PROYECTO BASE DE DATOS NOSQL

Raúl Benítez Ordóñez

IES Punta del Verde

Adolfo Salto Sánchez del Corral

15/11/2021

Propósito del proyecto

Realizar un proyecto subiéndose a nuestro repositorio GitHub con la estructura vista en clase:

Carpeta src (con los .js), carpeta doc (con un pdf que explique aspectos teóricos, técnicos y lo que pretende tu proyecto) y archivo README.md con la introducción al proyecto.

Se escribirán los métodos para insertar documentos en la colección o colecciones que utilices.

Las colecciones deberán tener campos de distintos tipos: cadenas de caracteres, números, tipo documento, tipo array, booleanos, fecha... y el suficiente juego de datos para que las consultas tengan sentido y se aprecie el filtro utilizado.

Se escribirán instrucciones con métodos find para obtener información de los documentos. Todos los métodos find deberán tener un enunciado con la redacción de lo que se pretende obtener con la consulta.

Se deberán utilizar todos los operadores vistos en las clases.

El proyecto tendrá una conexión con algún aspecto del mundo real y un nivel de complejidad en la estructura de los documentos y en las consultas superior a los ejemplos vistos en las clases.

Se valorará la aportación personal: Uso de nuevos operadores o consultas elaboradas sobre campos tipo documento, tipo array y tipo fecha..

Defenderemos el proyecto en clase y se valorará la presentación y las respuestas a cuestiones planteadas por el profesor.

Para la presentación, será de unos 10 a 15 minutos, se usará el repositorio GitHub en el que se ha subido el proyecto, se dará una explicación introductoria del proyecto mostrando el pdf, se explicará la estructura de los documentos de la colección o colecciones y se mostrarán las consultas más elaboradas que se hayan realizado. El profesor podrá pedir al alumno que instale su colección o colecciones en el equipo del profesor y ejecute alguna de sus consultas.

¿Qué pretende mi proyecto?

Mi proyecto consiste en una base de datos de una tienda de música vintage donde se venden discos de los grupos y artistas más míticos (principalmente CDs y Vinilos) y algo de merchandising de estos.

Podemos encontrar dos colecciones:

la primera colección de los discos con 81 registros, organizándose entre...

* Tipo (CD o Vinilo).
* Nombre del disco.
* Un ID con el que indexar.
* Fecha de lanzamiento.
* Género del disco.
* Artista.
* Formato (Álbum, Single o EP).
* Discográfica.
* Estado de conservación (Principalmente para organizar los Vinilos).
* Precio.
* Stock (Si hay stock o no)

y la segunda colección del merch con 35 registros, organizándose entre...

* Tipo de producto. (Camisetas, Sudaderas, Chapas, Llaveros o Tazas)

* Un ID con el que indexar e identificarlo.

* Nombre.

* Artista.

* Tamaño. (Grande, Mediano o Pequeño)

* Rebaja (Si el producto se encuentra rebajado o no).

* Precio.

Con estas colecciones, realizamos una serie de consultas o dudas de clientes con los métodos vistos en clase para resolverlos.

Este tipo de colecciones pueden venir muy bien para organizar todo el inventario de la tienda de música.

Métodos usados en el proyecto

Se utilizan todos los campos aprendidos en clase:

- Cadenas de caracteres,
- Números,
- Documento,
- Array,
- booleanos,
- fechas.

Se utilizan todos los operadores aprendidos en clase:

- $eq.
- $gt.
- $gte.
- $lt.
- $ni.
- $nin.
- $nor.
- $not.
- $and. (Implícito y explícito)
- $regex.
- $in.
- $elemMatch.
- new Date ()
- $exists.
- $all.

Métodos nuevos y explicación.

ObjectId (<hexadecimal>):

El valor de ObjectId de 12 bytes consta de:

- un valor de marca de tiempo de 4 bytes , que representa la creación del ObjectId, medido en segundos.
- un valor aleatorio de 5 bytes generado una vez por proceso. Este valor aleatorio es exclusivo de la máquina y el proceso.
- un contador incremental de 3 bytes , inicializado a un valor aleatorio.

Db.collection.update (query, update, options):

Modifica un documento o documentos existentes en una colección. El método puede modificar campos específicos de un documento o documentos existentes o reemplazar un documento existente por completo, según el parámetro de actualización .

De forma predeterminada, el db.collection.update()método actualiza un solo documento. Incluya la opción multi: true para actualizar todos los documentos que coincidan con los criterios de consulta.

$set:

El $set reemplaza el valor de un campo con el valor especificado.

La $set del operador tiene la siguiente forma:

```
{ $set: { <field1>: <value1>, ... } }
```

$unSet:

El $unset elimina un campo en particular. Sintaxis:

```
{ $unset: { <field1>: "", ... } }
```

El valor especificado en la $unset (es decir "") no afecta la operación.

$addToSet:

El $addToSet agrega un valor a una matriz a menos que el valor ya esté presente, en cuyo caso $addToSet hace nada a esa matriz.

El $addToSet tiene la forma:

```
{ $addToSet: { <field1>: <value1>, ... } }
```

FIN DEL PROYECTO