

Proyecto de Base de Datos.

En este proyecto, he realizado la inserción de una colección de datos acompañada de una serie de consultas que hacen posible la visualización de éstos. Esta colección de datos se basa en una tienda de videojuegos, la cual ofrece los productos que se indican en dicha colección.

Para llevar a cabo la elaboración de la colección de datos, se requiere el previo desarrollo de los siguientes campos:

- Tipo array.
- Tipo boolean.
- Tipo documento.
- Tipo fecha.
- Cadena de caracteres.
- Números.

Todos estos elementos, nos permiten ejecutar aquella serie de consultas necesarias para seleccionar y visualizar los datos. En el proceso de creación de estas consultas, he hecho uso de los siguientes operadores y sus respectivas variaciones:

- \$ regex.
- \$ eq. (menor que)
- \$lt. (menor que)
- \$lte. (menor o igual que)
- \$gt. (mayor que)
- \$gte. (mayor o igual que)
- \$ne. (distinto)
- \$in. (dentro de)
- \$nin. (no dentro de)
- \$and
- \$or

A continuación, en la siguiente hoja, adjunto unas capturas de pantalla de las correspondientes consultas e inserción de datos a través de insert.many:

```
File Edit Selection View Go Run Terminal Help
Consultas.js - TrabajoMongo - Visual Studio Code

EXPLORER
> OPEN EDITORS 1 UNSAVED
> TRABAJO MONGO
> doc
> src
  JS Consultas.js M
  JS InsertMany.js M
  Readme.md

Consultas.js
1 //Con esta consulta quiero que mongo me muestre las marcas las cuales empiecen por la letra b y que este no distinga entre mayusculas y minusculas
2 db.yumanyi.find( { Marca: { $regex: /^b/, $options: 'i' } } )
3 //Con esta consulta se mostraran todos los items que tengan un precio igual o superior a 400
4 db.yumanyi.find( { $and: [ { "qty": { $lte: 15 } }, { "Marca": { $in: "Nintendo" } } ] } )
5 //Con esta consulta se mostraran todos los items de marca Banpresto o a aquellos que su precio sea distinto de 400
6 db.yumanyi.find( { $or: [ { "Marca": "Banpresto" }, { "Precio": { $ne: 400 } } ] } )
7 //Con esta consulta se mostraran todos los items cuya fecha de entrada sea mayor o igual al 9/12/2020
8 db.yumanyi.find( { "FechaEntrada" : { $gt : new Date("2020-12-9") } } )
9 //Con esta consulta se mostraran todos los items cuyo precio no sea mayor de 15
10 db.yumanyi.find( { "Precio": { $not: { $gt: 15 } } } )
11 //Con esta consulta se mostraran todos los items cuya marca no empiece por la letra T y que este no distinga entre mayusculas y minusculas
12 db.yumanyi.find( { Marca: { $not: { $regex: /^T/, $options: 'i' } } } )
13 //Con esta consulta se mostraran todos los items cuya marca no acabe en s y que este no distinga entre mayusculas y minusculas y o aquellos cuya fecha de entrada no es igual a la indicada
14 db.yumanyi.find( { $or: [ { "Marca": { $not: { $regex: /^s/, $options: 'i' } } }, { "FechaEntrada": { $ne: new Date("2020-12-15") } } ] } )

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: mongo

> db.yumanyi.find( { Marca: { $not: { $regex: /^T/, $options: 'i' } } } )
{ "_id" : "Firin", "item" : "FigurasAnimadas", "qty" : 15, "Marca" : "Banpresto", "Precio" : 24, "FechaEntrada" : ISODate("2020-12-29T00:00:00Z") }
{ "_id" : "VidJue", "item" : "Videojuegos", "qty" : 45, "Marca" : "EaSports, Activision, Sony, Microsoft", "Precio" : 70, "FechaEntrada" : ISODate("2020-12-15T00:00:00Z") }
{ "_id" : "VidCon", "item" : "Videoconsolas", "qty" : 10, "Marca" : "PlayStation, Microsoft, Nintendo", "Precio" : 400, "FechaEntrada" : ISODate("2020-12-09T00:00:00Z") }
{ "_id" : "Jueles", "item" : "JuegosdeMesa", "qty" : 23, "Marca" : "Hasbro", "Precio" : 30, "FechaEntrada" : ISODate("2020-12-15T00:00:00Z") }
> db.yumanyi.find( { $or: [ { "Marca": { $not: { $regex: /^s/, $options: 'i' } } }, { "FechaEntrada": { $ne: new Date(2020) } } ] } )
{ "_id" : "Pel", "item" : "Peluches", "qty" : 20, "Marca" : "Ty", "Precio" : 10, "FechaEntrada" : ISODate("2020-12-01T00:00:00Z") }
{ "_id" : "Firin", "item" : "FigurasAnimadas", "qty" : 15, "Marca" : "Banpresto", "Precio" : 24, "FechaEntrada" : ISODate("2020-12-29T00:00:00Z") }
```

```
File Edit Selection View Go Run Terminal Help
InsertMany.js - TrabajoMongo - Visual Studio Code

EXPLORER
> OPEN EDITORS 1 UNSAVED
> TRABAJO MONGO
> doc
> src
  JS Consultas.js M
  JS InsertMany.js M
  Readme.md

InsertMany.js
1 db.yumanyi.insertMany([
2   { "_id": "Pel", "item": "Peluches", "qty": 20, "Marca": "Ty", "Precio": 10, "FechaEntrada": new Date("2020-12-1") },
3   { "_id": "Firin", "item": "FigurasAnimadas", "qty": 15, "Marca": "Banpresto", "Precio": 24, "FechaEntrada": new Date("2020-12-29") },
4   { "_id": "VidJue", "item": "Videojuegos", "qty": 45, "Marca": ("EaSports, Activision, Sony, Microsoft"), "Precio": 70, "FechaEntrada": new Date("2020-12-15") },
5   { "_id": "VidCon", "item": "Videoconsolas", "qty": 10, "Marca": ("PlayStation, Microsoft, Nintendo"), "Precio": 400, "FechaEntrada": new Date("2020-12-9") },
6   { "_id": "Jueles", "item": "JuegosdeMesa", "qty": 23, "Marca": ("Hasbro"), "Precio": 30, "FechaEntrada": new Date("2020-12-15") } ] )

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: mongo

> db.yumanyi.find( { Marca: { $not: { $regex: /^T/, $options: 'i' } } } )
{ "_id" : "Firin", "item" : "FigurasAnimadas", "qty" : 15, "Marca" : "Banpresto", "Precio" : 24, "FechaEntrada" : ISODate("2020-12-29T00:00:00Z") }
{ "_id" : "VidJue", "item" : "Videojuegos", "qty" : 45, "Marca" : "EaSports, Activision, Sony, Microsoft", "Precio" : 70, "FechaEntrada" : ISODate("2020-12-15T00:00:00Z") }
{ "_id" : "VidCon", "item" : "Videoconsolas", "qty" : 10, "Marca" : "PlayStation, Microsoft, Nintendo", "Precio" : 400, "FechaEntrada" : ISODate("2020-12-09T00:00:00Z") }
{ "_id" : "Jueles", "item" : "JuegosdeMesa", "qty" : 23, "Marca" : "Hasbro", "Precio" : 30, "FechaEntrada" : ISODate("2020-12-15T00:00:00Z") }
> db.yumanyi.find( { $or: [ { "Marca": { $not: { $regex: /^s/, $options: 'i' } } }, { "FechaEntrada": { $ne: new Date(2020) } } ] } )
{ "_id" : "Pel", "item" : "Peluches", "qty" : 20, "Marca" : "Ty", "Precio" : 10, "FechaEntrada" : ISODate("2020-12-01T00:00:00Z") }
{ "_id" : "Firin", "item" : "FigurasAnimadas", "qty" : 15, "Marca" : "Banpresto", "Precio" : 24, "FechaEntrada" : ISODate("2020-12-29T00:00:00Z") }
```