**INSERT DOCUMENTS**

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> use mongo\_practice

switched to db mongo\_practice

MongoDB Enterprise atlas-5d45wr-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-5d45wr-shard-0:PRIMARY> db.movies.insert({title:"Pulp Fiction", writer:"Quentin Tarantino", year:"1994", actors:["John Travolta", "Uma Thurman"]})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-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-5d45wr-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-5d45wr-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-5d45wr-shard-0:PRIMARY> db.movies.insert({title:"The Hobbit: The Battle of the Five Armies", writer:"J.R.R Tolkien", 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."})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.insert({title:"Pee Wee Herman's Big Adventures"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.insert({title:"Avatar"})

WriteResult({ "nInserted" : 1 })

**Query / Find Documents**

1.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find()

{ "\_id" : ObjectId("60e5ea9aade15b7261a741b7"), "title" : "Fight Club", "writer" : "Chuck Palahniuk", "year" : "1999", "actors" : [ "Brad Pitt", "Edward Norton" ] }

{ "\_id" : ObjectId("60e5eab8ade15b7261a741b8"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : "1994", "actors" : [ "John Travolta", "Uma Thurman" ] }

{ "\_id" : ObjectId("60e5eac5ade15b7261a741b9"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

{ "\_id" : ObjectId("60e5ead5ade15b7261a741ba"), "title" : "The Hobbit: An unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e5eae1ade15b7261a741bb"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "year" : "2013", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e5eaefade15b7261a741bc"), "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("60e5eb4bade15b7261a741bd"), "title" : "Pee Wee Herman's Big Adventures" }

{ "\_id" : ObjectId("60e5eb5bade15b7261a741be"), "title" : "Avatar" }

2.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY>db.movies.find({writer:"Quentin Tarantino"})

{ "\_id" : ObjectId("60e5eab8ade15b7261a741b8"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "John Travolta", "Uma Thurman" ] }

{ "\_id" : ObjectId("60e5eac5ade15b7261a741b9"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

3.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({actors:"Brad Pitt"})

{ "\_id" : ObjectId("60e5ea9aade15b7261a741b7"), "title" : "Fight Club", "writer" : "Chuck Palahniuk", "year" : "1999", "actors" : [ "Brad Pitt", "Edward Norton" ] }

{ "\_id" : ObjectId("60e5eac5ade15b7261a741b9"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY>

4.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({franchise:"The Hobbit"})

{ "\_id" : ObjectId("60e5ead5ade15b7261a741ba"), "title" : "The Hobbit: An unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e5eae1ade15b7261a741bb"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "year" : "2013", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e5eaefade15b7261a741bc"), "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." }

5.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({year:{$gt:"1990", $lt:"2000"}})

{ "\_id" : ObjectId("60e5ea9aade15b7261a741b7"), "title" : "Fight Club", "writer" : "Chuck Palahniuk", "year" : "1999", "actors" : [ "Brad Pitt", "Edward Norton" ] }

{ "\_id" : ObjectId("60e5eab8ade15b7261a741b8"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : "1994", "actors" : [ "John Travolta", "Uma Thurman" ] }

6.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({$or:[{year:{$gt:"2010"}},{year: {$lt:"2000"}}]})

{ "\_id" : ObjectId("60e5ea9aade15b7261a741b7"), "title" : "Fight Club", "writer" : "Chuck Palahniuk", "year" : "1999", "actors" : [ "Brad Pitt", "Edward Norton" ] }

{ "\_id" : ObjectId("60e5ead5ade15b7261a741ba"), "title" : "The Hobbit: An unexpected Journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "The Hobbit" }

{ "\_id" : ObjectId("60e5eae1ade15b7261a741bb"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "year" : "2013", "franchise" : "The Hobbit" }

## Update Documents

1.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.update({\_id:ObjectId("60e5ead5ade15b7261a741ba")}, {$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" : 1, "nUpserted" : 0, "nModified" : 1 })

2.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.update({\_id:ObjectId("60e5eae1ade15b7261a741bb")}, {$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" : 1, "nUpserted" : 0, "nModified" : 1 })

3.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.update({\_id:ObjectId("60e5eab8ade15b7261a741b8")}, {$push:{actors:"Samuel L. Jackson"}})

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

## Text Search

1.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({synopsis:{$regex:"Bilbo"}})

{ "\_id" : ObjectId("60e5ead5ade15b7261a741ba"), "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("60e5eae1ade15b7261a741bb"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "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("60e5eaefade15b7261a741bc"), "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("60e5ec94ade15b7261a741bf"), "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." }

2.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({synopsis:{$regex:"Gandalf"}})

{ "\_id" : ObjectId("60e5eae1ade15b7261a741bb"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "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." }

3.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({$and:[{synopsis:{$regex:"Bilbo"}}, {synopsis:{$not:/Gandalf/}}]})

{ "\_id" : ObjectId("60e5ead5ade15b7261a741ba"), "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("60e5eaefade15b7261a741bc"), "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("60e5ec94ade15b7261a741bf"), "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." }

4.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({$or:[{synopsis:{$regex:"dwarves"}}, {synopsis:{$regex:"hobbit"}}]})

{ "\_id" : ObjectId("60e5ead5ade15b7261a741ba"), "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("60e5eae1ade15b7261a741bb"), "title" : "The Hobbit: The Desolation of Smaug", "writer" : "J.R.R Tolkien", "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." }

5.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.find({$and:[{synopsis:{$regex:"gold"}}, {synopsis:{$regex:"dragon"}}]})

{ "\_id" : ObjectId("60e5ead5ade15b7261a741ba"), "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." }

## Delete Documents

1.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.remove({\_id:ObjectId("60e5eb4bade15b7261a741bd")})

WriteResult({ "nRemoved" : 1 })

2.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.movies.remove({\_id:ObjectId("60e5eb4bade15b7261a741be")})

WriteResult({ "nRemoved" : 0 })

**Relationships**

1.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.users.insert({\_id:1,username:"GoodGuyGreg", first\_name:"Good Guy", last\_name:"Greg"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.users.insert({\_id:2, username:"ScumbagSteve", fullname:{first: "Scumbag", last:"Steve"}})

WriteResult({ "nInserted" : 1 })

2.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.posts.insert({username:"GoodGuyGreg", title:"Passes out at Party", body:"Raises your credit score"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.posts.insert({ username:"GoodGuyGreg", title:"Steals your identity", body:"Raises your credit score"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-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-5d45wr-shard-0:PRIMARY> db.posts.insert({ username:"ScumbagSteve", title:"Borrows something", body:"Sells it"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.posts.insert({ username:"ScumbagSteve", title:"Borrows everything", body:"The end"})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.posts.insert({username:"ScumbagSteve", title:"Forks your repo on github", body:"Sets to private"})

WriteResult({ "nInserted" : 1 })

3.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.posts.find()

{ "\_id" : ObjectId("60e5fbe7ade15b7261a741c0"), "username" : "GoodGuyGreg", "title" : "Passes out at Party", "body" : "Raises your credit score" }

{ "\_id" : ObjectId("60e5fc03ade15b7261a741c1"), "username" : "GoodGuyGreg", "title" : "Steals your identity", "body" : "Raises your credit score" }

{ "\_id" : ObjectId("60e5fc1dade15b7261a741c2"), "username" : "GoodGuyGreg", "title" : "Reports a bug in your code", "body" : "Sends you a pull request" }

{ "\_id" : ObjectId("60e5fc31ade15b7261a741c3"), "username" : "ScumbagSteve", "title" : "Borrows something", "body" : "Sells it" }

{ "\_id" : ObjectId("60e5fc48ade15b7261a741c4"), "username" : "ScumbagSteve", "title" : "Borrows everything", "body" : "The end" }

{ "\_id" : ObjectId("60e5fc9bade15b7261a741c5"), "username" : "ScumbagSteve", "title" : "Forks your repo on github", "body" : "Sets to private" }

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.comments.insert({ username:"GoodGuyGreg", comment:"Hope you got a good deal!", post:ObjectId("60e5fc31ade15b7261a741c3")})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.comments.insert({username:"GoodGuyGreg", comment:"What's mine is yours!", post:ObjectId("60e5fc48ade15b7261a741c4")})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.comments.insert({username:"GoodGuyGreg", comment:"Don't violate the licensing agreement!", post:ObjectId("60e5fc9bade15b7261a741c5")})

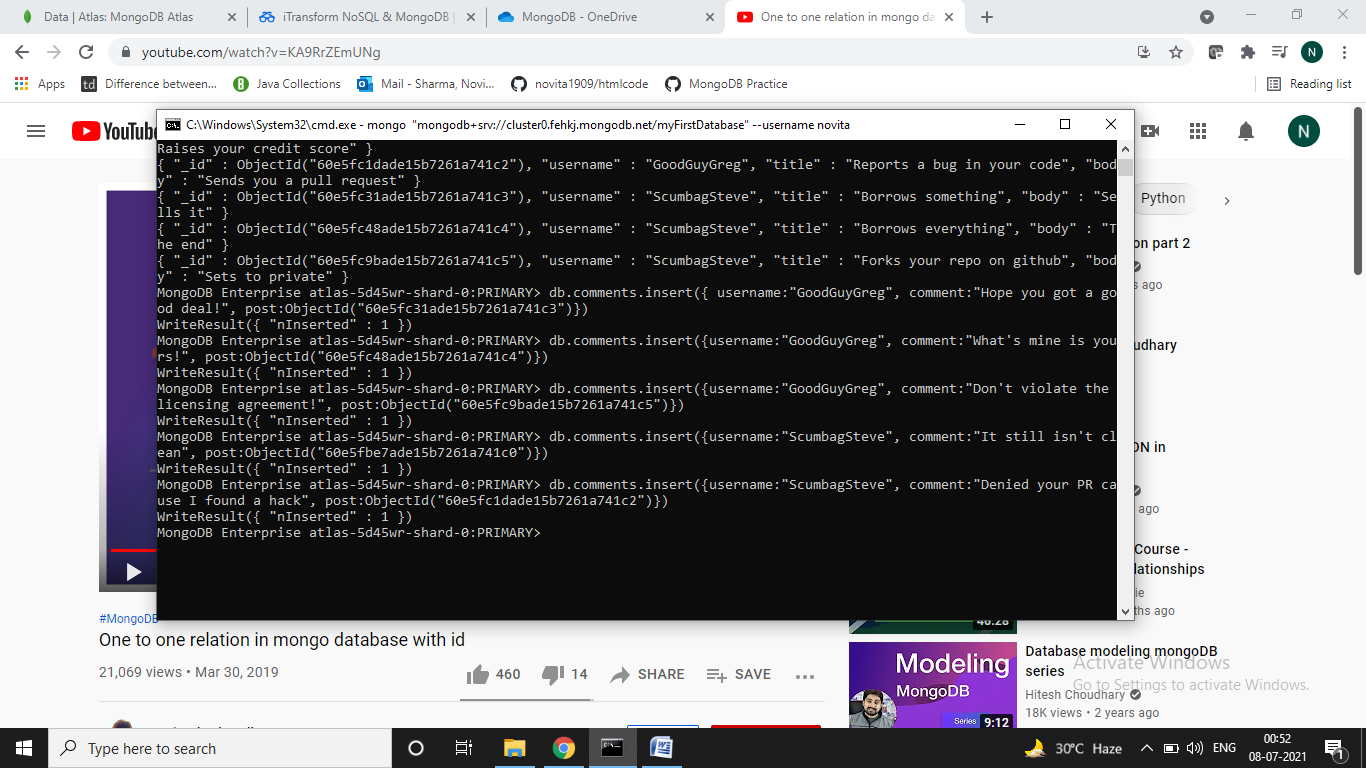
WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.comments.insert({username:"ScumbagSteve", comment:"It still isn't clean", post:ObjectId("60e5fbe7ade15b7261a741c0")})

WriteResult({ "nInserted" : 1 })

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.comments.insert({username:"ScumbagSteve", comment:"Denied your PR cause I found a hack", post:ObjectId("60e5fc1dade15b7261a741c2")})

WriteResult({ "nInserted" : 1 })



**Querying related collections**

1.

MongoDB Enterprise atlas-5d45wr-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"

}

}

2.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.posts.find().pretty()

{

"\_id" : ObjectId("60e5fbe7ade15b7261a741c0"),

"username" : "GoodGuyGreg",

"title" : "Passes out at Party",

"body" : "Raises your credit score"

}

{

"\_id" : ObjectId("60e5fc03ade15b7261a741c1"),

"username" : "GoodGuyGreg",

"title" : "Steals your identity",

"body" : "Raises your credit score"

}

{

"\_id" : ObjectId("60e5fc1dade15b7261a741c2"),

"username" : "GoodGuyGreg",

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

"body" : "Sends you a pull request"

}

{

"\_id" : ObjectId("60e5fc31ade15b7261a741c3"),

"username" : "ScumbagSteve",

"title" : "Borrows something",

"body" : "Sells it"

}

{

"\_id" : ObjectId("60e5fc48ade15b7261a741c4"),

"username" : "ScumbagSteve",

"title" : "Borrows everything",

"body" : "The end"

}

{

"\_id" : ObjectId("60e5fc9bade15b7261a741c5"),

"username" : "ScumbagSteve",

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

"body" : "Sets to private"

}

3.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.posts.find({username:"GoodGuyGreg"})

{ "\_id" : ObjectId("60e5fbe7ade15b7261a741c0"), "username" : "GoodGuyGreg", "title" : "Passes out at Party", "body" : "Raises your credit score" }

{ "\_id" : ObjectId("60e5fc03ade15b7261a741c1"), "username" : "GoodGuyGreg", "title" : "Steals your identity", "body" : "Raises your credit score" }

{ "\_id" : ObjectId("60e5fc1dade15b7261a741c2"), "username" : "GoodGuyGreg", "title" : "Reports a bug in your code", "body" : "Sends you a pull request" }

4.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.posts.find({username:"ScumbagSteve"})

{ "\_id" : ObjectId("60e5fc31ade15b7261a741c3"), "username" : "ScumbagSteve", "title" : "Borrows something", "body" : "Sells it" }

{ "\_id" : ObjectId("60e5fc48ade15b7261a741c4"), "username" : "ScumbagSteve", "title" : "Borrows everything", "body" : "The end" }

{ "\_id" : ObjectId("60e5fc9bade15b7261a741c5"), "username" : "ScumbagSteve", "title" : "Forks your repo on github", "body" : "Sets to private" }

5.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.comments.find().pretty()

{

"\_id" : ObjectId("60e5fd5dade15b7261a741c6"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("60e5fc31ade15b7261a741c3")

}

{

"\_id" : ObjectId("60e5fda0ade15b7261a741c7"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("60e5fc48ade15b7261a741c4")

}

{

"\_id" : ObjectId("60e5fe20ade15b7261a741c8"),

"username" : "GoodGuyGreg",

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

"post" : ObjectId("60e5fc9bade15b7261a741c5")

}

{

"\_id" : ObjectId("60e5fe4eade15b7261a741c9"),

"username" : "ScumbagSteve",

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

"post" : ObjectId("60e5fbe7ade15b7261a741c0")

}

{

"\_id" : ObjectId("60e5fe77ade15b7261a741ca"),

"username" : "ScumbagSteve",

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

"post" : ObjectId("60e5fc1dade15b7261a741c2")

}

6.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.comments.find({username:"GoodGuyGreg"})

{ "\_id" : ObjectId("60e5fd5dade15b7261a741c6"), "username" : "GoodGuyGreg", "comment" : "Hope you got a good deal!", "post" : ObjectId("60e5fc31ade15b7261a741c3") }

{ "\_id" : ObjectId("60e5fda0ade15b7261a741c7"), "username" : "GoodGuyGreg", "comment" : "What's mine is yours!", "post" : ObjectId("60e5fc48ade15b7261a741c4") }

{ "\_id" : ObjectId("60e5fe20ade15b7261a741c8"), "username" : "GoodGuyGreg", "comment" : "Don't violate the licensing agreement!", "post" : ObjectId("60e5fc9bade15b7261a741c5") }

7.

MongoDB Enterprise atlas-5d45wr-shard-0:PRIMARY> db.comments.find({username:"ScumbagSteve"})

{ "\_id" : ObjectId("60e5fe4eade15b7261a741c9"), "username" : "ScumbagSteve", "comment" : "It still isn't clean", "post" : ObjectId("60e5fbe7ade15b7261a741c0") }

{ "\_id" : ObjectId("60e5fe77ade15b7261a741ca"), "username" : "ScumbagSteve", "comment" : "Denied your PR cause I found a hack", "post" : ObjectId("60e5fc1dade15b7261a741c2") }

8.

db.comments.find({title:”Reports sa bug in your code”})

Screenshots of some of the queries—

