db.movies.insert({title : "Fight Club",

writer : "Chuck Palahniuko",

year : 1999,

actors :[

"Brad Pitt",

"Edward Norton"]

})

------>

db.movies.insert({title : "Pulp Fiction" ,

writer : "Quentin Tarantino" ,

year : 1994,

actors : [

"John Travolta",

"Uma Thurman"]

})

--------->

db.movies.insert({title : "Inglorious Basterds",

writer : "Quentin Tarantino",

year : 2009,

actors : [

"Brad Pitt",

"Diane Kruger",

"Eli Roth"

]

})

------------->

db.movies.insert({title : "The Hobbit: An Unexpected Journey",

writer : "J.R.R. Tolkein",

year : 2012,

franchise : "The Hobbit"

})

---------------->

db.movies.insert({title : "The Hobbit: The Desolation of Smaug",

writer : "J.R.R. Tolkein",

year : 2013,

franchise : "The Hobbit"

})

------------------>

db.movies.insert({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"

})

1.)db.movies.find()

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

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

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

5.)db.movies.find({$and: [{year: {$gt: 1900}}, {year: {$lt: 2000}}]})

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

----------------------

Update Documents--->

----------------------

1.) db.movies.update(

{"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." }});

2.)db.movies.update(

{"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." }});

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

---------------------------

Text Search ---->

---------------------------

1.)db.movies.createIndex({synopsis:"text"}) && db.movies.find({$text: {$search: "Bilbo"}})

2.)db.movies.find({$text: {$search: "Gandalf"}})

3.)db.movies.find({$text: {$search: "Bilbo -Gandalf"}})

4.)db.movies.find({$text: {$search: "dwarves hobbit"}})

5.)db.movies.find({$text: {$search: "gold dragon"}})

--------------------------

Delete Documents----->

--------------------------

1.)db.movies.remove({title: "Pee Wee Herman's Big Adventure"}) // db.movies.deleteOne({title : "Pee Wee Herman's Big Adventure"})

2.)db.movies.remove({title: "Avatar"}) // db.movies.deleteOne({title : Avatar})

---------------------------------

Querying related collections --------->

---------------------------------

1.) db.users.find()

2.)db.posts.find()

3.)db.posts.find({username: "GoodGuyGreg"})

4.)db.posts.find({username: "ScumbagSteve"})

5.)db.comments.find()

6.)db.comments.find({username: "GoodGuyGreg"})

7.)db.comments.find({username: "ScumbagSteve"})

8.)db.posts.aggregate([

{

$match: { title: 'Reports a bug in your code' }

},

{

$lookup: {

from: 'comments',

localField: '\_id',

foreignField: 'post',

as: 'comments'

}

}

]).pretty();