**INSERT THE FOLLOWING DOCUMENTS INTO A MOVIES COLLECTION.**

db.movies.insert(

{title:"Fight Club",

writer: "Chuck Palahniuk",

year: "1999",

actors:

["Brad Pitt",

"Edward Norton"]

})

db.movies.insert(

{title:"Pulp Fiction",

writer:"Quentin Tarantino",

year:"2009",

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 Tolkien",

year:"2013",

franchise:"The Hobbit"

})

db.movies.insert(

{title:"The Hobbit: The Battle of the Five Armies",

writer:"J.R.R Tolkien",

year:"2002", 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 Herman's Big Adventures"} )

db.movies.insert({ title:"Avatar" })

**Query / Find Documents**

**query the movies collection to**

1. get all documents

db.movies.find()

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

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

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

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

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

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

1. get all movies released in the 90s

db.movies.find({year:{$gt:"1990", $lt:"2000"}})

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

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

**Update Documents**

1. db.movies.update (

{\_id:ObjectId("5c9f98e5e5c2dfe9b3729bfe")},

{$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."}

})

1. db.movies.update(

{\_id:ObjectId("5c9fa42ae5c2dfe9b3729c03")},

{$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."}

})

1. db.movies.update(

{\_id:ObjectId("5c9f983ce5c2dfe9b3729bfc")},

{$push: {actors:"Samuel L. Jackson"}

})

**Text Search**

1. db.movies.find ({synopsis: {$regex:"Bilbo"}})
2. db.movies.find ({synopsis: {$regex:"Gandalf"}})
3. db.movies.find ({$and: [{synopsis:{$regex:"Bilbo"}},

{synopsis: {$not:/Gandalf/}}]

})

1. db.movies.find ({$or: [{synopsis:{$regex:"dwarves"}},

{synopsis: {$regex:"hobbit"}}]

})

1. db.movies.find ({ $and: [{synopsis:{$regex:"gold"}},

{synopsis: {$regex:"dragon"}}]

})

**Delete Documents**

1. db.movies.remove ( {\_id:ObjectId("5c9f992ae5c2dfe9b3729c00" ) } )
2. db.movies.remove ( {\_id:ObjectId("5c9f9936e5c2dfe9b3729c01") } )

**Relationships**

**Insert the following documents into a users collection**

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

**Insert the following documents into a posts collection**

1. db.posts.insert(

{username:"GoodGuyGreg",

title:"Passes out at Party",

body:"Raises your credit score" } )

1. db.posts.insert(

{ username:"GoodGuyGreg",

title:"Steals your identity",

body:"Raises your credit score" } )

1. db.posts.insert(

{username:"GoodGuyGreg",

title:"Reports a bug in your code",

body:"Sends you a pull request" } )

1. db.posts.insert(

{ username:"ScumbagSteve",

title:"Borrows something",

body:"Sells it" } )

1. db.posts.insert(

{ username:"ScumbagSteve",

title:"Borrows everything",

body:"The end" } )

1. db.posts.insert(

{username:"ScumbagSteve",

title:"Forks your repo on github",

body:"Sets to private" } )

Insert the following into a comments collection

1. db.comments.insert(

{ username:"GoodGuyGreg",

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

post:ObjectId("5ca0b7e96435f98b5901f463")

} )

1. db.comments.insert(

{username:"GoodGuyGreg",

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

post:ObjectId("5ca0b9706435f98b5901f46a")

} )

1. db.comments.insert(

{username:"GoodGuyGreg",

comment:"Don't violate the licensing agreement!",

post:ObjectId("5ca0b8766435f98b5901f467")

} )

1. db.comments.insert(

{username:"ScumbagSteve",

comment:"It still isn't clean",

post:ObjectId("5ca0b8546435f98b5901f466")

} )

1. db.comments.insert(

{username:"ScumbagSteve",

comment:"Denied your PR cause I found a hack",

post:ObjectId("5ca0b9256435f98b5901f469")

} )

**Querying related collections**

1. db.users.find().pretty()
2. db.posts.find().pretty()
3. db.posts.find({username:"GoodGuyGreg"})
4. db.posts.find({username:"ScumbagSteve"})
5. db.comments.find().pretty()
6. db.comments.find({username:"GoodGuyGreg"})
7. db.comments.find({username:"ScumbagSteve"})