```
> show dbs
admin              0.000GB
config             0.000GB
local              0.000GB
mongoDB_assignment  0.000GB
> show collections
movies
> db.movies.find().pretty
function() {
    this._prettyShell = true;
    return this;
}
> db.movies.find().pretty()
{
    "_id" : ObjectId("617519949fbb93826b86a065"),
    "title" : "Fight Club",
    "writer" : "Chuck Palahniuko",
    "year" : "1999",
    "actors" : [
        "Brad pitt",
        "Edward Norton"
    ]
}
{
    "_id" : ObjectId("61751bc39fbb93826b86a066"),
    "title" : "Pulp Friction",
    "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"]})

WriteResult({ "nInserted" : 1 })

> db.movies.insert({title:"The Hobbit: An Unexpected Journey", writer:"J.R.R. Tolkein", year:"2012", franchise: "The Hobbit"})

WriteResult({ "nInserted" : 1 })

> db.movies.insert({title:"The Hobbit:The Desolation of Smug", writer:"J.R.R. Tolkein", year:"2013", franchise: "The Hobbit"})

WriteResult({ "nInserted" : 1 })

> db.movies.insert({title:"The Hobbit:The Battle of the Five Armies", writer:"J.R.R. Tolkein", year:"2013", franchise: "The Hobbit",syposis:"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"})

WriteResult({ "nInserted" : 1 })

> db.movies.insert({title:"Pee Wee Hermans's Big Adventure"})

WriteResult({ "nInserted" : 1 })

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

WriteResult({ "nInserted" : 1 })

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

{

    "_id" : ObjectId("617519949fbb93826b86a065"),

    "title" : "Fight Club",

    "writer" : "Chuck Palahniuko",

    "year" : "1999",

    "actors" : [

        "Brad pitt",

        "Edward Norton"

    ]

}

{

```json
        "_id" : ObjectId("61751bc39fbb93826b86a066"),

        "title" : "Pulp Friction",

        "writer" : "Quentin Tarantino",

        "year" : "1994",

        "actors" : [

                "John Travolta",

                "Uma Thurman"

        ]

}

{

        "_id" : ObjectId("61751f799fbb93826b86a067"),

        "title" : "Inglorious Basterds",

        "writer" : "Quentin Tarantino",

        "year" : "2009",

        "actors" : [

                "Brad Pitt",

                "Diane Kruger",

                "Eli Roth"

        ]

}

{

        "_id" : ObjectId("61752abc9fbb93826b86a068"),

        "title" : "The Hobbit: An Unexpected Journey",

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

        "year" : "2012",

        "franchise" : "The Hobbit"

}

{
```

```
    "_id" : ObjectId("61752b2f9fbb93826b86a069"),

        "title" : "The Hobbit:The Desolation of Smug",

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

        "year" : "2013",

        "franchise" : "The Hobbit"

}

{

        "_id" : ObjectId("61752bf59fbb93826b86a06a"),

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

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

        "year" : "2013",

        "franchise" : "The Hobbit",

        "syposis" : "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"

}

{

        "_id" : ObjectId("61752c6f9fbb93826b86a06b"),

        "title" : "Pee Wee Hermans's Big Adventure"

}

{ "_id" : ObjectId("61752c8d9fbb93826b86a06c"), "title" : "Avatar" }

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

uncaught exception: SyntaxError: "" literal not terminated before end of script :

@(shell):1:43

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

{ "_id" : ObjectId("61751bc39fbb93826b86a066"), "title" : "Pulp Friction", "writer" : "Quentin Tarantino", "year" : "1994",
"actors" : [ "John Travolta", "Uma Thurman" ] }

{ "_id" : ObjectId("61751f799fbb93826b86a067"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" :
"2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

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

{ "_id" : ObjectId("61751f799fbb93826b86a067"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : ["Brad Pitt", "Diane Kruger", "Eli Roth"] }

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

{ "_id" : ObjectId("61752abc9fbb93826b86a068"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit" }

{ "_id" : ObjectId("61752b2f9fbb93826b86a069"), "title" : "The Hobbit:The Desolation of Smug", "writer" : "J.R.R. Tolkein", "year" : "2013", "franchise" : "The Hobbit" }

{ "_id" : ObjectId("61752bf59fbb93826b86a06a"), "title" : "The Hobbit:The Battle of the Five Armies", "writer" : "J.R.R. Tolkein", "year" : "2013", "franchise" : "The Hobbit", "syposis" : "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.find({year:{$gt:"1990", $lt:"2000"}})

{ "_id" : ObjectId("617519949fbb93826b86a065"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : "1999", "actors" : ["Brad pitt", "Edward Norton"] }

{ "_id" : ObjectId("61751bc39fbb93826b86a066"), "title" : "Pulp Friction", "writer" : "Quentin Tarantino", "year" : "1994", "actors" : ["John Travolta", "Uma Thurman"] }

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

{ "_id" : ObjectId("617519949fbb93826b86a065"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : "1999", "actors" : ["Brad pitt", "Edward Norton"] }

{ "_id" : ObjectId("61751bc39fbb93826b86a066"), "title" : "Pulp Friction", "writer" : "Quentin Tarantino", "year" : "1994", "actors" : ["John Travolta", "Uma Thurman"] }

{ "_id" : ObjectId("61752abc9fbb93826b86a068"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit" }

{ "_id" : ObjectId("61752b2f9fbb93826b86a069"), "title" : "The Hobbit:The Desolation of Smug", "writer" : "J.R.R. Tolkein", "year" : "2013", "franchise" : "The Hobbit" }

{ "_id" : ObjectId("61752bf59fbb93826b86a06a"), "title" : "The Hobbit:The Battle of the Five Armies", "writer" : "J.R.R. Tolkein", "year" : "2013", "franchise" : "The Hobbit", "syposis" : "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.update({title: "The Hobbit: An Unexpected Journey"},{$set:{synopsis:"The dwarves, along with Bibo 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"}})

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

> 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."}})

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

```
> db.movies.update({title: "Pulp Fiction"}, {$set:{actors: "Samuel Jackson"}})

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

> db.movies.update({title: "Pulp Friction"}, {$set:{actors: "Samuel Jackson"}})

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

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

{

    "_id" : ObjectId("617519949fbb93826b86a065"),

    "title" : "Fight Club",

    "writer" : "Chuck Palahniuko",

    "year" : "1999",

    "actors" : [

         "Brad pitt",

         "Edward Norton"

    ]

}

{

    "_id" : ObjectId("61751bc39fbb93826b86a066"),

    "title" : "Pulp Friction",

    "writer" : "Quentin Tarantino",

    "year" : "1994",

    "actors" : "Samuel Jackson"

}

{

    "_id" : ObjectId("61751f799fbb93826b86a067"),

    "title" : "Inglorious Basterds",

    "writer" : "Quentin Tarantino",

    "year" : "2009",
```

    "actors" : [

            "Brad Pitt",

            "Diane Kruger",

            "Eli Roth"

    ]

}

{

    "_id" : ObjectId("61752abc9fbb93826b86a068"),

    "title" : "The Hobbit: An Unexpected Journey",

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

    "year" : "2012",

    "franchise" : "The Hobbit",

    "synopsis" : "The dwarves, along with Bibo 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"

}

{

    "_id" : ObjectId("61752b2f9fbb93826b86a069"),

    "title" : "The Hobbit:The Desolation of Smug",

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

    "year" : "2013",

    "franchise" : "The Hobbit"

}

{

    "_id" : ObjectId("61752bf59fbb93826b86a06a"),

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

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

    "year" : "2013",

    "franchise" : "The Hobbit",

"syposis" : "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"

}

{

    "_id" : ObjectId("61752c6f9fbb93826b86a06b"),

    "title" : "Pee Wee Hermans's Big Adventure"

}

{ "_id" : ObjectId("61752c8d9fbb93826b86a06c"), "title" : "Avatar" }

> db.movies.createIndex({ title:"text", writer:"text", year:"text",actors:"text", synopsis:"text",franchise:"text"})

{

    "numIndexesBefore" : 1,

    "numIndexesAfter" : 2,

    "createdCollectionAutomatically" : false,

    "ok" : 1

}

> db.movies.find({ $text: {$search: "Bilbo"}})

{ "_id" : ObjectId("6173fd0579e9389f5c358e78"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R. Tolkein", "year" : "2013", "franchise" : "The Hobbit", "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." }

{ "_id" : ObjectId("6173febf79e9389f5c358e79"), "title" : "The Hobbit: The Battle of the Five Armies", "writer" : "J.R.R. Tolkein", "year" : "2013", "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" }

{ "_id" : ObjectId("6173fcd779e9389f5c358e77"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit", "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" }

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

{ "_id" : ObjectId("6173fd0579e9389f5c358e78"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R. Tolkein", "year" : "2013", "franchise" : "The Hobbit", "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." }

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

{ "_id" : ObjectId("6173febf79e9389f5c358e79"), "title" : "The Hobbit: The Battle of the Five Armies", "writer" : "J.R.R. Tolkein", "year" : "2013", "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" }

{ "_id" : ObjectId("6173fcd779e9389f5c358e77"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit", "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" }

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

{ "_id" : ObjectId("6173fcd779e9389f5c358e77"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit", "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" }

{ "_id" : ObjectId("6173fd0579e9389f5c358e78"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R. Tolkein", "year" : "2013", "franchise" : "The Hobbit", "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." }

{ "_id" : ObjectId("6173febf79e9389f5c358e79"), "title" : "The Hobbit: The Battle of the Five Armies", "writer" : "J.R.R. Tolkein", "year" : "2013", "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.find({ "$text": {"$search":"gold dragon" }})

{ "_id" : ObjectId("6173fcd779e9389f5c358e77"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit", "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" }

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

WriteResult({ "nRemoved" : 1 })

> db.movies.remove({title : "Avatar"})

WriteResult({ "nRemoved" : 1 })

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

{

    "_id" : ObjectId("6173fb4f79e9389f5c358e74"),

    "title" : "Fight Club",

    "writer" : "Chuck Palahniuko",

    "year" : "1999",

    "actors" : [

        "Brad Pitt",

        "Edward Norton"

    ]

}

{

        "_id" : ObjectId("6173fbc379e9389f5c358e75"),

        "title" : "Pulp Fiction",

        "writer" : "Quentin Tarantino",

        "year" : "1994",

        "actors" : "Samuel L. Jackson"

}

{

        "_id" : ObjectId("6173fc5a79e9389f5c358e76"),

        "title" : "Inglorious Basterds",

        "writer" : "Quentin Tarantino",

        "year" : "2009",

        "actors" : [

                "Brad Pitt",

                "Diane Kruger",

                "Eli Roth"

        ]

}

{

        "_id" : ObjectId("6173fcd779e9389f5c358e77"),

        "title" : "The Hobbit: An Unexpected Journey",

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

        "year" : "2012",

        "franchise" : "The Hobbit",

        "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"

}

{

```
"_id" : ObjectId("6173fd0579e9389f5c358e78"),

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

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

      "year" : "2013",

      "franchise" : "The Hobbit",

      "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."

}

{

      "_id" : ObjectId("6173febf79e9389f5c358e79"),

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

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

      "year" : "2013",

      "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.createCollection("users")

{ "ok" : 1 }

> db.users.insert([{username:"GoodGuyGreg", first_name:"Good Guy", last_name: "Greg"}, {username: "ScumbagSteve", full_name: { first: "Scumbag", last: "Steve" }}])

BulkWriteResult({

      "writeErrors" : [ ],

      "writeConcernErrors" : [ ],

      "nInserted" : 2,

      "nUpserted" : 0,

      "nMatched" : 0,

      "nModified" : 0,

      "nRemoved" : 0,

      "upserted" : [ ]

})
```

```
> db.posts.insert([{_id: 1, username:"GoodGuyGreg", title: "Passes out at party", body: "Wakes up early and cleans house"},
{_id: 2, username : "GoodGuyGreg", title: "Steals your identity", body : "Raises your credit score"},{_id: 4, username:
"ScumbagSteve", title : "Borrows something", body: "Sells it"},

... {_id:5, username: "ScumbagSteve", title:"Borrows everything", body: "The end"},{ _id:6, username: "ScumbagSteve", title:
"Forks your repo on github", body:"Sets to private"}])

BulkWriteResult({

        "writeErrors" : [ ],

        "writeConcernErrors" : [ ],

        "nInserted" : 5,

        "nUpserted" : 0,

        "nMatched" : 0,

        "nModified" : 0,

        "nRemoved" : 0,

        "upserted" : [ ]

})

> db.comments.insert([{ username:"GoodGuyGreg", comment: "Hope you got a good deal!", post: 4},
{username:"GoodGuyGreg",comment: "What's mine is yours!", post: 5},{ username:"GoodGuyGreg", comment:"Don't violate
the licensing agreement!", post: 6}, { username:"ScumbagSteve", comment:"It still isn't clean",
post:1},{username:"ScumbagSteve", comment:"Denied yout PR cause I found a hack", post:3}])

BulkWriteResult({

        "writeErrors" : [ ],

        "writeConcernErrors" : [ ],

        "nInserted" : 5,

        "nUpserted" : 0,

        "nMatched" : 0,

        "nModified" : 0,

        "nRemoved" : 0,

        "upserted" : [ ]

})

> db.users.find().pretty()

{
```

```
    "_id" : ObjectId("6174f171145a418ce6b01a50"),

    "username" : "GoodGuyGreg",

    "first_name" : "Good Guy",

    "last_name" : "Greg"

}

{

    "_id" : ObjectId("6174f171145a418ce6b01a51"),

    "username" : "ScumbagSteve",

    "full_name" : {

        "first" : "Scumbag",

        "last" : "Steve"

    }}

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

{

    "_id" : 1,

    "username" : "GoodGuyGreg",

    "title" : "Passes out at party",

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

}

{

    "_id" : 2,

    "username" : "GoodGuyGreg",

    "title" : "Steals your identity",

    "body" : "Raises your credit score"

}

{

    "_id" : 4,

    "username" : "ScumbagSteve",

    "title" : "Borrows something",

    "body" : "Sells it"

}
```

```
{

    "_id" : 5,

    "username" : "ScumbagSteve",

    "title" : "Borrows everything",

    "body" : "The end"

}

{

    "_id" : 6,

    "username" : "ScumbagSteve",

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

    "body" : "Sets to private"

}

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

{

    "_id" : ObjectId("6174f498145a418ce6b01a52"),

    "username" : "GoodGuyGreg",

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

    "post" : 4

}

{

    "_id" : ObjectId("6174f498145a418ce6b01a53"),

    "username" : "GoodGuyGreg",

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

    "post" : 5

}

{

    "_id" : ObjectId("6174f498145a418ce6b01a54"),

    "username" : "GoodGuyGreg",

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

    "post" : 6

}
```

```
{

    "_id" : ObjectId("6174f498145a418ce6b01a55"),

    "username" : "ScumbagSteve",

    "comment" : "It still isn't clean",

    "post" : 1

}

{

    "_id" : ObjectId("6174f498145a418ce6b01a56"),

    "username" : "ScumbagSteve",

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

    "post" : 3

}

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

{

    "_id" : 1,

    "username" : "GoodGuyGreg",

    "title" : "Passes out at party",

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

}

{

    "_id" : 2,

    "username" : "GoodGuyGreg",

    "title" : "Steals your identity",

    "body" : "Raises your credit score"

}

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

{

    "_id" : 4,

    "username" : "ScumbagSteve",
```

```
        "title" : "Borrows something",

        "body" : "Sells it"

}

{

        "_id" : 5,

        "username" : "ScumbagSteve",

        "title" : "Borrows everything",

        "body" : "The end"

}

{

        "_id" : 6,

        "username" : "ScumbagSteve",

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

        "body" : "Sets to private"

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

{

        "_id" : ObjectId("6174f498145a418ce6b01a52"),

        "username" : "GoodGuyGreg",

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

        "post" : 4

}

{

        "_id" : ObjectId("6174f498145a418ce6b01a53"),

        "username" : "GoodGuyGreg",

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

        "post" : 5

}
```

```
{
    "_id" : ObjectId("6174f498145a418ce6b01a54"),
    "username" : "GoodGuyGreg",
    "comment" : "Don't violate the licensing agreement!",
    "post" : 6
}
> db.comments.find({username: "ScumbagSteve"}).pretty()
{
    "_id" : ObjectId("6174f498145a418ce6b01a55"),
    "username" : "ScumbagSteve",
    "comment" : "It still isn't clean",
    "post" : 1
}
{
    "_id" : ObjectId("6174f498145a418ce6b01a56"),
    "username" : "ScumbagSteve",
    "comment" : "Denied yout PR cause I found a hack",
    "post" : 3
}

>
```