**Ejercicio práctico de MongoDB**

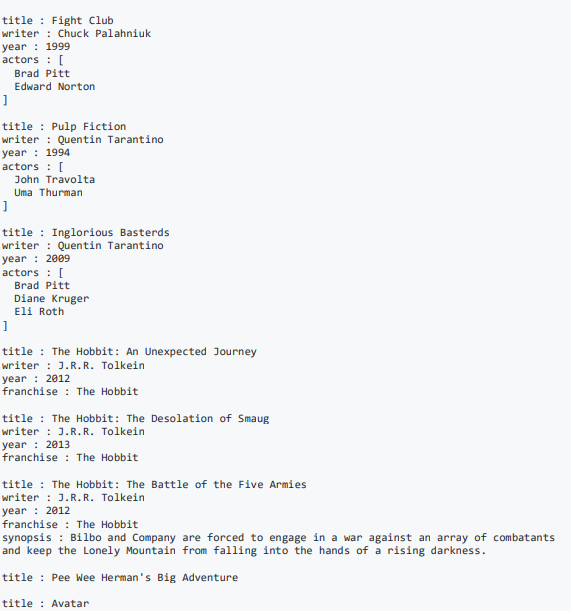
Para resolver este ejercicio se utilizara el 'mongo shell' de MongoDB.

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

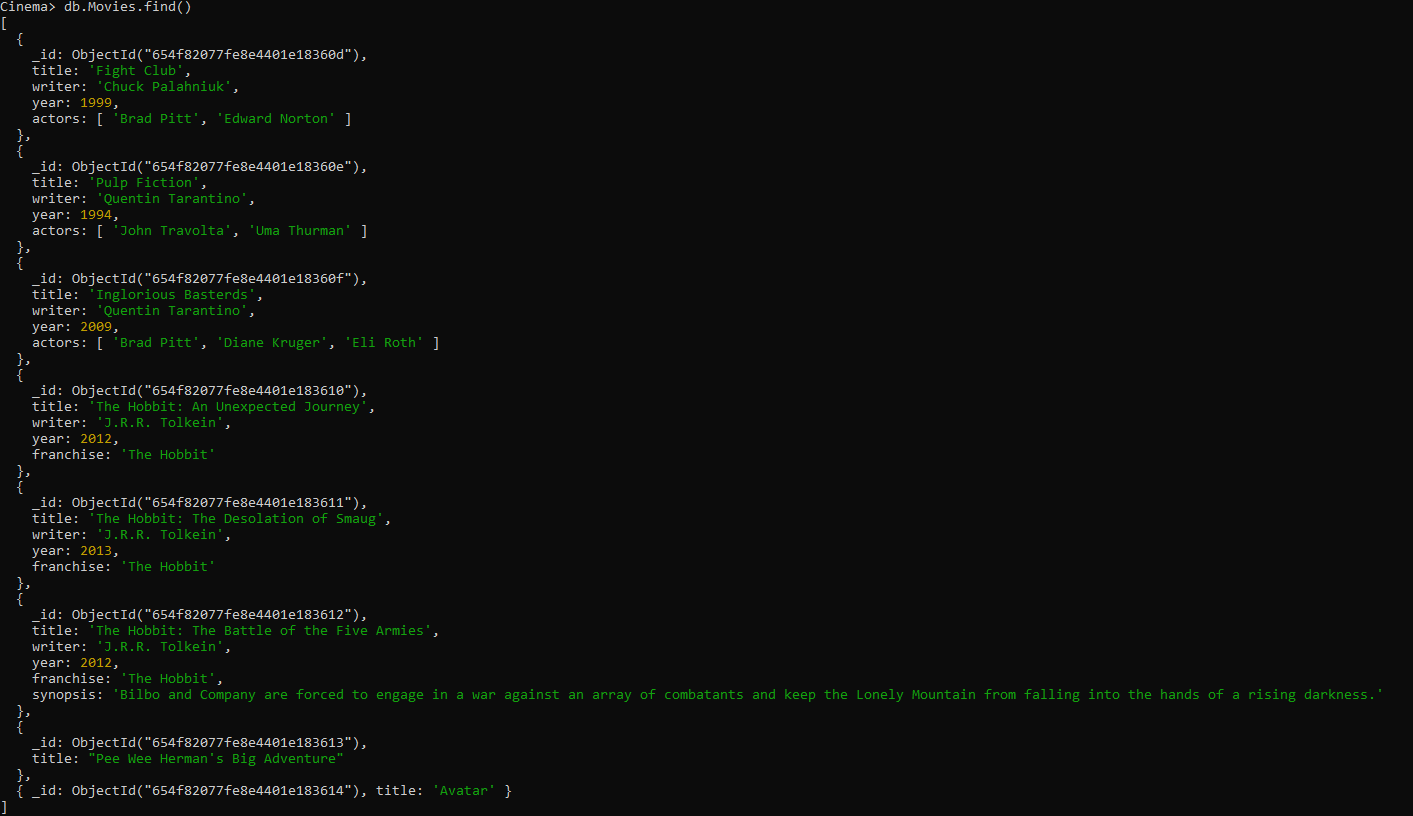


**Consultas / Buscar documentos**

Realizar las siguientes consultas en la colección movies:

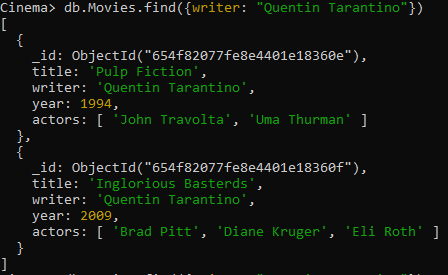
1. Obtener todos los documentos

Cinema> db.Movies.find()



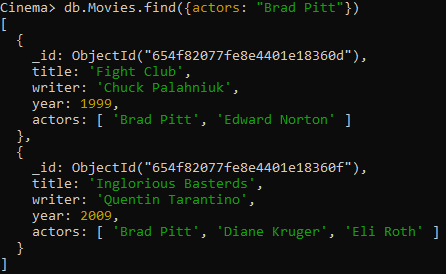
1. Obtener documentos con writer igual a "Quentin Tarantino"

db.Movies.find({writer: "Quentin Tarantino"})



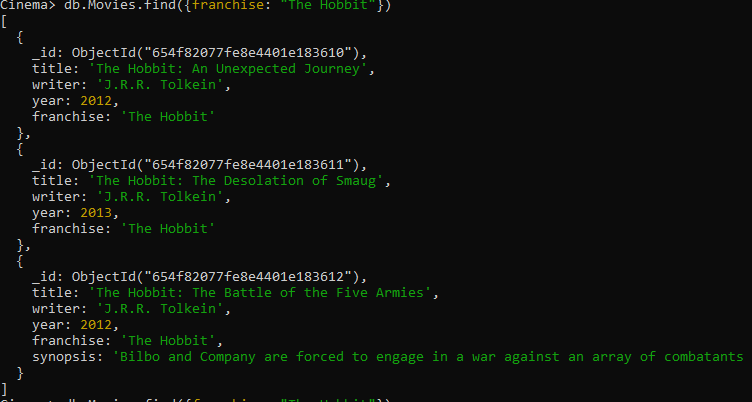
1. Obtener documentos con actors que incluyan a "Brad Pitt"

db.Movies.find({actors: "Brad Pitt"})



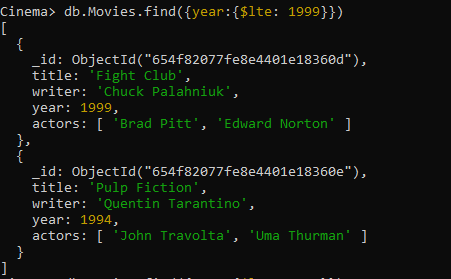
1. Obtener documentos con franchise igual a "The Hobbit"

db.Movies.find({franchise: "The Hobbit"})



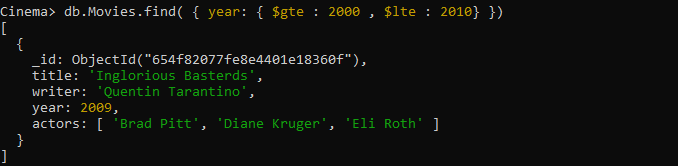
1. Obtener todas las películas de los 90s.

db.Movies.find({year:{$lte: 1999}})



1. Obtener las películas estrenadas entre el año 2000 y 2010.

db.Movies.find( { year: { $gte : 2000 , $lte : 2010} })

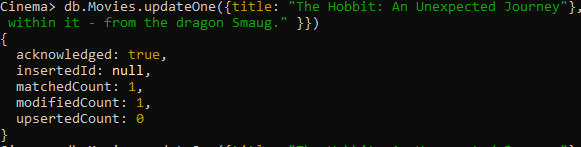


**Actualizar Documentos**

Actualizar Documentos

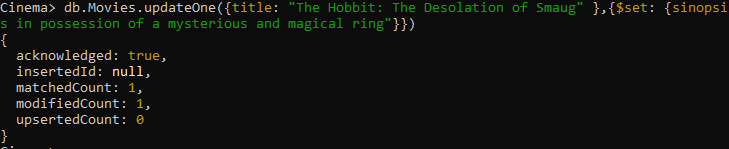
1. Agregar sinopsis a "The Hobbit: An Unexpected Journey" : "A reluctant hobbit, Bilbo Baggins, sets out to the Lonely Mountain with a spirited group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."

db.Movies.updateOne({title: "The Hobbit: An Unexpected Journey"},{$set: {sinopsis: "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." }})



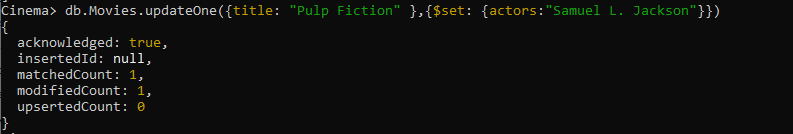
1. Agregar sinopsis a "The Hobbit: The Desolation of Smaug" : "The dwarves, along with Bilbo Baggins and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and magical ring”

db.Movies.updateOne({title: "The Hobbit: The Desolation of Smaug" },{$set: {sinopsis:"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"}})



1. Agregar una actor llamado "Samuel L. Jackson" a la película "Pulp Fiction"

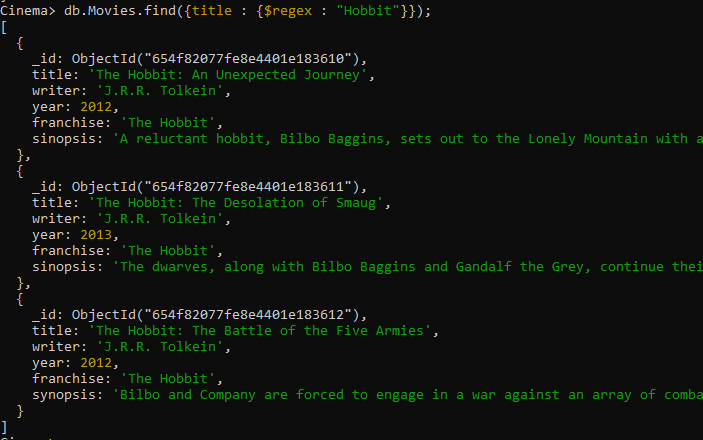
db.Movies.updateOne({title: "Pulp Fiction" },{$set: {actors:"Samuel L.Jackson"}})



**Búsqueda por Texto / Text Search**

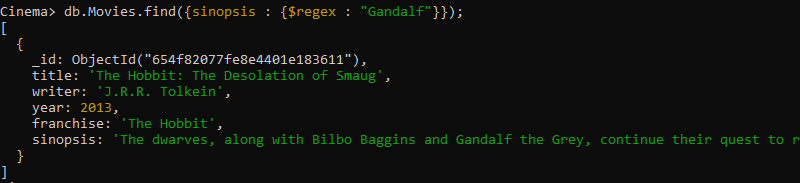
1. Encontrar las películas que en el título contengan la palabra "Hobbit"

db.Movies.find({title : {$regex : "Hobbit"}});



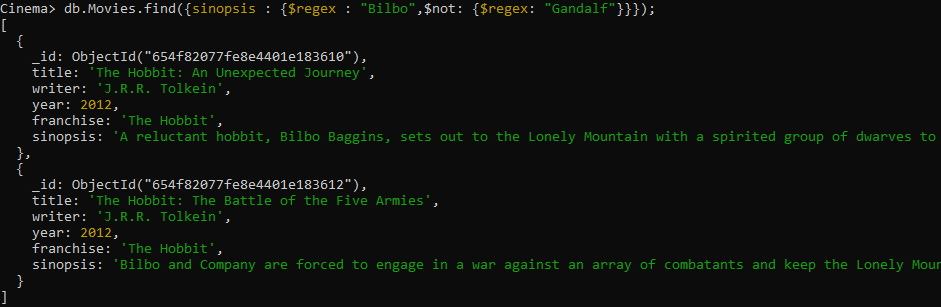
1. Encontrar las películas que en la sinopsis contengan la palabra "Gandalf"

db.Movies.find({sinopsis : {$regex : "Gandalf"}});



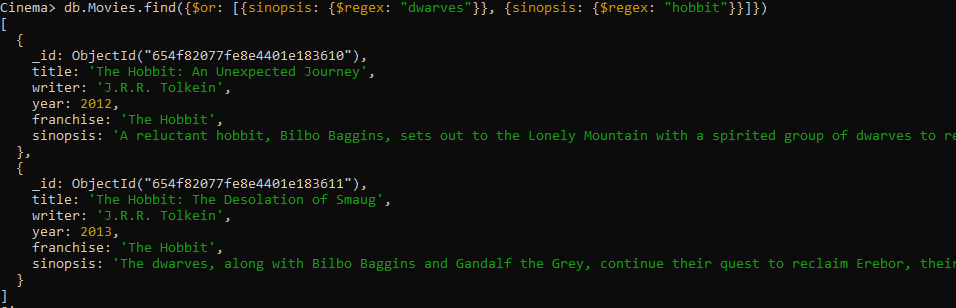
1. Encontrar las películas que en la sinopsis contengan la palabra "Bilbo" y no la palabra "Gandalf"

db.Movies.find({sinopsis : {$regex : "Bilbo",$not: {$regex: "Gandalf"}}});



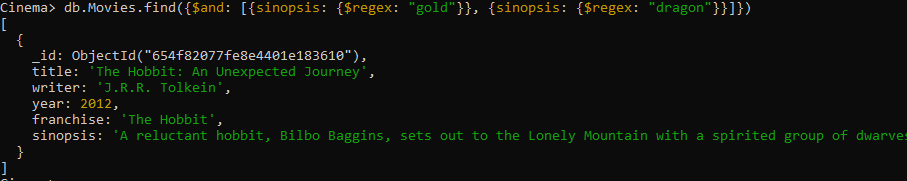
1. Encontrar las películas que en la sinopsis contengan la palabra "dwarves" ó "hobbit"

db.Movies.find({$or: [{sinopsis: {$regex: "dwarves"}}, {sinopsis: {$regex: "hobbit"}}]})



1. Encontrar las películas que en la sinopsis contengan la palabra "gold" y "dragon"

db.Movies.find({$and: [{sinopsis: {$regex: "gold"}}, {sinopsis: {$regex: "dragon"}}]})



**Eliminar Documentos**

1. Eliminar la película "Pee Wee Herman's Big Adventure"

db.Movies.deleteOne({title: "Pee Wee Herman's Big Adventure"})



1. Eliminar la película "Avatar

db.Movies.deleteOne({title: "Avatar”})



**Resultado Final Base de Datos**

