

Instrucciones aprendidas MongoDB

Vamos a ver cómo realizar instrucciones de CRUD básicas en MongoDB además de otras instrucciones que nos serán de gran ayuda y utilidad.

Lo primero que vamos hacer es comprobar si hay alguna base de datos creada en nuestro MongoDB con el siguiente comando:

show dbs

Éste comando mostrará una lista de las bases de datos que hay creadas en MongoDB. Para crear una base de datos nueva, ejecutaremos el siguiente comando:

use <nombre_bd>

Este comando use nos servirá tanto para crear bases de datos, como para seleccionarlas. Si la base de datos que vallamos a seleccionar no está, se crea una nueva.

Dentro de la base de datos, la información esta segmentada en colecciones. Para introducir datos a ésa colección:

db.<coleccion>.insertOne({campo:'valor', campo:'valor'})

Con ésto, crearemos una colección llamada contacto, con dos campos y sus valores.

Para mostrar los datos de esa colección:

db.<coleccion>.find()

Seleccionamos la colección y con el método find() lo que hacemos es visualizar todo el contenido que tiene ésa colección.

Para visualizarlo algo mas ordenado podemos utilizar “.pretty()”.

db.<coleccion>.find().pretty()

El siguiente comando visualizará las colecciones que tenemos creadas en nuestra base de datos:

show collections

Ahora vamos a introducir varios documentos más datos a nuestra colección:

db.<coleccion>.insertMany()

Para hacer comparaciones con números, si es mayor, menor o igual se utilizan unos selectores especiales que son los siguientes:

\$lt : Para menor que...
db.<coleccion>.find({<campo>:{\$lt: <valor>}})

\$lte : Para menor o igual que...
db.<coleccion>.find({<campo>:{\$lte: <valor>}})

\$gt : Para mayor que...
db.<coleccion>.find({<campo>:{\$gt: <valor>}})

\$gte : Para mayor o igual que...
db.<coleccion>.find({<campo>:{\$gte: <valor>}})

\$ne : Que no sea igual a...
db.<coleccion>.find({<campo>:{\$ne: <valor>}})

\$eq: Que sea igual a...
db.<coleccion>.find({<campo>:{\$eq: <valor>}})

También podemos comprobar si existe un campo o no en los documentos que tenemos almacenados en nuestra base de datos con el operador \$exists :

db.<coleccion>.find({<campo>:{\$exists: <true>}})
db.<coleccion>.find({<campo>:{\$exists: <false>}})

También podemos incluir varias ordenes en una misma consulta con los operadores \$and y \$or.

db.<coleccion>.find({\$or : [{<campo>:{\$lte: <valor>}},{<campo>:<valor>}]})
db.<coleccion>.find({\$and : [{<campo>:{\$lte: <valor>}},{<campo>:<valor>}]})

Al igual que podemos crear documentos en nuestra colección tambien podemos borrarlo

db.<coleccion>.deleteOne({campo:'valor', campo:'valor'})

o borrar mas de uno, incluso todos

db.<coleccion>.deleteMany()

También podemos ver si pertenece o no con la instrucción \$in y la instrucción \$nin.

db.<coleccion>.find({<campo>:{\$in:[<valor>,<valor>,<valor>]}})
db.<coleccion>.find({<campo>:{\$nin:[<valor>,<valor>,<valor>]}})

Otra instrucción que hemos visto es la instrucción \$ne que nos ayuda a buscar documentos donde el valor del campo no es igual al valor especificado.

db.<coleccion>.find({<campo>:{\$ne:[<valor>]}})

La instrucción \$all nos permite seleccionar los documentos donde el valor de un campo se compara con una matriz de valores. Actua como una \$and.

```
db.<coleccion>.find({<campo>:{$all:[<valor>,<valor>,<valor>]}})  
db.<coleccion>.find({$and:[{<campo>:<valor>},{<campo>:<valor>}]})
```

Por ultimo la instrucción \$regex nos ayuda a comparar un patron con el valor de un campo para comprobar si existe o no coincidencia.

```
db.<coleccion>.find({<campo>:{$regex: /<patron>/i}})
```

-La opcion “i” nos da insensibilidad a mayúsculas y minúsculas.