C:\Users\mkart\Downloads\mongodb-win32-x86\_64-windows-4.4.6\bin>mongo "mongodb+srv://mongodb.kkwgc.mongodb.net/myFirstDatabase" --username karthik

MongoDB shell version v4.4.6

Enter password:

connecting to: mongodb://mongodb-shard-00-00.kkwgc.mongodb.net:27017,mongodb-shard-00-01.kkwgc.mongodb.net:27017,mongodb-shard-00-02.kkwgc.mongodb.net:27017/myFirstDatabase?authSource=admin&compressors=disabled&gssapiServiceName=mongodb&replicaSet=atlas-z8cl0o-shard-0&ssl=true

Implicit session: session { "id" : UUID("9059d426-5351-4168-b187-194b6292eea4") }

MongoDB server version: 4.4.6

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> show dbs

admin 0.000GB

local 0.926GB

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> use mongo\_practice

switched to db mongo\_practice

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.insert({title:"Fight Club", writer: "Chuck Palahniuk", year: "1999", actors:["Brad Pitt", "Edward Norton"]})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.insert({title:"Pulp Fiction", writer:"Quentin Tarantino", year:"2009", actors:["John Travolta", "Uma Thurman"]})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.insert({title:"Inglorious Basterds", writer:"Quentin Tarantino", year:"2009", actors:["Brad Pitt", "Diane Kruger", "Eli Roth"]})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY>

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.insert({title:"The Hobbit: An unexpected Journey", writer:"J.R.R. Tolkein", year:"2012",franchise:"The Hobbit"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.insert({title:"The Hobbit: The Desolation of Smaug", writer:"J.R.R Tolkien", year:"2013", franchise:"The Hobbit"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> 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."})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.insert({title:"Pee Wee Herman's Big Adventures"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.insert({title:"Avatar"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find()

{ "\_id" : ObjectId("60e6d25fe6e758b1d43cf840"), "title" : "Fight Club", "writer" : "Chuck Palahniuk", "year" : "1999", "actors" : [ "Brad Pitt", "Edward Norton" ] }

{ "\_id" : ObjectId("60e6d271e6e758b1d43cf841"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "John Travolta", "Uma Thurman" ] }

{ "\_id" : ObjectId("60e6d27ae6e758b1d43cf842"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

{ "\_id" : ObjectId("60e6d288e6e758b1d43cf843"), "title" : "The Hobbit: An unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e6d291e6e758b1d43cf844"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "year" : "2013", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e6d2a2e6e758b1d43cf845"), "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." }

{ "\_id" : ObjectId("60e6d2afe6e758b1d43cf846"), "title" : "Pee Wee Herman's Big Adventures" }

{ "\_id" : ObjectId("60e6d2b9e6e758b1d43cf847"), "title" : "Avatar" }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({writer:"Quentin Tarantino"})

{ "\_id" : ObjectId("60e6d271e6e758b1d43cf841"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "John Travolta", "Uma Thurman" ] }

{ "\_id" : ObjectId("60e6d27ae6e758b1d43cf842"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({actors:"Brad Pitt"})

{ "\_id" : ObjectId("60e6d25fe6e758b1d43cf840"), "title" : "Fight Club", "writer" : "Chuck Palahniuk", "year" : "1999", "actors" : [ "Brad Pitt", "Edward Norton" ] }

{ "\_id" : ObjectId("60e6d27ae6e758b1d43cf842"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({franchise:"The Hobbit"})

{ "\_id" : ObjectId("60e6d288e6e758b1d43cf843"), "title" : "The Hobbit: An unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e6d291e6e758b1d43cf844"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "year" : "2013", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e6d2a2e6e758b1d43cf845"), "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." }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({year:{$gt:"1990", $lt:"2000"}})

{ "\_id" : ObjectId("60e6d25fe6e758b1d43cf840"), "title" : "Fight Club", "writer" : "Chuck Palahniuk", "year" : "1999", "actors" : [ "Brad Pitt", "Edward Norton" ] }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({$or:[{year:{$gt:"2010"}},{year: {$lt:"2000"}}]})

{ "\_id" : ObjectId("60e6d25fe6e758b1d43cf840"), "title" : "Fight Club", "writer" : "Chuck Palahniuk", "year" : "1999", "actors" : [ "Brad Pitt", "Edward Norton" ] }

{ "\_id" : ObjectId("60e6d288e6e758b1d43cf843"), "title" : "The Hobbit: An unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e6d291e6e758b1d43cf844"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "year" : "2013", "franchise" : "The Hobbit" }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> 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."}})

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

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> 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."}})

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

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.update({\_id:ObjectId("5c9f983ce5c2dfe9b3729bfc")}, {$push:{actors:"Samuel L. Jackson"}})

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

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({synopsis:{$regex:"Bilbo"}})

{ "\_id" : ObjectId("60e6d2a2e6e758b1d43cf845"), "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." }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({synopsis:{$regex:"Gandalf"}})

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({$and:[{synopsis:{$regex:"Bilbo"}}, {synopsis:{$not:/Gandalf/}}]})

{ "\_id" : ObjectId("60e6d2a2e6e758b1d43cf845"), "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." }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({$or:[{synopsis:{$regex:"dwarves"}}, {synopsis:{$regex:"hobbit"}}]})

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY>

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({$or:[{synopsis:{$regex:"dwarves"}}, {synopsis:{$regex:"hobbit"}}]})

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.find({$and:[{synopsis:{$regex:"gold"}}, {synopsis:{$regex:"dragon"}}]})

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.remove({\_id:ObjectId("5c9f992ae5c2dfe9b3729c00")})

WriteResult({ "nRemoved" : 0 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.movies.remove({\_id:ObjectId("5c9f9936e5c2dfe9b3729c01")})

WriteResult({ "nRemoved" : 0 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.users.insert({\_id:1,username:"GoodGuyGreg", first\_name:"Good Guy", last\_name:"Greg"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.users.insert({\_id:2, username:"ScumbagSteve", fullname:{first: "Scumbag", last:"Steve"}})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.insert({username:"GoodGuyGreg", title:"Passes out at Party", body:"Raises your credit score"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.insert({ username:"GoodGuyGreg", title:"Steals your identity", body:"Raises your credit score"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.insert({username:"GoodGuyGreg", title:"Reports a bug in your code", body:"Sends you a pull request"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.insert({ username:"ScumbagSteve", title:"Borrows something", body:"Sells it"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.insert({ username:"ScumbagSteve", title:"Borrows everything", body:"The end"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY>

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.insert({username:"ScumbagSteve", title:"Forks your repo on github", body:"Sets to private"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.comments.insert({ username:"GoodGuyGreg", comment:"Hope you got a good deal!", post:ObjectId("5ca0b7e96435f98b5901f463")})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.comments.insert({username:"GoodGuyGreg", comment:"What's mine is yours!", post:ObjectId("5ca0b9706435f98b5901f46a")})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.comments.insert({username:"GoodGuyGreg", comment:"Don't violate the licensing agreement!", post:ObjectId("5ca0b8766435f98b5901f467")})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.comments.insert({username:"ScumbagSteve", comment:"It still isn't clean", post:ObjectId("5ca0b8546435f98b5901f466")})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.comments.insert({username:"ScumbagSteve", comment:"Denied your PR cause I found a hack", post:ObjectId("5ca0b9256435f98b5901f469")})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.users.find().pretty()

{

"\_id" : 1,

"username" : "GoodGuyGreg",

"first\_name" : "Good Guy",

"last\_name" : "Greg"

}

{

"\_id" : 2,

"username" : "ScumbagSteve",

"fullname" : {

"first" : "Scumbag",

"last" : "Steve"

}

}

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.find().pretty()

{

"\_id" : ObjectId("60e6d370e6e758b1d43cf848"),

"username" : "GoodGuyGreg",

"title" : "Passes out at Party",

"body" : "Raises your credit score"

}

{

"\_id" : ObjectId("60e6d37ae6e758b1d43cf849"),

"username" : "GoodGuyGreg",

"title" : "Steals your identity",

"body" : "Raises your credit score"

}

{

"\_id" : ObjectId("60e6d384e6e758b1d43cf84a"),

"username" : "GoodGuyGreg",

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

"body" : "Sends you a pull request"

}

{

"\_id" : ObjectId("60e6d398e6e758b1d43cf84b"),

"username" : "ScumbagSteve",

"title" : "Borrows something",

"body" : "Sells it"

}

{

"\_id" : ObjectId("60e6d3a2e6e758b1d43cf84c"),

"username" : "ScumbagSteve",

"title" : "Borrows everything",

"body" : "The end"

}

{

"\_id" : ObjectId("60e6d3abe6e758b1d43cf84d"),

"username" : "ScumbagSteve",

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

"body" : "Sets to private"

}

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.find({username:"GoodGuyGreg"})

{ "\_id" : ObjectId("60e6d370e6e758b1d43cf848"), "username" : "GoodGuyGreg", "title" : "Passes out at Party", "body" : "Raises your credit score" }

{ "\_id" : ObjectId("60e6d37ae6e758b1d43cf849"), "username" : "GoodGuyGreg", "title" : "Steals your identity", "body" : "Raises your credit score" }

{ "\_id" : ObjectId("60e6d384e6e758b1d43cf84a"), "username" : "GoodGuyGreg", "title" : "Reports a bug in your code", "body" : "Sends you a pull request" }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.posts.find({username:"ScumbagSteve"})

{ "\_id" : ObjectId("60e6d398e6e758b1d43cf84b"), "username" : "ScumbagSteve", "title" : "Borrows something", "body" : "Sells it" }

{ "\_id" : ObjectId("60e6d3a2e6e758b1d43cf84c"), "username" : "ScumbagSteve", "title" : "Borrows everything", "body" : "The end" }

{ "\_id" : ObjectId("60e6d3abe6e758b1d43cf84d"), "username" : "ScumbagSteve", "title" : "Forks your repo on github", "body" : "Sets to private" }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.comments.find().pretty()

{

"\_id" : ObjectId("60e6d3b7e6e758b1d43cf84e"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("5ca0b7e96435f98b5901f463")

}

{

"\_id" : ObjectId("60e6d3c4e6e758b1d43cf84f"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("5ca0b9706435f98b5901f46a")

}

{

"\_id" : ObjectId("60e6d3cee6e758b1d43cf850"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("5ca0b8766435f98b5901f467")

}

{

"\_id" : ObjectId("60e6d3d7e6e758b1d43cf851"),

"username" : "ScumbagSteve",

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

"post" : ObjectId("5ca0b8546435f98b5901f466")

}

{

"\_id" : ObjectId("60e6d3e0e6e758b1d43cf852"),

"username" : "ScumbagSteve",

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

"post" : ObjectId("5ca0b9256435f98b5901f469")

}

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.comments.find({username:"GoodGuyGreg"})

{ "\_id" : ObjectId("60e6d3b7e6e758b1d43cf84e"), "username" : "GoodGuyGreg", "comment" : "Hope you got a good deal!", "post" : ObjectId("5ca0b7e96435f98b5901f463") }

{ "\_id" : ObjectId("60e6d3c4e6e758b1d43cf84f"), "username" : "GoodGuyGreg", "comment" : "What's mine is yours!", "post" : ObjectId("5ca0b9706435f98b5901f46a") }

{ "\_id" : ObjectId("60e6d3cee6e758b1d43cf850"), "username" : "GoodGuyGreg", "comment" : "Don't violate the licensing agreement!", "post" : ObjectId("5ca0b8766435f98b5901f467") }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY> db.comments.find({username:"ScumbagSteve"})

{ "\_id" : ObjectId("60e6d3d7e6e758b1d43cf851"), "username" : "ScumbagSteve", "comment" : "It still isn't clean", "post" : ObjectId("5ca0b8546435f98b5901f466") }

{ "\_id" : ObjectId("60e6d3e0e6e758b1d43cf852"), "username" : "ScumbagSteve", "comment" : "Denied your PR cause I found a hack", "post" : ObjectId("5ca0b9256435f98b5901f469") }

MongoDB Enterprise atlas-z8cl0o-shard-0:PRIMARY>