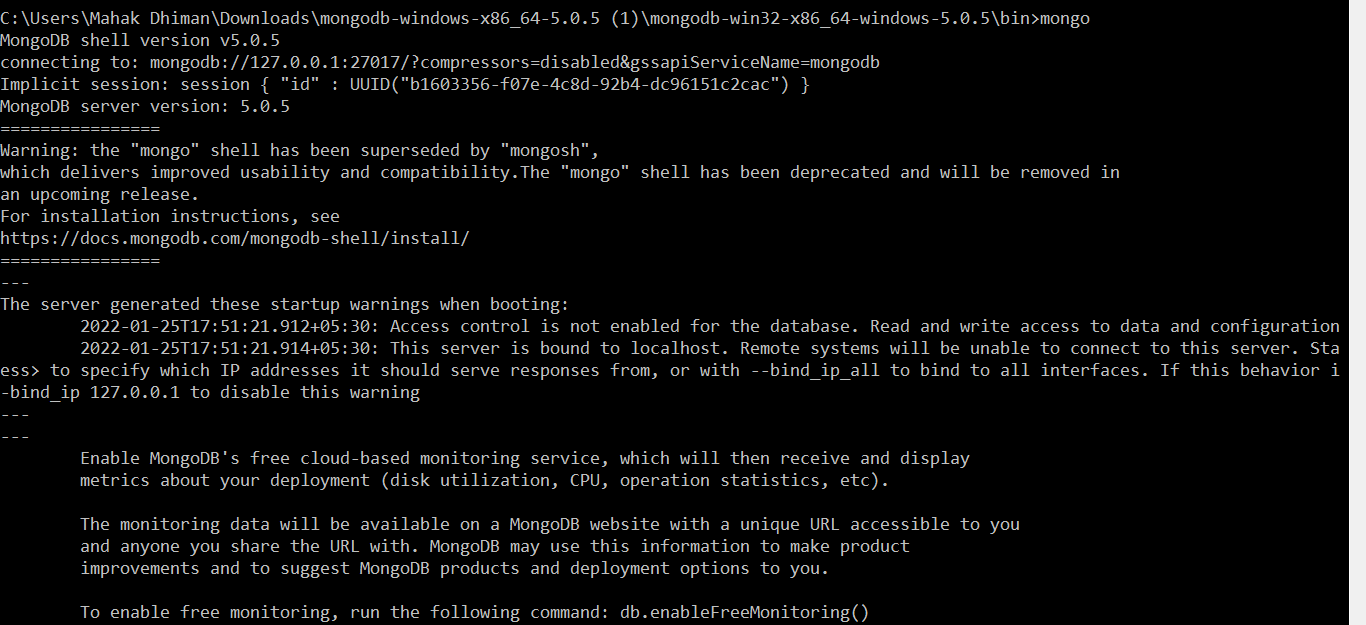
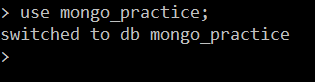
**BASIC ASSIGNMENT**

**Name: Mahak Dhiman**

**Email:** [**mahakdhiman710@gmail.com**](mailto:mahakdhiman710@gmail.com)

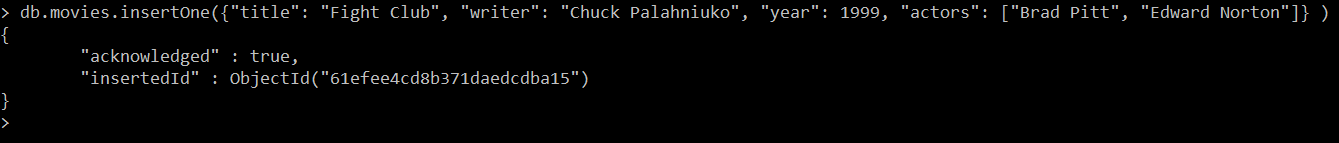
* MongoDB Exercise in mongo shell:  
  Connect to a running mongo instance, use a database named mongo\_practice. Document all your queries in a javascript file to use as a reference.



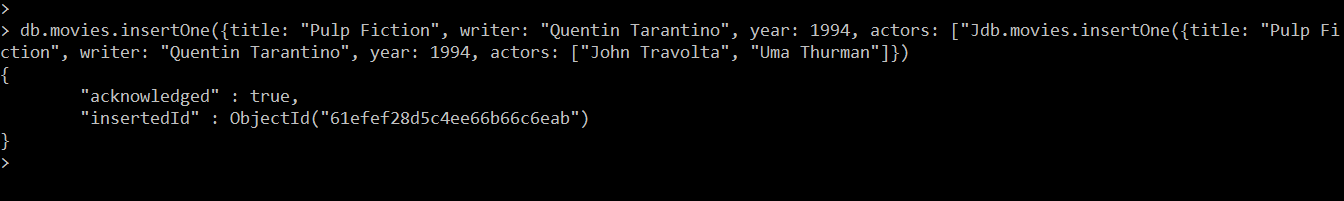


* Insert Documents: Insert the following documents into a movies collection.

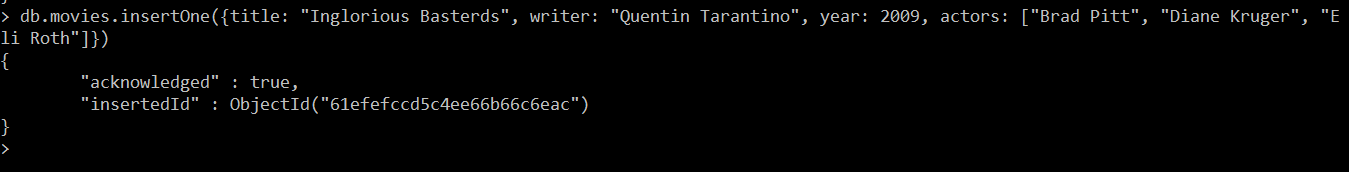
title : Fight Club writer : Chuck Palahniuko year : 1999 actors : [ Brad Pitt Edward Norton ]



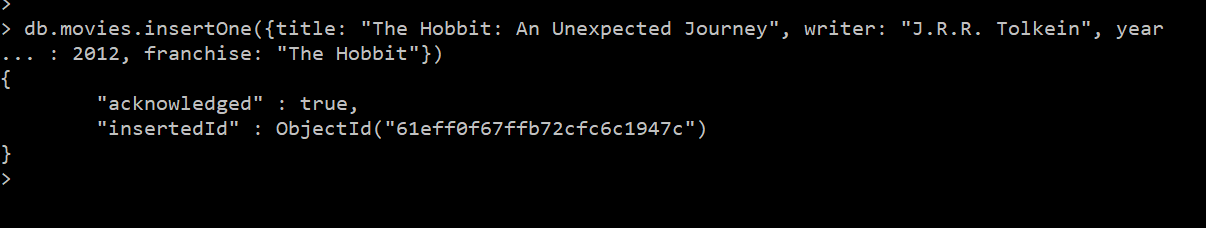
title : Pulp Fiction writer : Quentin Tarantino year : 1994 actors : [ John Travolta Uma Thurman ]



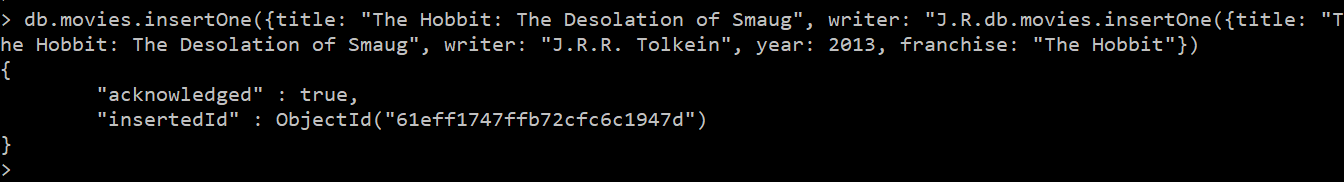
title : Inglorious Basterds writer : Quentin Tarantino year : 2009 actors : [ Brad Pitt Diane Kruger Eli Roth ]



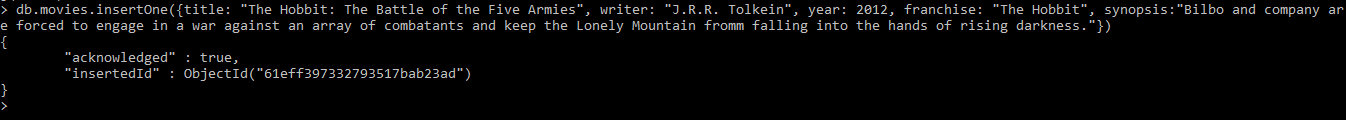
title : The Hobbit: An Unexpected Journey writer : J.R.R. Tolkein year : 2012 franchise : The Hobbit



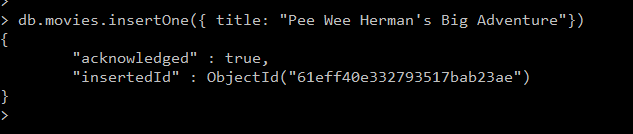
title : The Hobbit: The Desolation of Smaug writer : J.R.R. Tolkein year : 2013 franchise : The Hobbit



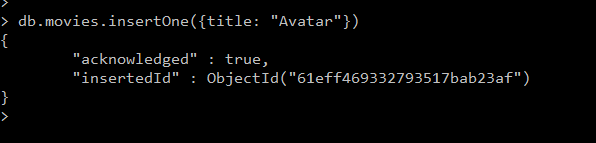
title : The Hobbit: The Battle of the Five Armies writer : J.R.R. Tolkein year : 2012 franchise : The Hobbit synopsis : Bilbo and Company are forced to engage in a war against an array of combatants and keep the Lonely Mountain from falling into the hands of a rising darkness.



title : Pee Wee Herman's Big Adventure

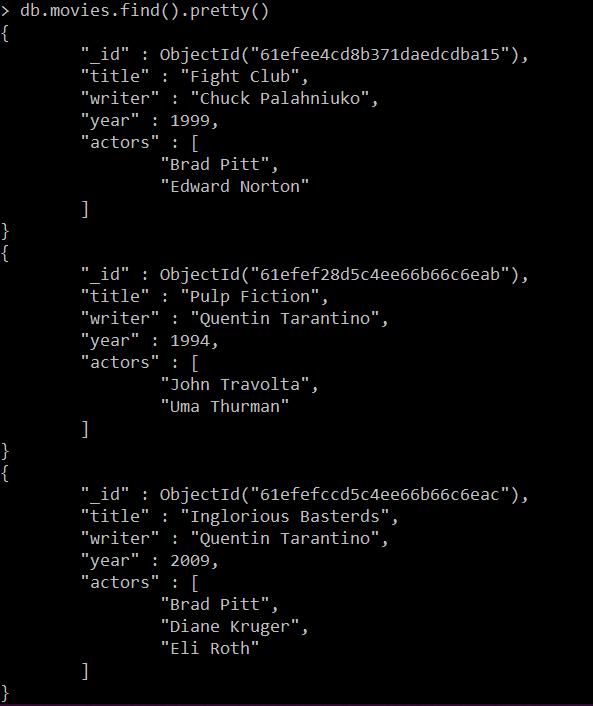


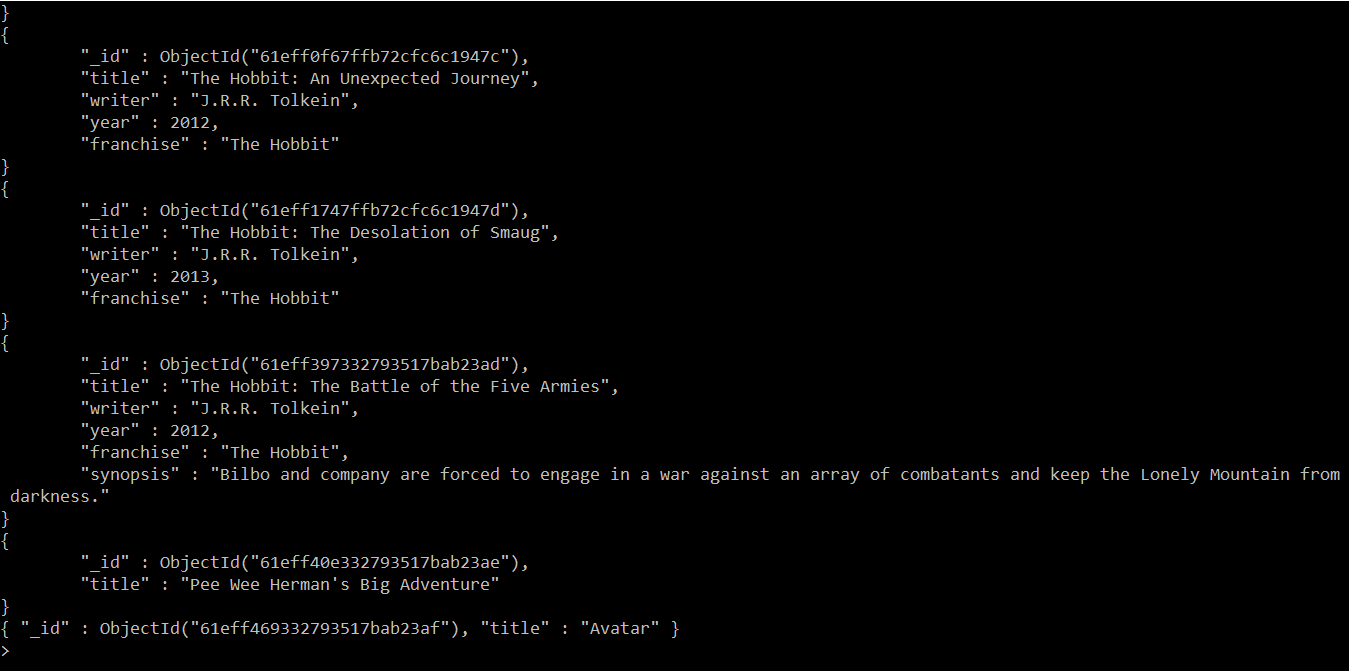
title : Avatar



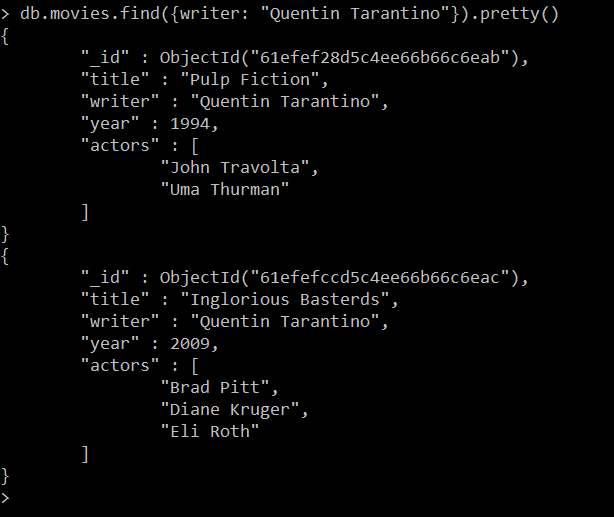
* Query / Find Documents: query the movies collection to

1. get all documents

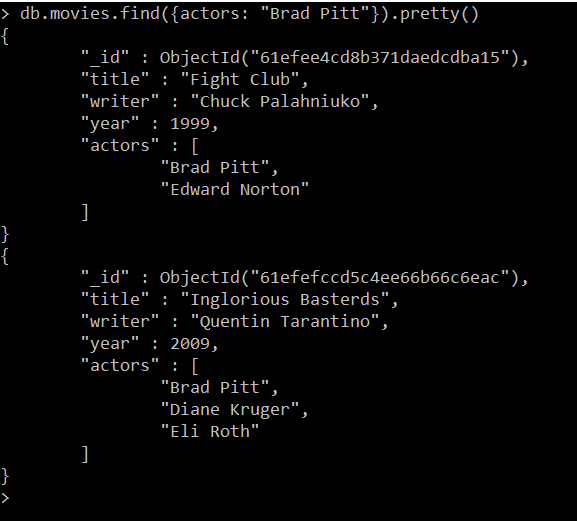




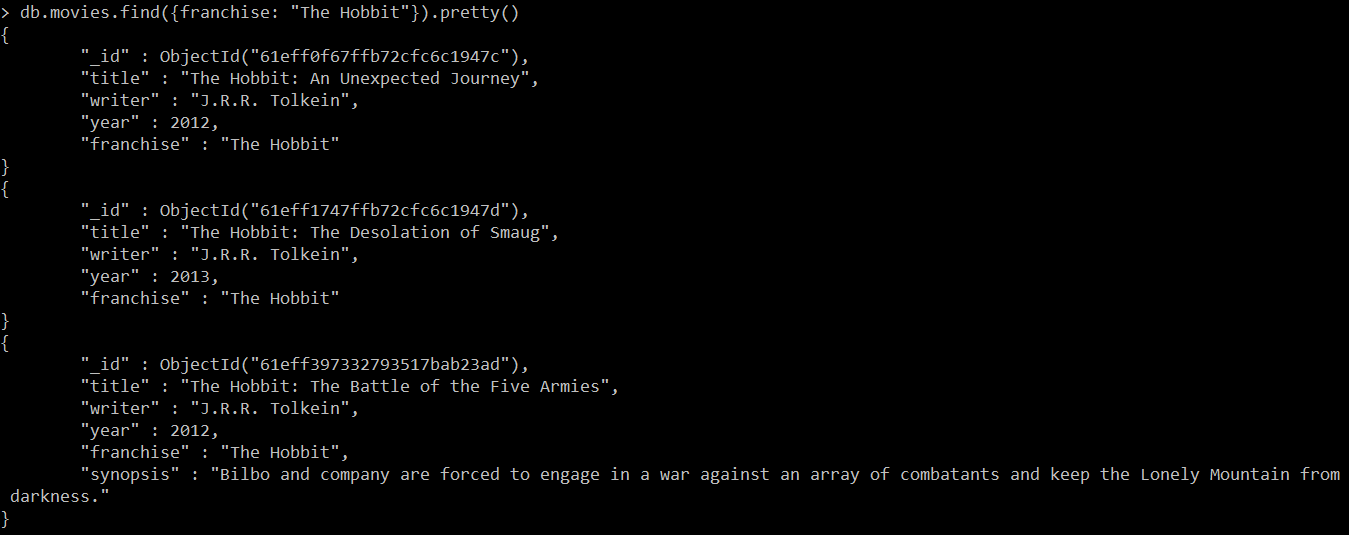
1. get all documents with writer set to "Quentin Tarantino"



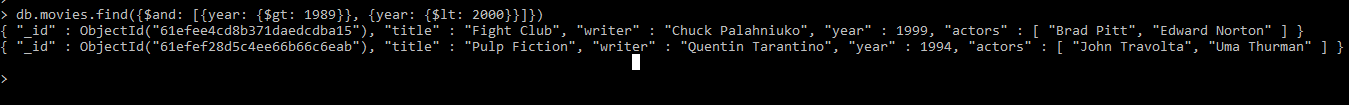
1. get all documents where actors include "Brad Pitt"



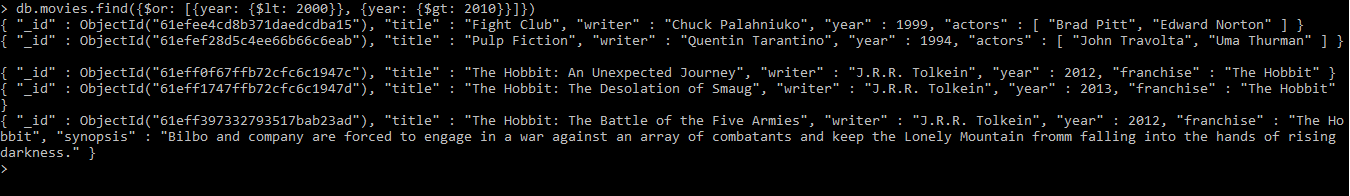
1. get all documents with franchise set to "The Hobbit"



1. get all movies released in the 90s

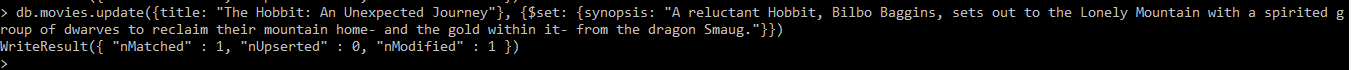


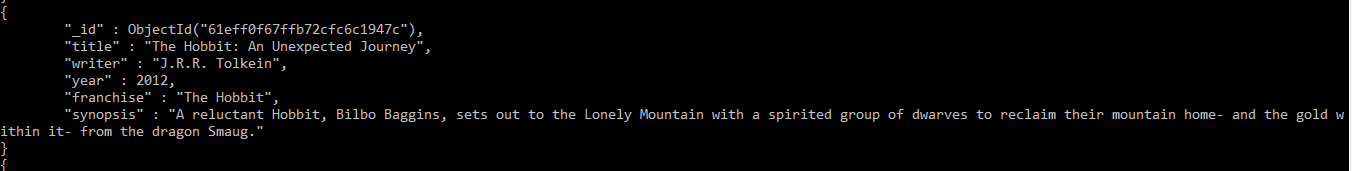
1. get all movies released before the year 2000 or after 2010



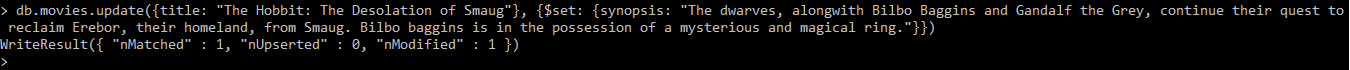
* Update Documents

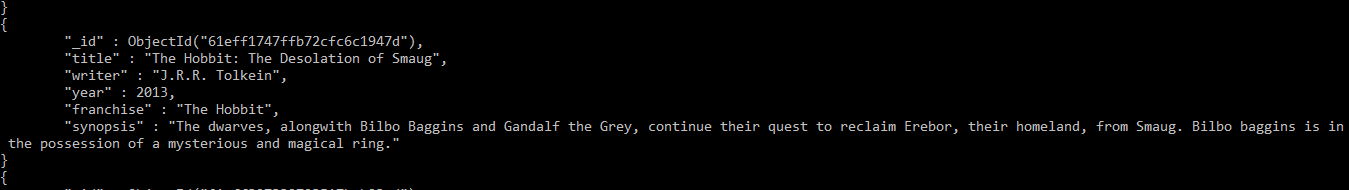
1. Add a synopsis to "The Hobbit: An Unexpected Journey" : "A reluctant hobbit, Bilbo Baggins, sets out to the Lonely Mountain with a spirited group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."



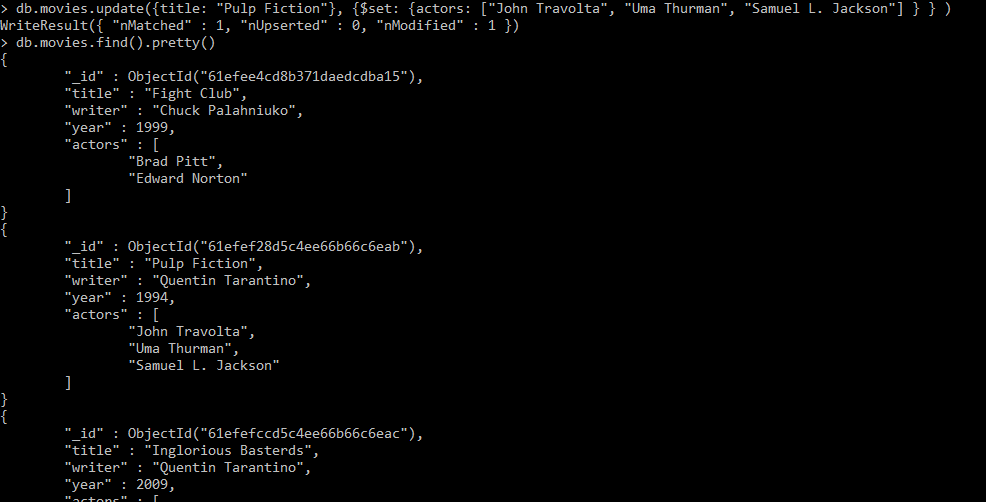


1. add a synopsis to "The Hobbit: The Desolation of Smaug" : "The dwarves, along with Bilbo Baggins and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and magical ring."



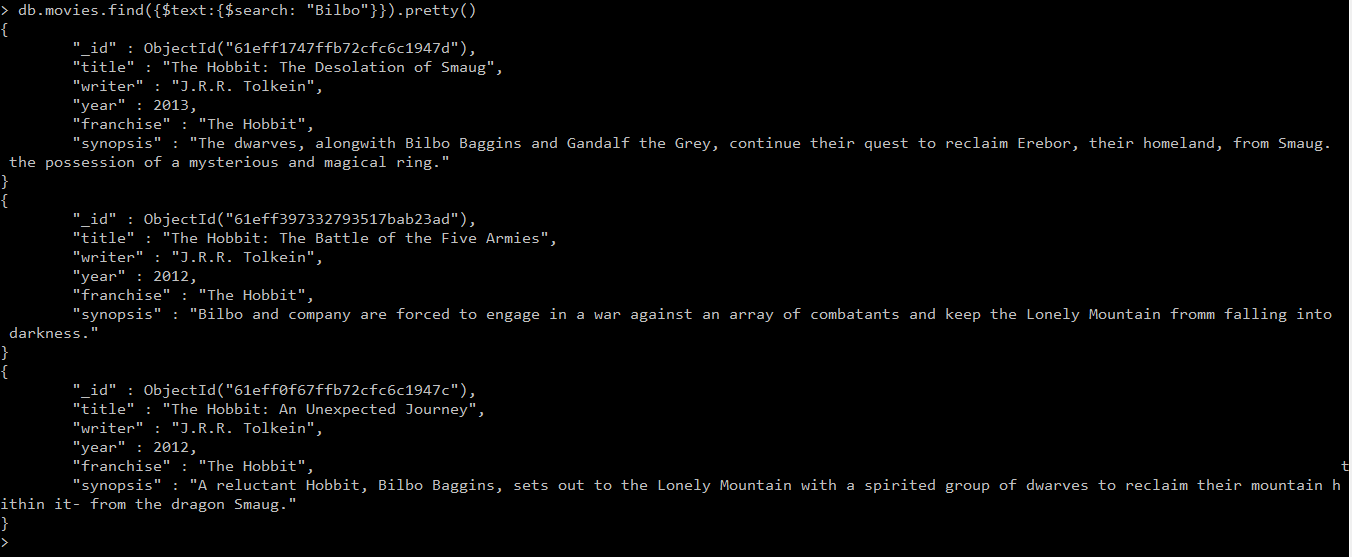
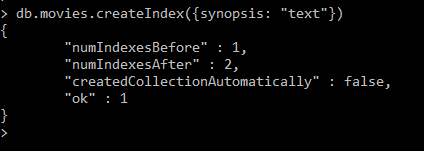


1. add an actor named "Samuel L. Jackson" to the movie "Pulp Fiction"

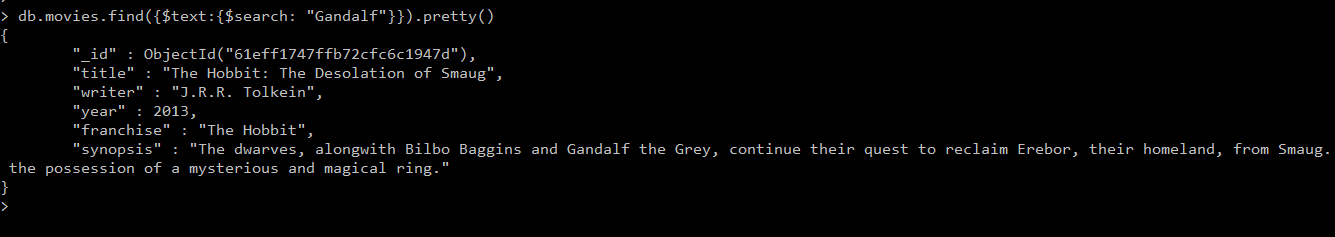


* Text Search:

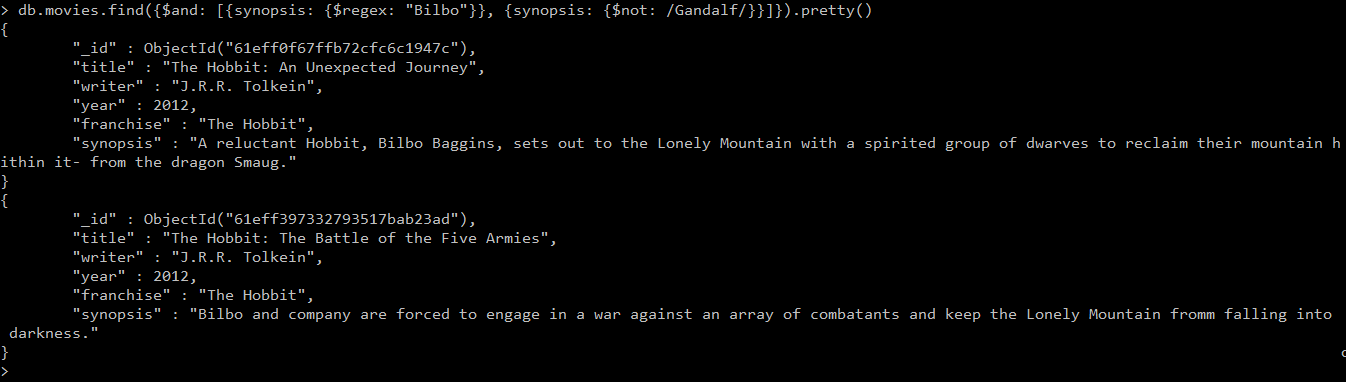
1. find all movies that have a synopsis that contains the word "Bilbo"



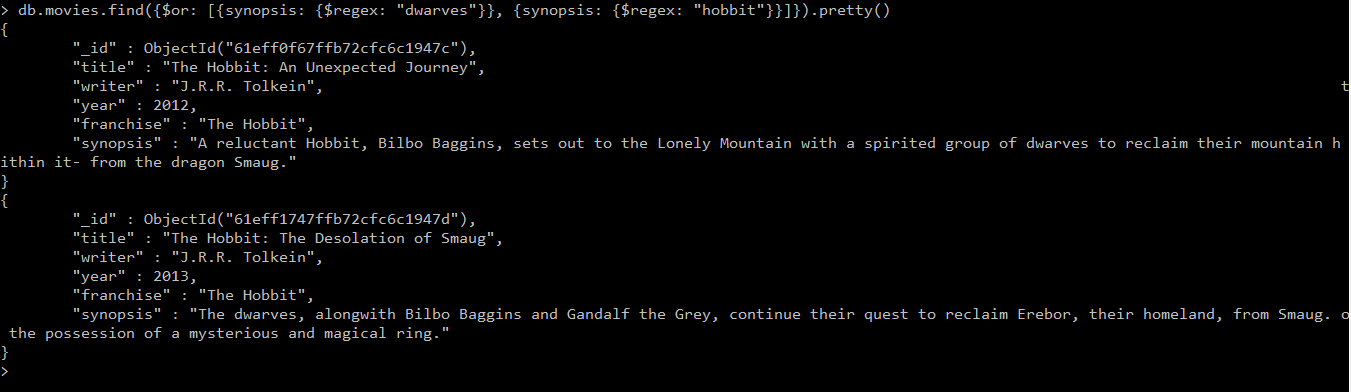
1. find all movies that have a synopsis that contains the word "Gandalf"



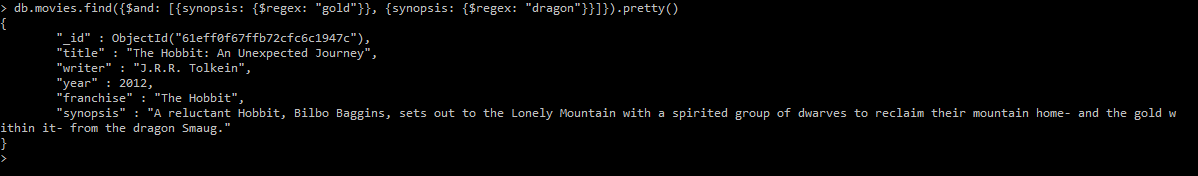
1. find all movies that have a synopsis that contains the word "Bilbo" and not the word "Gandalf"



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



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

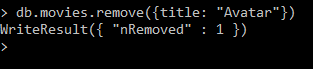


* Delete Documents:

1. delete the movie "Pee Wee Herman's Big Adventure"



1. delete the movie "Avatar"

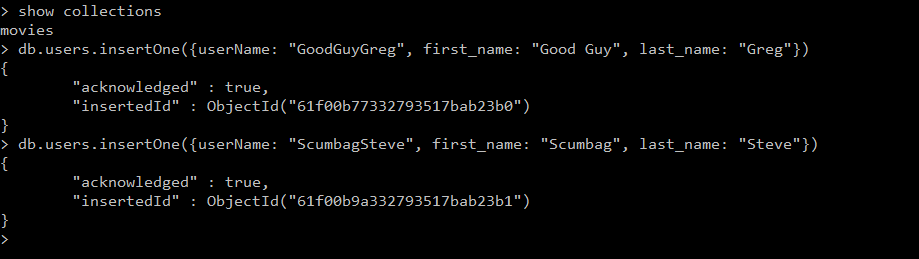


* Relationships

1. 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"

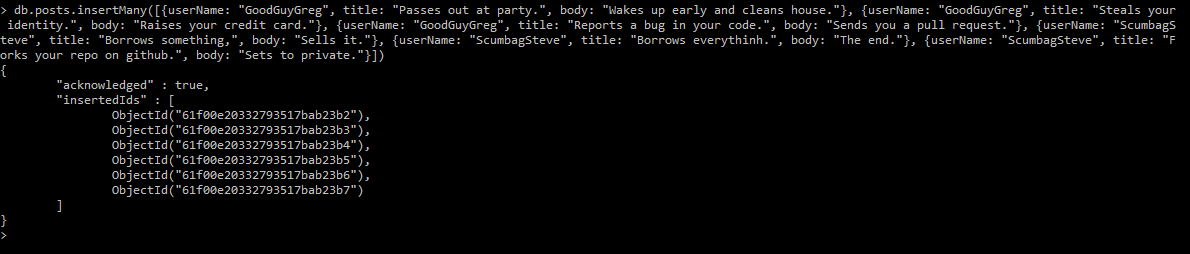


1. Insert the following documents into a posts collection

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



1. Insert the following documents into a comments collection

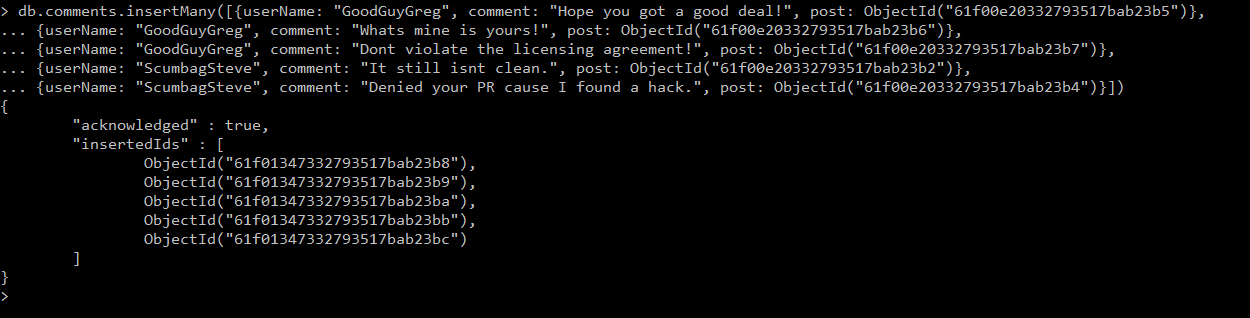
username : GoodGuyGreg comment : Hope you got a good deal! post : [post\_obj\_id] where [post\_obj\_id] is the ObjectId of the posts document: "Borrows something"

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

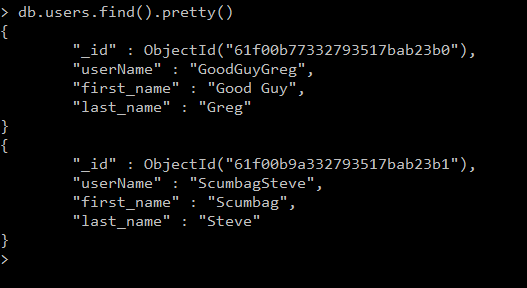
username : ScumbagSteve comment : It still isn't clean post : [post\_obj\_id] where [post\_obj\_id] is the ObjectId of the posts document: "Passes out at party"

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"

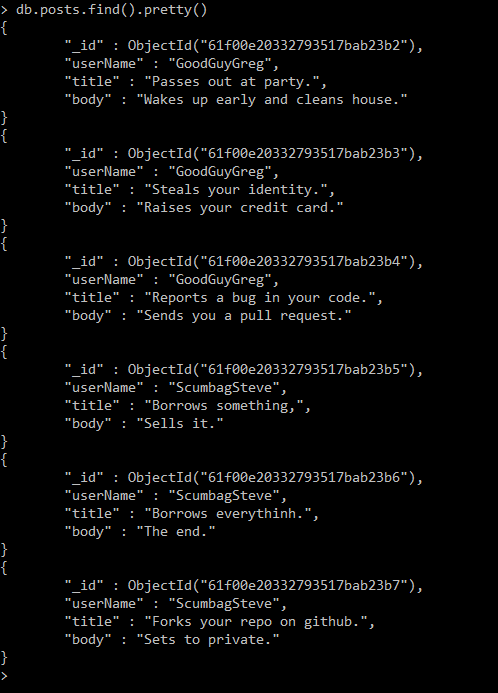


* Querying related collections:

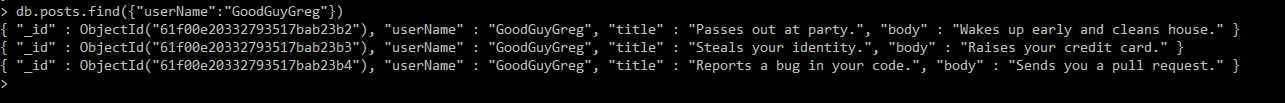
1. find all users



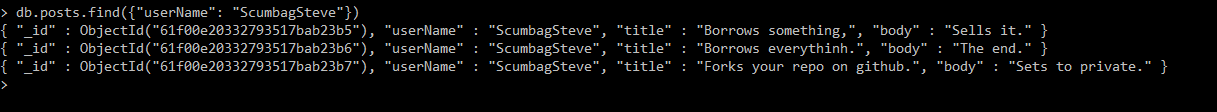
1. find all posts



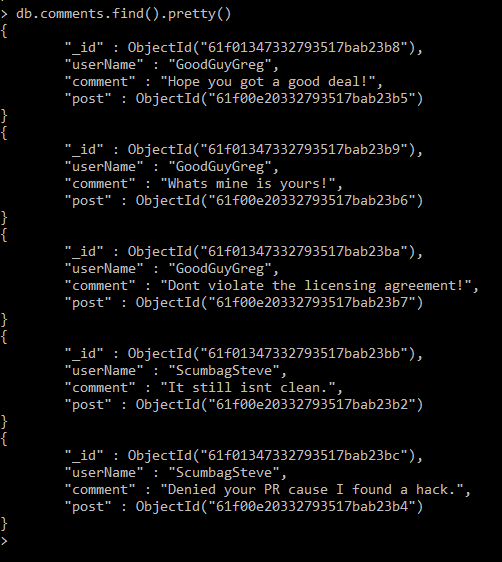
1. find all posts that was authored by "GoodGuyGreg"



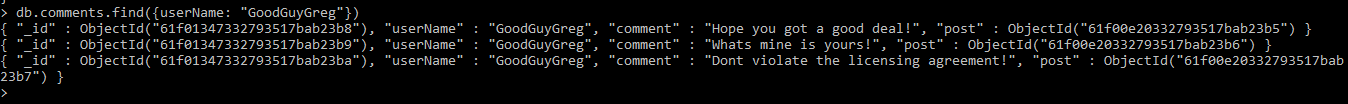
1. find all posts that was authored by "ScumbagSteve"



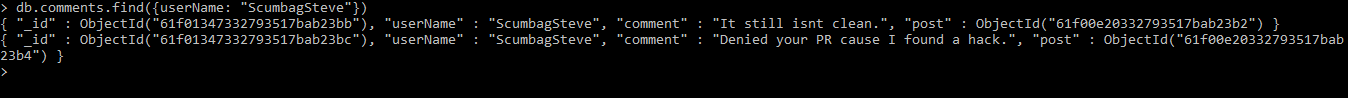
1. find all comments



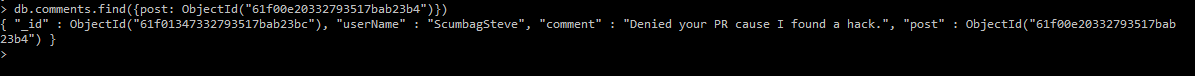
1. find all comments that was authored by "GoodGuyGreg"



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



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



----------------------------------------------------------(Finished)---------------------------------------------------