1. get all documents

db.movies.find()

{ "\_id" : ObjectId("5ffe77f9a24c1ad23dc52049"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : 1999, "actors" : [ { "actor1" : "brad pitt" }, { "actor2" : "Edward Norton" } ] }

{ "\_id" : ObjectId("5ffe7859a24c1ad23dc5204a"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : 1994, "actors" : [ { "actor1" : "John Tarvolta" }, { "actor2" : "Uma Thurman" } ] }

{ "\_id" : ObjectId("5ffe78d5a24c1ad23dc5204b"), "title" : "Inglorious Baterds", "writer" : "Quentin Tarantino", "year" : 2009, "actors" : [ { "actor1" : "Brad Pitt" }, { "actor2" : "Diane Kruger" }, { "actor3" : "Eli Roth" } ] }

{ "\_id" : ObjectId("5ffe796aa24c1ad23dc5204c"), "title" : "The Hobbit:An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe79aea24c1ad23dc5204d"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2013, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe80f7a24c1ad23dc5204e"), "title" : "The Hobbit:the Battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe904ba24c1ad23dc5204f"), "title" : "The Hobbit:the Battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Biblo and Companyy 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" }

{ "\_id" : ObjectId("5ffe909da24c1ad23dc52050"), "title" : "pee wee hermans Big Adventure" }

{ "\_id" : ObjectId("5ffe90aea24c1ad23dc52051"), "title" : "Avatar" }

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

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

{ "\_id" : ObjectId("5ffe7859a24c1ad23dc5204a"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", }

{ "\_id" : ObjectId("5ffe78d5a24c1ad23dc5204b"), "title" : "Inglorious Baterds", "writer" : "Quentin Tarantino", "year" : 2009, "actors" : [ { "actor1" : "Brad Pitt" }, { "actor2" : "Diane Kruger" }, { "actor3" : "Eli Roth" } ] }

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

db.movies.find({‘actor.actor1’:’brad pitt’}

> db.movies.find({'actors.actor1':'brad pitt'})

{ "\_id" : ObjectId("5ffe77f9a24c1ad23dc52049"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : 1999, "actors" : [ { "actor1" : "brad pitt" }, { "actor2" : "Edward Norton" } ] }

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

Db.movies.find({‘franchise’:’The Hobbit’})

> db.movies.find({'franchise':'The Hobbit'})

{ "\_id" : ObjectId("5ffe796aa24c1ad23dc5204c"), "title" : "The Hobbit:An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe79aea24c1ad23dc5204d"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2013, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe80f7a24c1ad23dc5204e"), "title" : "The Hobbit:the Battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe904ba24c1ad23dc5204f"), "title" : "The Hobbit:the Battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Biblo and Companyy 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" }

1. get all movies released in the 90s

db.movies.find({'year':{$gt:1899,$lt:2000}})

db.movies.find({'year':{$gt:1899,$lt:2000}})

{ "\_id" : ObjectId("5ffe77f9a24c1ad23dc52049"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : 1999, "actors" : [ { "actor1" : "brad pitt" }, { "actor2" : "Edward Norton" } ] }

{ "\_id" : ObjectId("5ffe7859a24c1ad23dc5204a"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : 1994, "actors" : [ { "actor1" : "John Tarvolta" }, { "actor2" : "Uma Thurman" } ] }

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

> db.movies.find({$or:[{'year':{$gt:2010}},{'year':{$lt:2000}}]})

{ "\_id" : ObjectId("5ffe77f9a24c1ad23dc52049"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : 1999, "actors" : [ { "actor1" : "brad pitt" }, { "actor2" : "Edward Norton" } ] }

{ "\_id" : ObjectId("5ffe7859a24c1ad23dc5204a"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : 1994, "actors" : [ { "actor1" : "John Tarvolta" }, { "actor2" : "Uma Thurman" } ] }

{ "\_id" : ObjectId("5ffe796aa24c1ad23dc5204c"), "title" : "The Hobbit:An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe79aea24c1ad23dc5204d"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2013, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe80f7a24c1ad23dc5204e"), "title" : "The Hobbit:the Battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("5ffe904ba24c1ad23dc5204f"), "title" : "The Hobbit:the Battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Biblo and Companyy 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" }

Updating 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."

> db.movies.update({'title':'The Hobbit:An Unexpected Journey'},{$set :{'synopsis':'Areluctant hobbit,Biblo Baggins ,sets out to the Lonely Mountain with a spirited groupof dwarves to reclaim their mountain home- and the gold within it - from the dragon Smug,'}})

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."

db.movies.update({'title':'The Hobbit:The Desolation of smaug'},{$set:{'synopsis':'The dwarves, alonng with Biblo Baggins and Gandalf the Grey,Continue their quest to reclaim Erebor , their homeland, from Smaug.Biblo Baggins is in possession of a mysterious and magical ring.'}})

WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

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

> db.movies.update({'title':'Pulp Fiction'},{$push:{'actors':'Samuel L.Jackson'}})

WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

text search:

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

db.movies.find({ synopsis:{$regex:"Biblo"}})

{ "\_id" : ObjectId("5ffe796aa24c1ad23dc5204c"), "title" : "The Hobbit:An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Areluctant hobbit,Biblo Baggins ,sets out to the Lonely Mountain with a spirited groupof dwarves to reclaim their mountain home- and the gold within it - from the dragon Smug," }

{ "\_id" : ObjectId("5ffe904ba24c1ad23dc5204f"), "title" : "The Hobbit:the Battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Biblo and Companyy 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" }

{ "\_id" : ObjectId("5ffe9fb9bfaf2e3c79ebdc04"), "title" : "The Hobbit:The Desolation of smaug", "writer" : "J.R.R.Tolkein", "year" : 2013, "franchise" : "The Hobbite", "synopsis" : "The dwarves, alonng with Biblo Baggins and Gandalf the Grey,Continue their quest to reclaim Erebor , their homeland, from Smaug.Biblo Baggins is in possession of a mysterious and magical ring." }

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

> db.movies.find({ synopsis:{$regex:"Gandalf"}})

{ "\_id" : ObjectId("5ffe9fb9bfaf2e3c79ebdc04"), "title" : "The Hobbit:The Desolation of smaug", "writer" : "J.R.R.Tolkein", "year" : 2013, "franchise" : "The Hobbite", "synopsis" : "The dwarves, alonng with Biblo Baggins and Gandalf the Grey,Continue their quest to reclaim Erebor , their homeland, from Smaug.Biblo Baggins is in possession of a mysterious and magical ring." }

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

> db.movies.find({$and:[{synopsis:{$regex:"Biblo"}},{synopsis:{$not:/Gandalf/}}]})

{ "\_id" : ObjectId("5ffe796aa24c1ad23dc5204c"), "title" : "The Hobbit:An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Areluctant hobbit,Biblo Baggins ,sets out to the Lonely Mountain with a spirited groupof dwarves to reclaim their mountain home- and the gold within it - from the dragon Smug," }

{ "\_id" : ObjectId("5ffe904ba24c1ad23dc5204f"), "title" : "The Hobbit:the Battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Biblo and Companyy 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" }

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

> db.movies.find({$or:[{synopsis:{$regex:"dwarves"}},{synopsis:{$regex:'hobbit'}}]})

{ "\_id" : ObjectId("5ffe796aa24c1ad23dc5204c"), "title" : "The Hobbit:An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Areluctant hobbit,Biblo Baggins ,sets out to the Lonely Mountain with a spirited groupof dwarves to reclaim their mountain home- and the gold within it - from the dragon Smug," }

{ "\_id" : ObjectId("5ffe9fb9bfaf2e3c79ebdc04"), "title" : "The Hobbit:The Desolation of smaug", "writer" : "J.R.R.Tolkein", "year" : 2013, "franchise" : "The Hobbite", "synopsis" : "The dwarves, alonng with Biblo Baggins and Gandalf the Grey,Continue their quest to reclaim Erebor , their homeland, from Smaug.Biblo Baggins is in possession of a mysterious and magical ring." }

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

> db.movies.find({$or:[{synopsis:{$regex:"dwarves"}},{synopsis:{$regex:'hobbit'}}]})

{ "\_id" : ObjectId("5ffe796aa24c1ad23dc5204c"), "title" : "The Hobbit:An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Areluctant hobbit,Biblo Baggins ,sets out to the Lonely Mountain with a spirited groupof dwarves to reclaim their mountain home- and the gold within it - from the dragon Smug," }

{ "\_id" : ObjectId("5ffe9fb9bfaf2e3c79ebdc04"), "title" : "The Hobbit:The Desolation of smaug", "writer" : "J.R.R.Tolkein", "year" : 2013, "franchise" : "The Hobbite", "synopsis" : "The dwarves, alonng with Biblo Baggins and Gandalf the Grey,Continue their quest to reclaim Erebor , their homeland, from Smaug.Biblo Baggins is in possession of a mysterious and magical ring." }

> db.movies.find({$or:[{synopsis:{$regex:"dragon"}},{synopsis:{$regex:'gold'}}]})

{ "\_id" : ObjectId("5ffe796aa24c1ad23dc5204c"), "title" : "The Hobbit:An Unexpected Journey", "writer" : "J.R.R.Tolkein", "year" : 2012, "franchise" : "The Hobbit", "synopsis" : "Areluctant hobbit,Biblo Baggins ,sets out to the Lonely Mountain with a spirited groupof dwarves to reclaim their mountain home- and the gold within it - from the dragon Smug," }

1. find all users?

> db.users.find()

{ "\_id" : ObjectId("5ffeda4065c16b5a80eb81ba"), "username" : "GoodGuyGerg", "first\_name" : "Good Guy", "last\_name" : "Greg" }

{ "\_id" : ObjectId("5ffedad365c16b5a80eb81bb"), "username" : "ScumbagSteve", "fullname" : [ { "first" : "Scumbag", "last" : "Steve" } ] }

2.find all posts?

> db.posts.find().pretty()

{

"\_id" : ObjectId("5ffedb6065c16b5a80eb81bc"),

"username" : "GoodGuyGreg",

"title" : "Passes out at party",

"body" : "Wakes up early and cleans house"

}

{

"\_id" : ObjectId("5ffedb9665c16b5a80eb81bd"),

"username" : "GoodGuyGreg",

"title" : "Steals your identity",

"body" : "Raises your credit score"

}

{

"\_id" : ObjectId("5ffedbf065c16b5a80eb81be"),

"username" : "GoodGuyGreg",

"title" : "Reports a bug in your code",

"body" : "Sends you a pull request"

}

{

"\_id" : ObjectId("5ffedc6865c16b5a80eb81bf"),

"username" : "ScumbagSteve",

"title" : "Borrows everything",

"body" : "Sells it"

}

{

"\_id" : ObjectId("5ffedc8a65c16b5a80eb81c0"),

"username" : "ScumbagSteve",

"title" : "Borrows somthing",

"body" : "the end"

}

{

"\_id" : ObjectId("5ffedcba65c16b5a80eb81c1"),

"username" : "ScumbagSteve",

"title" : "Forks your repo on github",

"body" : "Sets to private"

}

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

> db.posts.find({'username':'GoodGuyGreg'})

{ "\_id" : ObjectId("5ffedb6065c16b5a80eb81bc"), "username" : "GoodGuyGreg", "title" : "Passes out at party", "body" : "Wakes up early and cleans house" }

{ "\_id" : ObjectId("5ffedb9665c16b5a80eb81bd"), "username" : "GoodGuyGreg", "title" : "Steals your identity", "body" : "Raises your credit score" }

{ "\_id" : ObjectId("5ffedbf065c16b5a80eb81be"), "username" : "GoodGuyGreg", "title" : "Reports a bug in your code", "body" : "Sends you a pull request" }

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

> db.posts.find({'username':'ScumbagSteve'})

{ "\_id" : ObjectId("5ffedc6865c16b5a80eb81bf"), "username" : "ScumbagSteve", "title" : "Borrows everything", "body" : "Sells it" }

{ "\_id" : ObjectId("5ffedc8a65c16b5a80eb81c0"), "username" : "ScumbagSteve", "title" : "Borrows somthing", "body" : "the end" }

{ "\_id" : ObjectId("5ffedcba65c16b5a80eb81c1"), "username" : "ScumbagSteve", "title" : "Forks your repo on github", "body" : "Sets to private" }

>

1. find all comments

> db.comments.find().pretty()

{

"\_id" : ObjectId("5ffee4413498c9b97709c033"),

"username" : "GoodGuyGreg",

"comment" : "What mine is yours!",

"post" : ObjectId("5ffedc8a65c16b5a80eb81c0")

}

{

"\_id" : ObjectId("5ffee4c33498c9b97709c034"),

"username" : "ScumbagSteve",

"comment" : "denied your PR cause i found a hack",

"post" : ObjectId("5ffedbf065c16b5a80eb81be")

}

{

"\_id" : ObjectId("5ffee4fb3498c9b97709c035"),

"username" : "ScumbagSteve",

"comment" : "it still isnt clean",

"post" : ObjectId("5ffedcba65c16b5a80eb81c1")

}

{

"\_id" : ObjectId("5ffee51d3498c9b97709c036"),

"username" : "GoodGuyGreg",

"comment" : "Dont violate the licensing agreemend",

"post" : ObjectId("5ffedcba65c16b5a80eb81c1")

}

{

"\_id" : ObjectId("5ffee5433498c9b97709c037"),

"username" : "GoodGuyGreg",

"comment" : "Hope you got a good deal",

"post" : ObjectId("5ffedc6865c16b5a80eb81bf")

}

6 find all comments that was authored by "GoodGuyGreg"

> db.comments.find({'username':'GoodGuyGreg'})

{ "\_id" : ObjectId("5ffee4413498c9b97709c033"), "username" : "GoodGuyGreg", "comment" : "What mine is yours!", "post" : ObjectId("5ffedc8a65c16b5a80eb81c0") }

{ "\_id" : ObjectId("5ffee51d3498c9b97709c036"), "username" : "GoodGuyGreg", "comment" : "Dont violate the licensing agreemend", "post" : ObjectId("5ffedcba65c16b5a80eb81c1") }

{ "\_id" : ObjectId("5ffee5433498c9b97709c037"), "username" : "GoodGuyGreg", "comment" : "Hope you got a good deal", "post" : ObjectId("5ffedc6865c16b5a80eb81bf") }

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

> db.comments.find({'username':'ScumbagSteve'})

{ "\_id" : ObjectId("5ffee4c33498c9b97709c034"), "username" : "ScumbagSteve", "comment" : "denied your PR cause i found a hack", "post" : ObjectId("5ffedbf065c16b5a80eb81be") }

{ "\_id" : ObjectId("5ffee4fb3498c9b97709c035"), "username" : "ScumbagSteve", "comment" : "it still isnt clean", "post" : ObjectId("5ffedcba65c16b5a80eb81c1") }

>

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

db.posts.aggregate([{$match:{title:"Reports a bug in your code"}},

{$lookup:{from:"comments",

localField:"\_id",

foreignField:"post",

as:"comments"}}]).pretty()