

## Insert

Para insertar documentos en una base de datos NoSQL en MongoDB, primero debemos abrir el archivo en cuestión donde pondremos todos los documentos. En el terminal estableceremos que base de datos vamos a usar, tan sencillo como abrir MongoDB escribiendo “**mongo**” y eligiendo la base poniendo “**use BaseDeDatos**”, poniendo en BaseDeDatos el nombre de la base que vayamos a usar. En caso de que no exista, se creará al insertarle documentos.

Tras ello, en el archivo que hayamos creado (que deberá ser **.js**), pondremos “**db.Collection.insert()**”. Con **db** especificamos que usaremos una base de datos, en **Collection** pondremos el nombre de la colección que vayamos a usar. Y por último dentro del **insert()** pondremos los documentos. Hay tres tipos de insert, **insertOne()** que inserta un solo documento, **insertMany([])** que inserta varios documentos y **insert()** que funciona para ambos. En caso de usar **insert()** cuando queremos meter más de un documento deberemos usar [] dentro del paréntesis.

Cuando son más de un documento, se denomina array.

Tras ello, cargando un **load()** en el terminal deberán añadirse nuestros archivos a la base de datos.

## Query

Para hacer búsquedas bastará con poner en el terminal (con MongoDB ejecutado) “**db.Collection.find( {} )**”. Dentro del **find()** pondremos las búsquedas que haremos con sus cláusulas, si lo dejamos vacío nos devolverá todos los resultados.

### Igual

Para poner un igual bastará con dentro del **find()** dejar el nombre del campo con su valor. Por ejemplo: **db.Collection.find( {status: “A”} )**.

### Or

Para poner un Or usamos **\$or**. Este irá delante de los campos, tal que así **db.Collection.find( { \$or: [{status: “A”}, {status: “B”}] } )**.

### And

Para poner un and es tan sencillo como poner dentro del **find()** cada uno separado por una coma

### Mayor y menor

Para poner mayor o menor se debe usar el **\$lt** para menor y el **\$gt** para mayor. Se pone dentro de búsquedas, por ejemplo “**...status:(\$gt: 20)**”.

### Count y pretty

Al final de cada búsqueda se puede añadir “**.pretty()**” o “**.count()**”. **Pretty** muestra los documentos más ordenados. Y **count** devuelve el número de documentos encontrados con la búsqueda.