

Ejercicio práctico de MongoDB

Para resolver este ejercicio se utilizara el 'mongo shell' de MongoDB.

Debe conectarse a una instancia activa de mongod, se puede utilizar la base de datos test, o crear una nueva llamada `mongo_movies`, o crear otra a elección.

Documentar todas sus consultas en un archivo, para usar como referencia y almacenarlo en su repositorio particular.

Documentos a Insertar

Insertar los siguientes documentos en una colección llamada `movies`.

```
title : Fight Club
writer : Chuck Palahniuk
year : 1999
actors : [
  Brad Pitt
  Edward Norton
]

title : Pulp Fiction
writer : Quentin Tarantino
year : 1994
actors : [
  John Travolta
  Uma Thurman
]

title : Inglorious Basterds
writer : Quentin Tarantino
year : 2009
actors : [
  Brad Pitt
  Diane Kruger
  Eli Roth
]

title : The Hobbit: An Unexpected Journey
writer : J.R.R. Tolkein
year : 2012
franchise : The Hobbit

title : The Hobbit: The Desolation of Smaug
writer : J.R.R. Tolkein
year : 2013
franchise : The Hobbit

title : The Hobbit: The Battle of the Five Armies
writer : J.R.R. Tolkein
year : 2012
franchise : The Hobbit
synopsis : Bilbo and Company are forced to engage in a war against an array of combatants
and keep the Lonely Mountain from falling into the hands of a rising darkness.

title : Pee Wee Herman's Big Adventure

title : Avatar
```

Consultas / Buscar documentos

Realizar las siguientes consultas en la colección `movies`:

1. Obtener todos los documentos
`db.movies.find()`
2. Obtener documentos con `writer` igual a "Quentin Tarantino"
`db.movies.find({writer: /^quentin tarantino/i},{_id:0})`
3. Obtener documentos con `actors` que incluyan a "Brad Pitt"
`db.movies.find({actors: /brad pitt/i}, {_id:0})`
4. Obtener documentos con `franchise` igual a "The Hobbit"
`db.movies.find({franchise: {$eq:'The Hobbit'}},{_id:0})`
5. Obtener todas las películas de los 90s.
`db.movies.find({year: {$lte:1999}},{_id:0})`
6. Obtener las películas estrenadas entre el año 2000 y 2010.
`db.movies.find({year: {$gte:2000}},{_id:0})`

Actualizar Documentos

1. Agregar sinopsis a "The Hobbit: An Unexpected Journey" : "A reluctant hobbit, Bilbo Baggins, sets out to the Lonely Mountain with a spirited group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."
`db.movies.updateMany({title: {$eq:"The Hobbit: An Unexpected Journey"}}, {$set:{synopsis: "A reluctant hobbit, Bilbo Baggins, sets out to the Lonely Mountain with a spirited group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."}})`
2. Agregar sinopsis a "The Hobbit: The Desolation of Smaug" : "The dwarves, along with Bilbo Baggins and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and magical ring."
`db.movies.updateMany({title: {$eq:"The Hobbit: The Desolation of Smaug"}}, {$set:{synopsis: "The dwarves, along with Bilbo Baggins and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and magical ring."}})`
3. Agregar un actor llamado "Samuel L. Jackson" a la película "Pulp Fiction"
`db.movies.update({title:"Pulp Fiction"}, {$push:{actors: {$each:["Samuel L. Jackson"]}}})`

Búsqueda por Texto / Text Search

1. Encontrar las películas que en el título contengan la palabra "Hobbit"
`db.movies.find({title: /hobbit/i},{_id:0})`
2. Encontrar las películas que en la sinopsis contengan la palabra "Gandalf"
`db.movies.find({synopsis: /gandalf/i},{_id:0})`
3. Encontrar las películas que en la sinopsis contengan la palabra "Bilbo" y no la palabra "Gandalf"
`db.movies.find({$text:{$search:"bilbo -gandalf"}},{_id:0})`

4. Encontrar las películas que en la sinopsis contengan la palabra "dwarves" ó "hobbit"
db.movies.find({\$or:[{synopsis:/dwarves/},{synopsis:/hobbit/}]})
5. Encontrar las películas que en la sinopsis contengan la palabra "gold" y "dragon"
db.movies.find({\$or:[{synopsis:/gold/},{synopsis:/dragon/}]})

Eliminar Documentos

1. Eliminar la película "Pee Wee Herman's Big Adventure"
db.movies.deleteOne({title: "Pee Wee Herman's Big Adventure"})
2. Eliminar la película "Avatar"
db.movies.deleteOne({title: "Avatar"})

Ricardo Laurens