db.Movies.insert({'title':'Fight Club','writer':'Chuck Palahniuko','year':1999,'actors':['Brad Pitt','Edward Norton']})

db.Movies.insert({'title':'Pulp Fiction','writer':'Quantin Taratino','year':1994,'actors':['John Travolta','Uma Thurman']})

db.Movies.insert({'title':'Inglorious Basterds','writer':'Quantin Taratino','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' })

db.Movies.insert({'title':'Pee Wee Hermans Big Adventure'})

db.Movies.insert({'title':'Avatar'})

Find Documents

1. db.Movies.find()
2. db.Movies.find({'writer':'Quantin Taratino'})
3. db.Movies.find({'actors':'Brad Pitt'})
4. db.Movies.find({'Franchise':'The Hobbit'})
5. db.Movies.find({'year':{$gte:1990,$lte:1999}})
6. db.Movies.find({ $or: [{ year: { $lt: 2000 } }, { year: { $gt: 2010 } }] }).pretty()

Update Documents

1. db.Movies.update({title: "The Hobbit: An Unexpected Journey"}, {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"}, {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.find({ synopsis: /Bilbo/g })
2. db.Movies.find({ 'synopsis': /Gandalf/g })
3. db.Movies.find({ $and: [{ synopsis: /Bilbo/g }, { synopsis: { $not: /Gandalf/g } }] })
4. db.Movies.find({ synopsis: /(dwarves|hobbit)/g })
5. db.Movies.find({ $and: [{ synopsis: /gold/g }, { synopsis: /dragon/g }] })

Delete Documents

1. db.Movies.deleteMany({'title' : 'Pee Wee Hermans Big Adventure'})
2. db.Movies.deleteMany({'title' : 'Avatar'})

Relationships:

Inserting into users:

db.createCollection('users')

use users

1. db.users.insert({'username':'GoodGuyGreg','first\_name':'Good Guy','last\_name':'Greg'})
2. db.users.insert({'username':'ScumbagSteve','fullname':{'first':'Scumbag','last':'Steve'}})

INSERTNG INTO POSTS

db.posts.insertMany([  
{  
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"  
}  
])

INSERT INTO COMMENTS COLLECTION

db.comments.insertMany([  
{  
username: "GoodGuyGreg",  
comment: "Hope you got a good deal!",  
post: ObjectId("61f141d4c7b7b60d42db0ea1")  
},  
{  
username: "GoodGuyGreg",  
comment: "Don't violate the licensing agreement!",  
post: ObjectId("61f141d4c7b7b60d42db0ea2")  
},  
{  
username: "GoodGuyGreg",  
comment: "Don't violate the licensing agreement!",  
post: ObjectId("61f141d4c7b7b60d42db0ea3")  
},  
{  
username: "ScumbagSteve",  
comment: "It still isn't clean",  
post: ObjectId("61f141d4c7b7b60d42db0e9e")  
},  
{  
username: "ScumbagSteve",  
comment: "Denied your PR cause I found a hack",  
post: ObjectId("61f141d4c7b7b60d42db0ea0")  
}  
])

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.find({'title':'Reports a bug in your code'})