**Insert Document:**

> use mongo\_practice

switched to db mongo\_practice

> db.movies.insertMany([

... {title: 'Fight Club',

... writer: 'Chuck Palahniuko',

... year: 1999,

... actors: ['Brad Pitt','Ekward 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',

... 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'},

... {title: "Pee Wee Herman's Big Adventure",

... {title: 'Avtar'}

... ])

{

"acknowledged" : true,

"insertedIds" : [

ObjectId("6107904ab5347c52cee0e502"),

ObjectId("6107904ab5347c52cee0e503"),

ObjectId("6107904ab5347c52cee0e504"),

ObjectId("6107904ab5347c52cee0e505"),

ObjectId("6107904ab5347c52cee0e506"),

ObjectId("6107904ab5347c52cee0e507"),

ObjectId("6107904ab5347c52cee0e508"),

ObjectId("6107904ab5347c52cee0e509")

]

}

**Find documents**

**1)**

> db.movies.find()

{ "\_id" : ObjectId("6107904ab5347c52cee0e502"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : 1999, "actors" : [ "Brad Pitt", "Ekward Norton" ] }

{ "\_id" : ObjectId("6107904ab5347c52cee0e503"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : 1994, "actors" : [ "John Travolta", "Uma Thurman" ] }

{ "\_id" : ObjectId("6107904ab5347c52cee0e504"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : 2009, "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

{ "\_id" : ObjectId("6107904ab5347c52cee0e505"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("6107904ab5347c52cee0e506"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R. Tolkein", "Year" : 2013, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("6107904ab5347c52cee0e507"), "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" }

{ "\_id" : ObjectId("6107904ab5347c52cee0e508"), "title" : "Pee Wee Herman's Big Adventure" }

{ "\_id" : ObjectId("6107904ab5347c52cee0e509"), "title" : "Avtar" }

**2)**

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

{ "\_id" : ObjectId("6107904ab5347c52cee0e503"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : 1994, "actors" : [ "John Travolta", "Uma Thurman" ] }

{ "\_id" : ObjectId("6107904ab5347c52cee0e504"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : 2009, "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

**3)**

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

{ "\_id" : ObjectId("6107904ab5347c52cee0e502"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : 1999, "actors" : [ "Brad Pitt", "Ekward Norton" ] }

{ "\_id" : ObjectId("6107904ab5347c52cee0e504"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : 2009, "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

**4)**

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

{ "\_id" : ObjectId("6107904ab5347c52cee0e505"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("6107904ab5347c52cee0e506"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R. Tolkein", "Year" : 2013, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("6107904ab5347c52cee0e507"), "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" }

**5)**

> db.movies.find({year:{$gte: 1990,$lte: 1999}})

{ "\_id" : ObjectId("6107904ab5347c52cee0e502"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : 1999, "actors" : [ "Brad Pitt", "Ekward Norton" ] }

{ "\_id" : ObjectId("6107904ab5347c52cee0e503"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : 1994, "actors" : [ "John Travolta", "Uma Thurman" ] }

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

{ "\_id" : ObjectId("6107904ab5347c52cee0e502"), "title" : "Fight Club", "writer" : "Chuck Palahniuko", "year" : 1999, "actors" : [ "Brad Pitt", "Ekward Norton" ] }

{ "\_id" : ObjectId("6107904ab5347c52cee0e503"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : 1994, "actors" : [ "John Travolta", "Uma Thurman" ] }

{ "\_id" : ObjectId("6107904ab5347c52cee0e505"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : 2012, "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("6107904ab5347c52cee0e507"), "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" }

**6)**

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

{

"\_id" : ObjectId("6107904ab5347c52cee0e502"),

"title" : "Fight Club",

"writer" : "Chuck Palahniuko",

"year" : 1999,

"actors" : [

"Brad Pitt",

"Ekward Norton"

]

}

{

"\_id" : ObjectId("6107904ab5347c52cee0e503"),

"title" : "Pulp Fiction",

"writer" : "Quentin Tarantino",

"year" : 1994,

"actors" : [

"John Travolta",

"Uma Thurman"

]

}

{

"\_id" : ObjectId("6107904ab5347c52cee0e505"),

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

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

"year" : 2012,

"franchise" : "The Hobbit"

}

{

"\_id" : ObjectId("6107904ab5347c52cee0e507"),

"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"

}

**Update Document:**

> db.movies.update({\_id : ObjectId("6107904ab5347c52cee0e503")},{$set:{actors:["Samuel L. Jackson","John Travolta","Uma Thurman"]}})

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

> db.movies.update({\_id:ObjectId("6107904ab5347c52cee0e506")},{$set:{synopsis:"The dwarves, along with Biblo Baggins and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug"}})

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

> db.movies.update({\_id:ObjectId("6107904ab5347c52cee0e505")},{$set:{synopsis:" A reluctant hobbit, Biblo Baggins, sets out to the Lonely Mountain with a spiritedgroup of dwarves to reclaim their mountain home - and the gold with it - from the dragon smaug"}})

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

**Delete Documents:**

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

WriteResult({ "nRemoved" : 1 })

> db.movies.remove({title:'Avtar'})

WriteResult({ "nRemoved" : 1 })

**Relationships:**

**Insert Users Collection**

> db.users.insertMany([

... {username: 'GoodGuyGreg',

... first\_name: 'Good Guy',

... last\_name: 'Greg'},

... {username: 'ScumbagSteve',

... full\_name: [{first: 'Scumbag'}, {last: 'Steve'}]}

... ])

{

"acknowledged" : true,

"insertedIds" : [

ObjectId("6108a24f7f9641831d88974f"),

ObjectId("6108a24f7f9641831d889750")

]

}

**Insert Posts Collection**

> 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: 'Fork your repo on github',

... body: 'Sets to private'}

... ])

{

"acknowledged" : true,

"insertedIds" : [

ObjectId("6108a3ad7f9641831d889751"),

ObjectId("6108a3ad7f9641831d889752"),

ObjectId("6108a3ad7f9641831d889753"),

ObjectId("6108a3ad7f9641831d889754"),

ObjectId("6108a3ad7f9641831d889755"),

ObjectId("6108a3ad7f9641831d889756")

]

}

**Insert comments collection**

> db.comments.insertMany([

... {username: 'GoodGuyGreg',

... comment: 'Hope you got a good deal!',

... post: ObjectId("610677ee1defc01e7ba96951")},

... {username: 'GoodGuyGreg',

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

... post: ObjectId("610677ee1defc01e7ba96952")},

... {username: 'GoodGuyGreg',

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

... post: ObjectId("610677ee1defc01e7ba96953")},

... {username: 'ScumbagSteve',

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

... post: ObjectId("610677ee1defc01e7ba9694e")},

... {username: 'ScumbagSteve',

... comment: 'Denied your PR cause I found a hack',

... post: ObjectId("610677ee1defc01e7ba96950")}

... ])

{

"acknowledged" : true,

"insertedIds" : [

ObjectId("6108a4c37f9641831d889757"),

ObjectId("6108a4c37f9641831d889758"),

ObjectId("6108a4c37f9641831d889759"),

ObjectId("6108a4c37f9641831d88975a"),

ObjectId("6108a4c37f9641831d88975b")

]

}

**Querying Related Collection**

**1)**

{

“\_id” : ObjectId(“6108a24f7f9641831d88974f”),

“username” : “GoodGuyGreg”,

“first\_name” : “Good Guy”,

“last\_name” : “Greg”

}

{

“\_id” : ObjectId(“6108a24f7f9641831d889750”),

“username” : “ScumbagSteve”,

“full\_name” : [

{

“first” : “Scumbag”

},

{

“last” : “Steve”

}

]

}

**2)**

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

{

"\_id" : ObjectId("6108a3ad7f9641831d889751"),

"username" : "GoodGuyGreg",

"title" : "Passes out at party",

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

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889752"),

"username" : "GoodGuyGreg",

"title" : "Steals your identity",

"body" : "Raises your credit score"

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889753"),

"username" : "GoodGuyGreg",

"title" : "Reports a bug in your code",

"body" : "Sends you a Pull Request"

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889754"),

"username" : "ScumbagSteve",

"title" : "Borrows something",

"body" : "Sells it"

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889755"),

"username" : "ScumbagSteve",

"title" : "Borrows everything",

"body" : "The end"

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889756"),

"username" : "ScumbagSteve",

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

"body" : "Sets to private"

}

**3)**

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

{

"\_id" : ObjectId("6108a3ad7f9641831d889751"),

"username" : "GoodGuyGreg",

"title" : "Passes out at party",

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

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889752"),

"username" : "GoodGuyGreg",

"title" : "Steals your identity",

"body" : "Raises your credit score"

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889753"),

"username" : "GoodGuyGreg",

"title" : "Reports a bug in your code",

"body" : "Sends you a Pull Request"

}

**4)**

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

{

"\_id" : ObjectId("6108a3ad7f9641831d889754"),

"username" : "ScumbagSteve",

"title" : "Borrows something",

"body" : "Sells it"

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889755"),

"username" : "ScumbagSteve",

"title" : "Borrows everything",

"body" : "The end"

}

{

"\_id" : ObjectId("6108a3ad7f9641831d889756"),

"username" : "ScumbagSteve",

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

"body" : "Sets to private"

}

**5)**

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

{

"\_id" : ObjectId("6108a4c37f9641831d889757"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("610677ee1defc01e7ba96951")

}

{

"\_id" : ObjectId("6108a4c37f9641831d889758"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("610677ee1defc01e7ba96952")

}

{

"\_id" : ObjectId("6108a4c37f9641831d889759"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("610677ee1defc01e7ba96953")

}

{

"\_id" : ObjectId("6108a4c37f9641831d88975a"),

"username" : "ScumbagSteve",

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

"post" : ObjectId("610677ee1defc01e7ba9694e")

}

{

"\_id" : ObjectId("6108a4c37f9641831d88975b"),

"username" : "ScumbagSteve",

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

"post" : ObjectId("610677ee1defc01e7ba96950")

}

**6)**

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

{

"\_id" : ObjectId("6108a4c37f9641831d889757"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("610677ee1defc01e7ba96951")

}

{

"\_id" : ObjectId("6108a4c37f9641831d889758"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("610677ee1defc01e7ba96952")

}

{

"\_id" : ObjectId("6108a4c37f9641831d889759"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("610677ee1defc01e7ba96953")

}

**7)**

{

"\_id" : ObjectId("6108a4c37f9641831d88975a"),

"username" : "ScumbagSteve",

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

"post" : ObjectId("610677ee1defc01e7ba9694e")

}

{

"\_id" : ObjectId("6108a4c37f9641831d88975b"),

"username" : "ScumbagSteve",

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

"post" : ObjectId("610677ee1defc01e7ba96950")

}

**8)**

> db.posts.find({title: 'Reports a bug in your code'}).pretty()

{

"\_id" : ObjectId("6108a3ad7f9641831d889753"),

"username" : "GoodGuyGreg",

"title" : "Reports a bug in your code",

"body" : "Sends you a Pull Request"

}