Inserting movies collection

>>db.movies.insertMany([{"title" : "Fight Club",

"writer" : "Chuck Palahniuk",

"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"}])

1.select all documents

>>db.movies.find().pretty()

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

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

[

{

\_id: ObjectId("63892fc6f086c3dc93796097"),

title: 'Pulp Fiction',

writer: 'Quentin Tarantino',

year: 1994,

actors: [ 'John', 'Travolta', 'Uma', 'Thurman' ]

},

{

\_id: ObjectId("63892fc6f086c3dc93796098"),

title: 'Inglorious Basterds',

writer: 'Quentin Tarantino',

year: 2009,

actors: [ 'Brad', 'Pitt', 'Diane', 'Kruger', 'Eli', 'Roth' ]

}

]

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

MongoDataBase> db.movies.find({actors:{$in:["Brad"," Pitt"]}})

[

{

\_id: ObjectId("63892fc6f086c3dc93796096"),

title: 'Fight Club',

writer: 'Chuck Palahniuk',

year: 1999,

actors: [ 'Brad', 'Pitt', 'Edward', 'Norton' ]

},

{

\_id: ObjectId("63892fc6f086c3dc93796098"),

title: 'Inglorious Basterds',

writer: 'Quentin Tarantino',

year: 2009,

actors: [ 'Brad', 'Pitt', 'Diane', 'Kruger', 'Eli', 'Roth' ]

}

]

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

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

[

{

\_id: ObjectId("63892fc6f086c3dc93796099"),

title: 'The Hobbit: An Unexpected Journey',

writer: 'J.R.R. Tolkein',

year: 2012,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("63892fc6f086c3dc9379609a"),

title: 'The Hobbit: The Desolation of Smaug',

writer: 'J.R.R. Tolkein',

year: 2013,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("63892fc6f086c3dc9379609b"),

title: 'The Hobbit: The Battle of the Five Armies',

writer: 'J.R.R. Tolkein',

year: 2012,

franchise: 'The Hobbit'

}

]

5. get all movies released in the 90s

MongoDataBase> db.movies.find({$and:[{year:{$gte:1900}},{year:{$lt:2000}}]})

[

{

\_id: ObjectId("63892fc6f086c3dc93796096"),

title: 'Fight Club',

writer: 'Chuck Palahniuk',

year: 1999,

actors: [ 'Brad', 'Pitt', 'Edward', 'Norton' ]

},

{

\_id: ObjectId("63892fc6f086c3dc93796097"),

title: 'Pulp Fiction',

writer: 'Quentin Tarantino',

year: 1994,

actors: [ 'John', 'Travolta', 'Uma', 'Thurman' ]

}

]

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

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

[

{

\_id: ObjectId("63892fc6f086c3dc93796096"),

title: 'Fight Club',

writer: 'Chuck Palahniuk',

year: 1999,

actors: [ 'Brad', 'Pitt', 'Edward', 'Norton' ]

},

{

\_id: ObjectId("63892fc6f086c3dc93796097"),

title: 'Pulp Fiction',

writer: 'Quentin Tarantino',

year: 1994,

actors: [ 'John', 'Travolta', 'Uma', 'Thurman' ]

},

{

\_id: ObjectId("63892fc6f086c3dc93796099"),

title: 'The Hobbit: An Unexpected Journey',

writer: 'J.R.R. Tolkein',

year: 2012,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("63892fc6f086c3dc9379609a"),

title: 'The Hobbit: The Desolation of Smaug',

writer: 'J.R.R. Tolkein',

year: 2013,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("63892fc6f086c3dc9379609b"),

title: 'The Hobbit: The Battle of the Five Armies',

writer: 'J.R.R. Tolkein',

year: 2012,

franchise: 'The Hobbit'

}

]

7. count the number of writer from the movie collection

MongoDataBase> db.movies.count()

8. update an actor named "Samuel L. Jackson" to the movie "Pulp Fiction"

MongoDataBase> db.movies.updateOne({movie:"Pulp Fiction"},{$set:{actor:"Samuel L. Jackson"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 0,

modifiedCount: 0,

upsertedCount: 0

}

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

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

10. display the movie collection based on year in descending order.

MongoDataBase> db.movies.find().sort({year:-1})

[

{

\_id: ObjectId("63892fc6f086c3dc9379609a"),

title: 'The Hobbit: The Desolation of Smaug',

writer: 'J.R.R. Tolkein',

year: 2013,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("63892fc6f086c3dc93796099"),

title: 'The Hobbit: An Unexpected Journey',

writer: 'J.R.R. Tolkein',

year: 2012,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("63892fc6f086c3dc9379609b"),

title: 'The Hobbit: The Battle of the Five Armies',

writer: 'J.R.R. Tolkein',

year: 2012,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("63892fc6f086c3dc93796098"),

title: 'Inglorious Basterds',

writer: 'Quentin Tarantino',

year: 2009,

actors: [ 'Brad', 'Pitt', 'Diane', 'Kruger', 'Eli', 'Roth' ]

},

{

\_id: ObjectId("63892fc6f086c3dc93796096"),

title: 'Fight Club',

writer: 'Chuck Palahniuk',

year: 1999,

actors: [ 'Brad', 'Pitt', 'Edward', 'Norton' ]

},

{

\_id: ObjectId("63892fc6f086c3dc93796097"),

title: 'Pulp Fiction',

writer: 'Quentin Tarantino',

year: 1994,

actors: [ 'John', 'Travolta', 'Uma', 'Thurman' ]

}

]

11. Display last two recent movie

MongoDataBase> db.movies.find().limit(2).sort({$natural:-1})

[To exit, press Ctrl+C again or Ctrl+D or type .exit)

{

\_id: ObjectId("63892fc6f086c3dc9379609b"),

title: 'The Hobbit: The Battle of the Five Armies',

writer: 'J.R.R. Tolkein',

year: 2012,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("63892fc6f086c3dc9379609a"),

title: 'The Hobbit: The Desolation of Smaug',

writer: 'J.R.R. Tolkein',

year: 2013,

franchise: 'The Hobbit'

}

]