MongoDB Lab Assignments -Day 1

Query / Find Documents:

query the movies collection to

**1. get all documents**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.insertMany([{'title':'Fight Club', 'writer': 'Chuck Palahniuko','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 Krunger','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 Hermans Big Adventure'},{'title':'Avatar'}])

{

acknowledged: true,

insertedIds: {

'0': ObjectId("617a6225304f6b88dbc1870e"),

'1': ObjectId("617a6225304f6b88dbc1870f"),

'2': ObjectId("617a6225304f6b88dbc18710"),

'3': ObjectId("617a6225304f6b88dbc18711"),

'4': ObjectId("617a6225304f6b88dbc18712"),

'5': ObjectId("617a6225304f6b88dbc18713"),

'6': ObjectId("617a6225304f6b88dbc18714"),

'7': ObjectId("617a6225304f6b88dbc18715")

}

}

**1. get all documents**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find().pretty()

[

{

\_id: ObjectId("617a600e304f6b88dbc1870c"),

title: 'Fight Club',

writer: 'Chuck Palahniuko',

year: 1999,

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

},

{

\_id: ObjectId("617a6134304f6b88dbc1870d"),

title: 'Pulp Fiction',

writer: 'Quentin Tarantino',

year: 1994,

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

},

{

\_id: ObjectId("617a6225304f6b88dbc1870e"),

title: 'Fight Club',

writer: 'Chuck Palahniuko',

year: 1999,

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

},

{

\_id: ObjectId("617a6225304f6b88dbc1870f"),

title: 'Pulp Fiction',

writer: 'Quentin Tarantino',

year: 1994,

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

},

{

\_id: ObjectId("617a6225304f6b88dbc18710"),

title: 'Inglorious Basterds',

writer: 'Quentin Tarantino',

year: 2009,

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

},

{

\_id: ObjectId("617a6225304f6b88dbc18711"),

title: 'The Hobbit: An Unexpected Journey',

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

year: 2012,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("617a6225304f6b88dbc18712"),

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

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

year: 2013,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("617a6225304f6b88dbc18713"),

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("617a6225304f6b88dbc18714"),

title: 'Pee Wee Hermans Big Adventure'

},

{ \_id: ObjectId("617a6225304f6b88dbc18715"), title: 'Avatar' }

]

**2.get all documents where actors include "Brad Pitt"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find({actors:'Brad Pitt'})

[

{

\_id: ObjectId("617a600e304f6b88dbc1870c"),

title: 'Fight Club',

writer: 'Chuck Palahniuko',

year: 1999,

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

},

{

\_id: ObjectId("617a6225304f6b88dbc1870e"),

title: 'Fight Club',

writer: 'Chuck Palahniuko',

year: 1999,

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

},

{

\_id: ObjectId("617a6225304f6b88dbc18710"),

title: 'Inglorious Basterds',

writer: 'Quentin Tarantino',

year: 2009,

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

}

]

1. **get all documents with franchise set to "The Hobbit"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find({franchise: 'The Hobbit'})

[

{

\_id: ObjectId("617a6225304f6b88dbc18711"),

title: 'The Hobbit: An Unexpected Journey',

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

year: 2012,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("617a6225304f6b88dbc18712"),

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

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

year: 2013,

franchise: 'The Hobbit'

},

{

\_id: ObjectId("617a6225304f6b88dbc18713"),

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.'

}

]

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

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find({year:{$lt: 2010, $gt: 2000}})

[

{

\_id: ObjectId("617a6225304f6b88dbc18710"),

title: 'Inglorious Basterds',

writer: 'Quentin Tarantino',

year: 2009,

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

}

]

# Update Documents

1. **add a synopsis to "The Hobbit: An Unexpected Journey" : "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."**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.update({title:'The Hobbit: An Unexpected Journey'},{$set: {synopsis: 'A reluctant hobbit, Bilbo Baggins, sets out to the Lonely Mountain with a spirited roup of dwarves to reclaim their mountain home - and the gold within it - from the dragon smaug.'} })

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

1. **add a synopsis to "The Hobbit: The Desolation of Smaug" : "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."**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> 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 magical ring.'} })

{

acknowledged: true,

insertedId: null,

matchedCount: 0,

modifiedCount: 0,

upsertedCount: 0

}

1. **add an actor named "Samuel L. Jackson" to the movie "Pulp Fiction"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.update({title:'Pulp Fiction'},{$push:{actors:' Samuel L. Jackson'}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

# Text Search

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.createIndex({synopsis:'text'})

synopsis\_text

1. **find all movies that have a synopsis that contains the word "Bilbo"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find({$text:{$search:'Bilbo'}})

[

{

\_id: ObjectId("617a6225304f6b88dbc18713"),

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("617a6225304f6b88dbc18711"),

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 roup of dwarves to reclaim their mountain home - and the gold within it - from the dragon smaug.'

}

]

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

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find({$text:{$search:'Gandalf'}})

1. **find all movies that have a synopsis that contains the word "Bilbo" and not the word "Gandalf"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find({$text:{$search:'Bilbo -Gandalf'}})

[

{

\_id: ObjectId("617a6225304f6b88dbc18713"),

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("617a6225304f6b88dbc18711"),

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 roup of dwarves to reclaim their mountain home - and the gold within it - from the dragon smaug.'

}

]

1. **find all movies that have a synopsis that contains the word "dwarves" or "hobbit"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find({$text:{$search:'dwarves hobbit'}})

[

{

\_id: ObjectId("617a6225304f6b88dbc18711"),

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 roup of dwarves to reclaim their mountain home - and the gold within it - from the dragon smaug.'

}

]

1. **find all movies that have a synopsis that contains the word "gold" and "dragon"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.find({$text:{$search:"'gold' 'dragon'"}})

[

{

\_id: ObjectId("617a6225304f6b88dbc18711"),

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 roup of dwarves to reclaim their mountain home - and the gold within it - from the dragon smaug.'

}

]

# Delete Documents

1. **delete the movie "Avatar"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.remove({title:"Avatar"})

{ acknowledged: true, deletedCount: 0 }

1. **delete the movie "Pee Wee Herman's Big Adventure"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.movies.remove({title:"Pee Wee Herman's Big Adventure"})

{ acknowledged: true, deletedCount: 0 }

# Relationships

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.createCollection('users')

{ ok: 1 }

insert the following documents into a **users** collection

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.users.insertMany([ { username: "GoodGuyGreg", first\_name:"Good Guy",last\_name:"Greg"},{ username: "ScumbagSteve", full\_name:{first\_name:"Scumbag",last\_name:"Steve"} } ]);

{

acknowledged: true,

insertedIds: {

'0': ObjectId("617a7e24304f6b88dbc18716"),

'1': ObjectId("617a7e24304f6b88dbc18717")

}

}

Insert the following documents into a **posts** collection

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> 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:"Forks your repo on github",body:"Sets to private"} ]);

{

acknowledged: true,

insertedIds: {

'0': ObjectId("617a801d304f6b88dbc1871d"),

'1': ObjectId("617a801d304f6b88dbc1871e"),

'2': ObjectId("617a801d304f6b88dbc1871f"),

'3': ObjectId("617a801d304f6b88dbc18720"),

'4': ObjectId("617a801d304f6b88dbc18721"),

'5': ObjectId("617a801d304f6b88dbc18722")

}

}

Insert the following documents into a **comments** collection

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.comments.insertMany([{ username: "GoodGuyGreg", comment:"Hope you got a good deal!",post:"607f49ca845ea1cba8e2ij18"},{username: "GoodGuyGreg", comment:"What's mine is yours!",post:"607f49ca845ea1cba8e2ij19"},{ username: "GoodGuyGreg", comment:"Don't violate the licensing agreement!",post:"607f49ca845ea1cba8e2ij20"},{ username: "ScumbagSteve", comment:"It still isn't clean",post:"607f49ca845ea1cba8e2ij21"},{ username: "ScumbagSteve", comment:"Denied your PR cause I found a hack",post:"607f49ca845ea1cba8e2ij22"}]);

{

acknowledged: true,

insertedIds: {

'0': ObjectId("617a8854304f6b88dbc18732"),

'1': ObjectId("617a8854304f6b88dbc18733"),

'2': ObjectId("617a8854304f6b88dbc18734"),

'3': ObjectId("617a8854304f6b88dbc18735"),

'4': ObjectId("617a8854304f6b88dbc18736")

}

}

1. **find all users**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.user.find()

**2. find all posts**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.posts.find()

[

{

\_id: ObjectId("617a7fcc304f6b88dbc18718"),

username: 'GoodGuyGreg',

title: 'Passes out at party',

body: 'Wakes up early and cleans house'

},

{

\_id: ObjectId("617a7fcc304f6b88dbc18719"),

username: 'GoodGuyGreg',

title: 'Steals your identity',

body: 'Raises your credit score'

},

{

\_id: ObjectId("617a7fcc304f6b88dbc1871a"),

username: 'GoodGuyGreg',

title: 'Reports a bug in your code',

body: 'Sends you a pull request'

},

{

\_id: ObjectId("617a7fcc304f6b88dbc1871b"),

username: 'GoodGuyGreg',

title: 'Passes out at party',

body: 'Wakes up early and cleans house'

},{ username: "ScumbagSteve", comment:"Denied your PR cause I found a hack",post:"607f49ca845ea1cba8e2ij22"}

{ ]);

\_id: ObjectId("617a7fcc304f6b88dbc1871c"),

username: 'ScumbagSteve',

title: 'Borrows something',

body: 'Sells it'17a80f3304f6b88dbc18728"),

},'1': ObjectId("617a80f3304f6b88dbc18729"),

{ '2': ObjectId("617a80f3304f6b88dbc1872a"),

\_id: ObjectId("617a801d304f6b88dbc1871d"),

username: 'GoodGuyGreg',04f6b88dbc1872c")

title: 'Passes out at party',

body: 'Wakes up early and cleans house'

},s atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.comments.insertMany([{u

{

\_id: ObjectId("617a801d304f6b88dbc1871e"),

username: 'GoodGuyGreg',

title: 'Steals your identity',

body: 'Raises your credit score'

},

{

\_id: ObjectId("617a801d304f6b88dbc1871f"),

username: 'GoodGuyGreg',

title: 'Reports a bug in your code',

body: 'Sends you a pull request'

},

{

\_id: ObjectId("617a801d304f6b88dbc18720"),

username: 'ScumbagSteve',

title: 'Borrows something',

body: 'Sells it'

},

{

\_id: ObjectId("617a801d304f6b88dbc18721"),

username: 'ScumbagSteve',

title: 'Borrows everything',

body: 'The end'

},

{

\_id: ObjectId("617a801d304f6b88dbc18722"),

username: 'ScumbagSteve',

title: 'Forks your repo on github',

body: 'Sets to private'

}

]

1. **find all posts that was authored by "GoodGuyGreg"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.posts.find({username:"GoodGuyGreg"})

[

{

\_id: ObjectId("617a7fcc304f6b88dbc18718"),

username: 'GoodGuyGreg',

title: 'Passes out at party',

body: 'Wakes up early and cleans house'

},

{

\_id: ObjectId("617a7fcc304f6b88dbc18719"),

username: 'GoodGuyGreg',

title: 'Steals your identity',

body: 'Raises your credit score'

},

{

\_id: ObjectId("617a7fcc304f6b88dbc1871a"),

username: 'GoodGuyGreg',

title: 'Reports a bug in your code',

body: 'Sends you a pull request'

},

{

\_id: ObjectId("617a7fcc304f6b88dbc1871b"),

username: 'GoodGuyGreg',

title: 'Passes out at party',

body: 'Wakes up early and cleans house'

},

{

\_id: ObjectId("617a801d304f6b88dbc1871d"),

username: 'GoodGuyGreg',

title: 'Passes out at party',

body: 'Wakes up early and cleans house'

},

{

\_id: ObjectId("617a801d304f6b88dbc1871e"),

username: 'GoodGuyGreg',

title: 'Steals your identity',

body: 'Raises your credit score'

},

{

\_id: ObjectId("617a801d304f6b88dbc1871f"),

username: 'GoodGuyGreg',

title: 'Reports a bug in your code',

body: 'Sends you a pull request'

}

]

1. **find all posts that was authored by "ScumbagSteve"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.posts.find({username:"ScumbagSteve"})

[

{

\_id: ObjectId("617a7fcc304f6b88dbc1871c"),

username: 'ScumbagSteve',

title: 'Borrows something',

body: 'Sells it'

},

{

\_id: ObjectId("617a801d304f6b88dbc18720"),

username: 'ScumbagSteve',

title: 'Borrows something',

body: 'Sells it'

},

{

\_id: ObjectId("617a801d304f6b88dbc18721"),

username: 'ScumbagSteve',

title: 'Borrows everything',

body: 'The end'

},

{

\_id: ObjectId("617a801d304f6b88dbc18722"),

username: 'ScumbagSteve',

title: 'Forks your repo on github',

body: 'Sets to private'

}

]

1. **find all comments**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.comments.find()

[

{

\_id: ObjectId("617a80d1304f6b88dbc18723"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij18'

},

{

\_id: ObjectId("617a80d1304f6b88dbc18724"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij19'

},

{

\_id: ObjectId("617a80d1304f6b88dbc18725"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij20'

},

{

\_id: ObjectId("617a80d1304f6b88dbc18726"),

username: 'ScumbagSteve',

comment: "It still isn't clean",

post: '607f49ca845ea1cba8e2ij21'

},

{

\_id: ObjectId("617a80d1304f6b88dbc18727"),

username: 'ScumbagSteve',

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

post: '607f49ca845ea1cba8e2ij22'

},

{

\_id: ObjectId("617a80f3304f6b88dbc18728"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij18'

},

{

\_id: ObjectId("617a80f3304f6b88dbc18729"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij19'

},

{

\_id: ObjectId("617a80f3304f6b88dbc1872a"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij20'

},

{

\_id: ObjectId("617a80f3304f6b88dbc1872b"),

username: 'ScumbagSteve',

comment: "It still isn't clean",

post: '607f49ca845ea1cba8e2ij21'

},

{

\_id: ObjectId("617a80f3304f6b88dbc1872c"),

username: 'ScumbagSteve',

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

post: '607f49ca845ea1cba8e2ij22'

},

{

\_id: ObjectId("617a86e2304f6b88dbc1872d"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij18'

},

{

\_id: ObjectId("617a86e2304f6b88dbc1872e"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij19'

},

{

\_id: ObjectId("617a86e2304f6b88dbc1872f"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij20'

},

{

\_id: ObjectId("617a86e2304f6b88dbc18730"),

username: 'ScumbagSteve',

comment: "It still isn't clean",

post: '607f49ca845ea1cba8e2ij21'

},

{

\_id: ObjectId("617a86e2304f6b88dbc18731"),

username: 'ScumbagSteve',

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

post: '607f49ca845ea1cba8e2ij22'

},

{

\_id: ObjectId("617a8854304f6b88dbc18732"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij18'

},

{

\_id: ObjectId("617a8854304f6b88dbc18733"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij19'

},

{

\_id: ObjectId("617a8854304f6b88dbc18734"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij20'

},

{

\_id: ObjectId("617a8854304f6b88dbc18735"),

username: 'ScumbagSteve',

comment: "It still isn't clean",

post: '607f49ca845ea1cba8e2ij21'

},

{

\_id: ObjectId("617a8854304f6b88dbc18736"),

username: 'ScumbagSteve',

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

post: '607f49ca845ea1cba8e2ij22'

}

]

Type "it" for more

1. **find all comments that was authored by "GoodGuyGreg"**

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.comments.find({username:"GoodGuyGreg"})

[

{

\_id: ObjectId("617a80d1304f6b88dbc18723"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij18'

},

{

\_id: ObjectId("617a80d1304f6b88dbc18724"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij19'

},

{

\_id: ObjectId("617a80d1304f6b88dbc18725"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij20'

},

{

\_id: ObjectId("617a80f3304f6b88dbc18728"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij18'

},

{

\_id: ObjectId("617a80f3304f6b88dbc18729"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij19'

},

{

\_id: ObjectId("617a80f3304f6b88dbc1872a"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij20'

},

{

\_id: ObjectId("617a86e2304f6b88dbc1872d"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij18'

},

{

\_id: ObjectId("617a86e2304f6b88dbc1872e"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij19'

},

{

\_id: ObjectId("617a86e2304f6b88dbc1872f"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij20'

},

{

\_id: ObjectId("617a8854304f6b88dbc18732"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij18'

},

{

\_id: ObjectId("617a8854304f6b88dbc18733"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij19'

},

{

\_id: ObjectId("617a8854304f6b88dbc18734"),

username: 'GoodGuyGreg',

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

post: '607f49ca845ea1cba8e2ij20'

}

]

1. find all comments that was authored by "ScumbagSteve"

Atlas atlas-7z2ycj-shard-0 [primary] mongodb\_practice> db.comments.find({username:"ScumbagSteve"})

[

{

\_id: ObjectId("617a80d1304f6b88dbc18726"),

username: 'ScumbagSteve',

comment: "It still isn't clean",

post: '607f49ca845ea1cba8e2ij21'

},

{

\_id: ObjectId("617a80d1304f6b88dbc18727"),

username: 'ScumbagSteve',

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

post: '607f49ca845ea1cba8e2ij22'

},

{

\_id: ObjectId("617a80f3304f6b88dbc1872b"),

username: 'ScumbagSteve',

comment: "It still isn't clean",

post: '607f49ca845ea1cba8e2ij21'

},

{

\_id: ObjectId("617a80f3304f6b88dbc1872c"),

username: 'ScumbagSteve',

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

post: '607f49ca845ea1cba8e2ij22'

},

{

\_id: ObjectId("617a86e2304f6b88dbc18730"),

username: 'ScumbagSteve',

comment: "It still isn't clean",

post: '607f49ca845ea1cba8e2ij21'

},

{

\_id: ObjectId("617a86e2304f6b88dbc18731"),

username: 'ScumbagSteve',

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

post: '607f49ca845ea1cba8e2ij22'

},

{

\_id: ObjectId("617a8854304f6b88dbc18735"),

username: 'ScumbagSteve',

comment: "It still isn't clean",

post: '607f49ca845ea1cba8e2ij21'

},

{

\_id: ObjectId("617a8854304f6b88dbc18736"),

username: 'ScumbagSteve',

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

post: '607f49ca845ea1cba8e2ij22'

}

]