gradePointAverage: 98

Veamos el resultado:

Con esto, se determina que contamos con:

* Una base de datos llamada ucamp
* Una colección (categoría) llamada estudiantes
* Tres documentos con estudiantes reales, que incluyen las propiedades:
  + **\_id.** Se refiere al identificador del documento. Es único.
  + **name.** El nombre del estudiante.
  + **country.** El país del estudiante
  + **previousExperience.** Un booleano que determina si tiene experiencia previa.
  + **gradePointAverage.** Un tipo de dato Number, bajo el esquema de Int32, que me dice el promedio de calificación del estudiante.

**Filtro de documentos con "queries"**

Para poder filtrar documentos, necesitamos utilizar la barra de búsqueda que se encuentra en la sección posterior de MongoDB Compass.

Nos centraremos en cómo filtrar por nombre y por ID.

* Para filtrar por nombre, escribirás:

{ name: "Lorena" }

El resultado será que filtrará por nombre los documentos que contengan la propiedad "Lorena".

* Para filtrar por ID, copia el valor de un \_id de cualquier documento, escribirás en la barra de búsqueda este comando y reemplazaras IDNUMBER por tu ObjectId

{ \_id: ObjectId(IDNUMBER) }

Obtendrás un resultado como este:

Esa es la idea de los queries. Dentro de la gran cantidad de datos que puedas llegar a crear, necesitarás en algún punto filtrar tus documentos por criterios. Te recomendamos explorar la documentación para que conozcas todas las posibilidades de búsqueda:

<https://docs.mongodb.com/compass/current/query/filter/>

**¡Muy bien!, has llegado al final del contenido de esta semana, espero hayas aprendido. Recuerda que los coaches estan disponibles para aclarar tus dudas.**

**¡Nos vemos el sábado! 😃**