

MongoDB Crash Course - WEB DEV SIMPLIFIED

1. 1:40 -- Comandos

test> show dbs

Crear una db nueva --> use appdb

Mostrar Collections --> show collections

Borrar db en la que estoy --> db.dropDatabase()

clear --> cls

Insertar un registro en una collection(**users**) si la collection no existe te la crea automáticamente al insertar --> db.users.insertOne({ x: 1 })

2. 4:59 Find

db.users.find() Va a buscar todo lo que esté en esa collection.

ALGO MUY POSITIVO DE MONGO es que genera las id por ti, así que es algo menos de que preocuparse.

cada objeto que abras es un document:

e.g:

db.users.insertOne({ x: 1 })

db -> collections -> documents

Inserting Nested Object:

db.users.insertOne({name: "Ricardo" , age:36, direccion:{calle:"vargas con bermudez"}, hobbies:["Running"]})

We can insert many objects at the same time:

db.users.insertMany([{name:"Fernando"},{name:"David"}])

Quedé en minuto 9 [Queries](#)

Puedes probar con estos datos de abajo que ya introduciste...

db.users.insertMany([{name: "Jupiter" , age:36, direccion:{calle:" bermudez"}, hobbies: ["Running"]}, {name: "Sally" , age:26, direccion:{calle:"pilar"}, hobbies:["Jogging"]}, {name: "Mike" , age:16, direccion:{calle:"Bolivar"}, hobbies:["Swiming"]}])

Supongamos que quiero que me devuelva a Mike:

* db.users.find({name:'Mike'})

La edad de Mike sin el campo id:

```
db.users.find({name:'Mike'}, {age:1, _id:0})
```

More Complex Queries

1. `db.users.find({name:{ $eq:"Sally" }})` -> Iguales a Sally

2. `db.users.find({name:{ $ne:"Sally" }})` -> Los que sean diferentes a Sally

Quedé en minuto 13:55 mas queries complejos...

Greater \$eg , Ej: Edades mayores a 30:

```
db.users.find({age:{ $gt:30}})
```

Mayores ó **iguales** a 26

```
db.users.find({age:{ $gte:30}})
```

Menores o iguales a 26:

```
db.users.find({age:{ $lte:26}})
```

Menores a 20:

```
db.users.find({age:{ $lt:20}})
```

Iguales a : Sally ó Mike

```
db.users.find({name:{ $in:["Sally","Mike"]}})
```

Que no sean Sally ó Mike

```
db.users.find({name:{ $nin:["Sally","Mike"]}})
```

Solo los objetos que tengan edad:

```
db.users.find({age:{ $exists:true}})
```

Edades entre 20 y 39 años:

```
db.users.find({age:{ $gte:20, $lte:39}})
```

Quedé minuto 17

```
appdb> db.users.find({age:{$gte:10, $lte:19}, name: "Mike"})
[
  {
    _id: ObjectId("6167955aebbe78794d2e10a4"),
    name: 'Mike',
    age: 16,
    direccion: { calle: 'Bolivar' },
    hobbies: [ 'Swiming' ]
  }
]
```

* Combinar de forma mas directa..

```
appdb> db.users.find({$and:[{age: 16},{name:"Mike"}]})
```

* OR

```
appdb> db.users.find({$or:[{age: {$lte:30}},{name:"Mike"}]})
[
  {
    _id: ObjectId("616794c2ebbe78794d2e10a3"),
    name: 'Sally',
    age: 26,
    direccion: { calle: 'pilar' },
    hobbies: [ 'Jogging' ]
  },
  {
    _id: ObjectId("6167955aebbe78794d2e10a4"),
    name: 'Mike',
    age: 16,
    direccion: { calle: 'Bolivar' },
    hobbies: [ 'Swiming' ]
  }
]
```

*Que no sean menores a 26

```
db.users.find({age:{$not : { $lte:26}}})
```

Quedé en minuto 20

```
appdb> db.users.find({"direccion.calle": "Bolivar"}) es lo mismo que ...
```

```
appdb> db.users.find({"direccion": {"calle":"Bolivar"}})
```

*Cuantos hay que sean mayores de 20

```
appdb> db.users.countDocuments({"age":{"$gte:20}})
```

29:09 Update Data

Update One

```
appdb> db.users.updateOne({age:16},{ $set:{age:18}})
```

```
db.users.updateOne({_id:ObjectId("6167955aebbe78794d2e10a4")},{ $set:{age:17}})
```

Incrementar un Valor:

```
appdb> db.users.updateOne({_id:ObjectId("6167955aebbe78794d2e10a4")},{ $inc:{age:2}})
```

CAMBIAR NOMBRE DE UNA COLUMNA ó «FIELD»

```
appdb> db.users.updateOne({_id:ObjectId("6167955aebbe78794d2e10a4")},{ $rename:
{age:"edad"}})
```

ELIMINAR COMPLETAMENTE UN «FIELD»

```
appdb> db.users.updateOne({_id:ObjectId("6167955aebbe78794d2e10a4")},{ $unset:
{edad:""}})
```

AGREGAR ELEMENTO A UN ARRAY

```
appdb> db.users.updateOne({_id:ObjectId("6167955aebbe78794d2e10a4")},{ $push:
{hobbies:"Correr"}})
```

Output

```
{
  _id: ObjectId("6167955aebbe78794d2e10a4"),
  name: 'Mike',
  direccion: { calle: 'Bolivar' },
  hobbies: [ 'Swiming', 'Correr' ]
}
```

SACAR UN ELEMENTO

```
appdb> db.users.updateOne({_id:ObjectId("6167955aebbe78794d2e10a4")},{ $pull:
{hobbies:"Correr"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

QUITAR DIRECCION A TODOS LOS QUE TENGAN DIRECCION

```
appdb> db.users.updateMany({direccion:{$exists:true}},{$unset:{direccion:''}})
```

REPLACE ES MUY RARO DE USAR POR QUE TE REEMPLAZA TODOS LOS CAMPOS POR UNO SOLO

DELETE

La sintaxis es igual a la de find

Borrar todos los que sean mayores a 25

```
appdb> db.users.deleteMany({age:{$gte:25}})
```

También está `deleteOne`