

## 5.6

Elimina una colección

```
db.países.drop()
```

Elimina la base de datos

```
db.dropDatabase()
```

## 5.9

Evita o salta la obtención de los primeros “n” documentos.

Si en la colección hay 105 documentos, este comando devolvera solo el último, formateado por pretty()

```
db.países.find().skip(104).pretty()
```

```
> db.países.find().skip(104).pretty()
{
  "_id" : ObjectId("5e8cf6551cc5c667b160b55a"),
  "codigoPaís" : 258,
  "nombrePaís" : "Mozambique",
  "capitalPaís" : "Maputo",
  "region" : "Africa",
  "poblacion" : 26423700,
  "latitud" : -18.25,
  "longitud" : 35,
  "date" : ISODate("2020-04-07T21:53:25.517Z"),
  "_class" : "com.example.demo.model.País"
}
```

## 5.10

```
db.países.find({"nombrePaís":/Ar/}).pretty()
```

o similar:

```
db.países.find({"nombrePaís":/.*Ar.*}).pretty()
```

Están buscando algo que contenga "Ar" en algún nombre de país (el operador % de SQL es equivalente a '.\*' de Regexp ).

nota: mongodb usa expresiones regulares que son más poderosas que "LIKE" en sql. Con expresiones regulares se pueden crear muchos patrones.

Más info:

enlace [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular\\_Expressions](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions)

```

db.países.find({"nombrePais":/Ar/}).pretty()
{
  "_id" : ObjectId("5e8cf28dd7b1b55d8340be19"),
  "codigoPais" : 54,
  "nombrePais" : "Argentina",
  "capitalPais" : "Buenos Aires",
  "region" : "Americas",
  "poblacion" : 43590400,
  "latitud" : -34,
  "longitud" : -64,
  "date" : ISODate("2020-04-07T21:37:17.203Z"),
  "_class" : "com.example.demo.model.Pais"
}

{
  "_id" : ObjectId("5e8cf30fd7b1b55d8340be68"),
  "codigoPais" : 297,
  "nombrePais" : "Aruba",
  "capitalPais" : "Oranjestad",
  "region" : "Americas",
  "poblacion" : 107394,
  "latitud" : 12.5,
  "longitud" : -69.96666666,
  "date" : ISODate("2020-04-07T21:39:27.976Z"),
  "_class" : "com.example.demo.model.Pais"
}

```

### 5.11

De manera predeterminada, la creación de un índice es una operación de bloqueo que bloquea la base de datos hasta que se complete. Cuando ya tiene muchos datos en su colección, esto puede llevar mucho tiempo. Pero puede crear un índice en segundo plano utilizando la `background:true` opción

```
db.collection.createIndex({ keyfield: 1, otherkeyfield: 1 }, { background: true });
```

En general, llevará más tiempo, pero la base de datos seguirá respondiendo mientras lo hace.

### 5.12

Antes que nada si estamos usando windows hay que agregar al path la dirección de la carpeta bin de mongoDB

Lo primero que tenemos que hacer es iniciar el servidor de MongoDB a través del comando:

`mongod`

Para luego poder ejecutar el comando, que inicia un shell de MongoDB, con el comando:

`mongo`

Una vez inicializado el shell de Mongo, ya podemos crear los backups de todas las bases de datos que tenemos en MongoDB, a través del comando:

`mongodump`

El cual, por defecto crea una carpeta llamada **dump**, que contiene todas nuestras bases de datos separadas por carpetas.

Ahora vamos a hacer una prueba, en la misma sesión del shell de mongo; es decir en la misma consola que se ejecutó el comando **mongo** vamos a eliminar las bases de datos que tengamos hasta el momento, para poder restaurarlas a partir de los backups que hemos generado previamente.

Para eliminar bases de datos de MongoDB, primero tienes que seleccionarlas a través del comando: `use países_db`; en donde es el nombre de **países\_db** es la base de datos que quieres eliminar. lo que te permitirá ejecutar el comando `db.dropDatabase()` que se encargará de eliminar la base de datos que seleccionaste.

Una vez la base de datos este eliminada, vamos a usar el backup para restaurarla, y para hacer eso, desde tu consola o terminal necesitas ubicarte en la ruta en donde este almacenado el backup, es decir en donde este la carpeta **dump** y ejecutar el comando **mongorestore**. este comando lo hará es leer la carpeta dump y restaurar todas las bases de datos que encuentre ella.

Pero mucha veces tan solo vas a necesitar hacer un backup de una sola base de datos, y para eso tan solo necesitas agregar una opción al comando mongodump. indicándole el nombre de la base de datos a la que quieres hacer una copia de seguridad:

```
mongodump --db paises_db
```

este comando de nuevo creara una carpeta llamada **dump**, pero esta vez tan solo con la base de datos que hemos seleccionado.

Y para restaurar esta única base de datos, ejecutamos el comando **mongorestore** pero esta vez con la opción --db y la ruta de la carpeta en donde se encuentra la base de datos que queremos restaurar:

```
mongorestore --db paises_db dump/paises_db
```

Finalmente la otra opción, es que quieras restaurar tan solo una colección, de una base de datos. para esta tarea tan solo le agregamos la opción **--collection** al comando mongodump:

```
mongodump --db paises_db --collection paises
```

Y para restaurar tan solo una colección, tan solo le agregamos la opción --collection al comando mongorestore, además de la ruta de en donde se encuentra tal colección:

```
mongorestore --db paises_db --collection paises dump/paises_db/paises
```