**MongoDB Assignment-1**

1. Insert Documents

Ans. db.mongo\_practice.insertMany();

Text

Description automatically generated with medium confidence

1. Query/ Find Documents

* Get all documents: db.mongo\_practice.find().pretty();

Text

Description automatically generatedText

Description automatically generated

* Get all documents with writer set to “Quentin Tarantino”

db.movies.find({writer:”Quentin Tarantino”})

Text

Description automatically generated

* Get all documents where actors include “Brad Pitt”.

db.movies.find({actors:”Brad Pitt”}).pretty();

Text

Description automatically generated

* Get all documents where franchise set to “The Hobbit”.

db.movies.find({franchise: “The Hobbit”}).pretty();

Text

Description automatically generated

* Get all movies released in 90s.

db.movies.find({year:{$gte:1990}, year:{$lte:1999}}).pretty();



* Get all movies released before 2000 and after 2010.

db.movies.find({$or:[{year:{$lte:2000}},{year:{$gte:2010}}]}).pretty()

Text

Description automatically generated

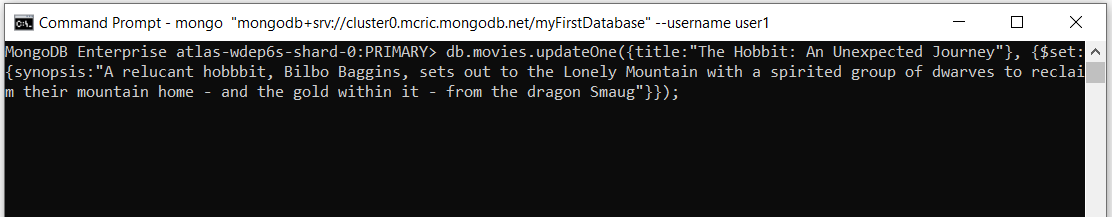
Text

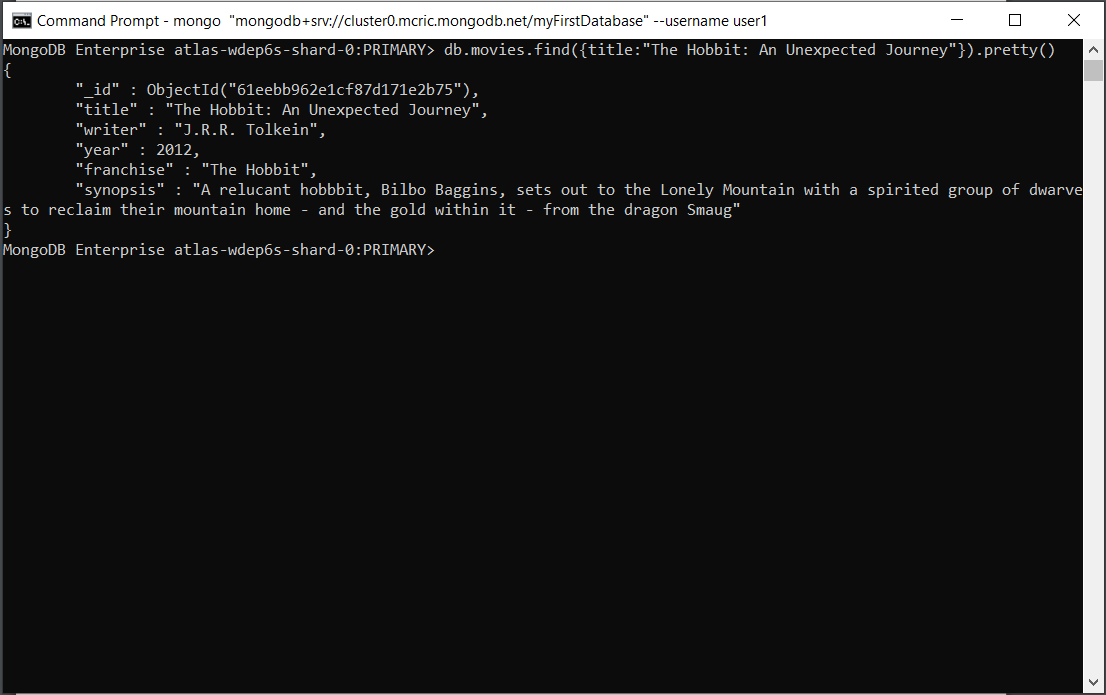
Description automatically generated

1. Update Documents

* add a synopsis to "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:{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.”}}).pretty();





* add a synopsis to "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:{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.”}}).pretty();

Text

Description automatically generated

* add an actor named "Samuel L. Jackson" to the movie "Pulp Fiction"

db.movies.updateOne({title:”Pulp Fiction”}, {$push:{actors:”Samuel L. Jackson”}})

Text

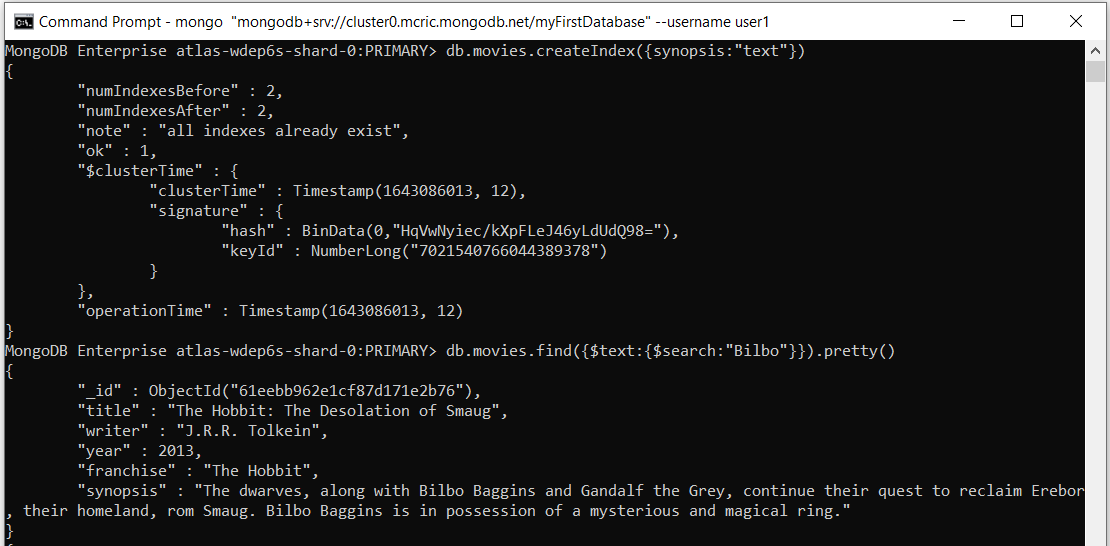
Description automatically generated

1. Text Search

* find all movies that have a synopsis that contains the word "Bilbo"

db.movies.createIndex({synopsis:”text”});

db.movies.find({$text:{$search:”Bilbo”}}).pretty();



Text

Description automatically generated

* find all movies that have a synopsis that contains the word "Gandalf".

db.movies.find({$text:{$search:”Gandalf”}}).pretty();

Text

Description automatically generated

* find all movies that have a synopsis that contains the word "Bilbo" and not the word "Gandalf"

Text

Description automatically generated

* find all movies that have a synopsis that contains the word "dwarves" or "hobbit"

Text

Description automatically generated

* find all movies that have a synopsis that contains the word "gold" and "dragon"

Text

Description automatically generated

1. Delete Documents

* delete the movie "Pee Wee Herman's Big Adventure"

Text

Description automatically generated

* delete the movie "Avatar"

Text

Description automatically generated

1. Relationships

* Insert the following documents into a users collection

username : GoodGuyGreg

first\_name : "Good Guy"

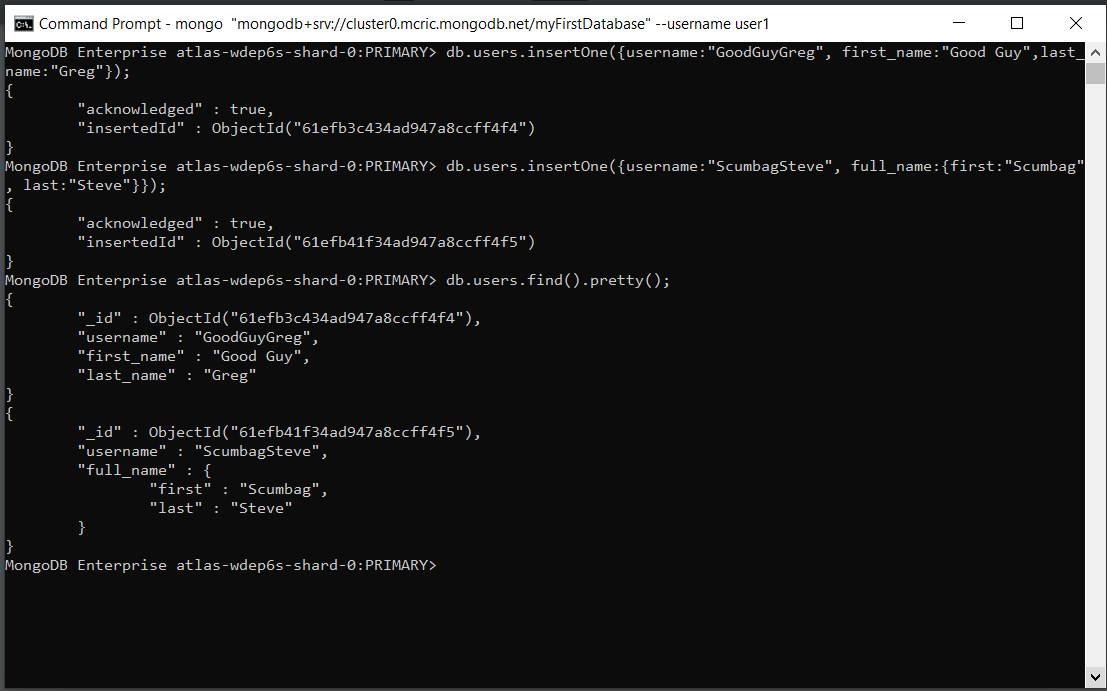
last\_name : "Greg"

username : ScumbagSteve

full\_name :

first : "Scumbag"

last : "Steve"



* Insert the following documents into a posts collection

username : GoodGuyGreg title : Passes out at party body : Wakes up early and cleans house

username : GoodGuyGreg title : Steals your identity body : Raises your credit score

username : GoodGuyGreg title : Reports a bug in your code body : Sends you a Pull Request

username : ScumbagSteve title : Borrows something body : Sells it

username : ScumbagSteve title : Borrows everything body : The end

username : ScumbagSteve title : Forks your repo on github body : Sets to private

Text

Description automatically generated

* Insert the following documents into a comments collection

username : GoodGuyGreg comment : Hope you got a good deal! post : [post\_obj\_id] where [post\_obj\_id] is the ObjectId of the posts document: "Borrows something"

username : GoodGuyGreg comment : What's mine is yours! post : [post\_obj\_id] where [post\_obj\_id] is the ObjectId of the posts document: "Borrows everything"

username : GoodGuyGreg comment : Don't violate the licensing agreement! post : [post\_obj\_id] where [post\_obj\_id] is the ObjectId of the posts document: "Forks your repo on github

username : ScumbagSteve comment : It still isn't clean post : [post\_obj\_id] where [post\_obj\_id] is the ObjectId of the posts document: "Passes out at party"

username : ScumbagSteve comment : Denied your PR cause I found a hack post : [post\_obj\_id] where [post\_obj\_id] is the ObjectId of the posts document: "Reports a bug in your code"

Text

Description automatically generated

1. Find all users

Text

Description automatically generated

1. Find all posts

Text

Description automatically generated

1. find all posts that was authored by "GoodGuyGreg"

Text

Description automatically generated

1. find all posts that was authored by "ScumbagSteve"

Text

Description automatically generated

1. Find all comments

Text

Description automatically generated

1. find all comments that was authored by "GoodGuyGreg"

Text

Description automatically generated

1. find all comments that was authored by "ScumbagSteve"

Text

Description automatically generated

1. find all comments belonging to the post "Reports a bug in your code"

Text

Description automatically generated

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*The end\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*