

MongoDB Lab Assignments – A1

MongoDB Exercise in mongo shell

Connect to a running mongo instance, use a database named mongo_practice. Document all your queries in a javascript file to use as a reference.

Insert Documents

Insert the following documents into a movies collection.

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 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: Avatar

Reference

https://www.tutorialspoint.com/mongodb/mongodb_insert_document.htm

Ans:- 1st step = show dbs
2nd = use mongo_practice
switched to db mongo_practice
3rd = show collections
movies
users

for inserting documents:-

```
➤ 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 Kruger", "Eli Roth"]}, {Title: "The Hobbit:An Unexpected Journey", Writer: "J.R.R. Tolkein", year: 2012, franchise: "The Hobbit"}, {Title: "The Hobbit:The Dessolation 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 combantants and keep the Lonely Mountain form falling into hands of a rising darkness."}, {Title: "Pee Wee Herman's Blg Adventure"}, {Title: "Avatar"}}])

{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("60a4cade4a251f11e4193da8"),
    ObjectId("60a4cade4a251f11e4193da9"),
    ObjectId("60a4cade4a251f11e4193daa"),
    ObjectId("60a4cade4a251f11e4193dab"),
    ObjectId("60a4cade4a251f11e4193dac"),
    ObjectId("60a4cade4a251f11e4193dad"),
    ObjectId("60a4cade4a251f11e4193dae"),
    ObjectId("60a4cade4a251f11e4193daf")
  ]
}
```

Query/Find Documents

query the movies collection to

1. get all documents
Ans:- > db.movies.find().pretty()

```
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
```

```
]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "Title" : "The Hobbit:An Unexpected Journey",
  "Writer" : "J.R.R. Tolkein",
  "Year" : 2012,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkein",
  "year" : 2013,
  "franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dad"),
  "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 combantants and keep the Lonely Mountain form falling into hands of a rising darkness."
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dae"),
  "Title" : "Pee Wee Herman's Blg Adventure"
}
{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
```

2. get all documents with writer set to "Quentin Tarantino"

Ans:- > db.movies.find({Writer: "Quentin Tarantino"}).pretty()

```
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
```

3. get all documents where actors include "Brad Pitt"

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

```
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
```

4. get all documents with franchise set to "The Hobbit"

Ans:- > db.movies.find({Franchise: "The Hobbit"}).pretty()

```
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "Title" : "The Hobbit:An Unexpected Journey",
  "Writer" : "J.R.R. Tolkein",
  "Year" : 2012,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkein",
  "year" : 2013,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dad"),
  "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 combantants and keep the Lonely Mountain form falling into hands of a rising darkness."
}
```

5. get all movies released in the 90s

Ans:- db.movies.find({"Year":{\$gte:1990, \$lt:2000}}).pretty()

```
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
```

```
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}
```

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

Ans:-

Reference:

https://www.tutorialspoint.com/mongodb/mongodb_query_document.htm

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."

Ans:- > 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 group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."}})

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

2. 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."

Ans:- > db.movies.update({Title: "The Hobbit:The Dessolation 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" : 1, "nUpserted" : 0, "nModified" : 1 })

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

Ans:- > db.movies.update({Title: "Pulp Fiction"},{\$set:{actors:"Samuel L. Jackson"}})

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

Reference:

https://www.tutorialspoint.com/mongodb/mongodb_update_document.htm

Text Search

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

Ans:- > db.movies.createIndex({ Title: "text", Synopses: "text"}) // You have to create once after that search only

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

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

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

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

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

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

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

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

Ans:- > db.movies.find({\$text: {\$search: "gold dragon"}}).pretty()

Reference: https://www.tutorialspoint.com/mongodb/mongodb_text_search.htm

Delete Documents

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

Ans:- > db.movies.remove({Title: "Pee Wee Herman's Blg Adventure"})

WriteResult({ "nRemoved" : 1 })

2. delete the movie "Avatar"

Ans:- db.movies.remove({Title: "Avatar"})

WriteResult({ "nRemoved" : 1 })

Reference: https://www.tutorialspoint.com/mongodb/mongodb_delete_document.htm

Relationships

Insert the following documents into a users collection

username: GoodGuyGreg

first name: "Good Guy"

last name: "Greg"

Ans:- > db.users.insertOne({"username" : "GoodGuyGreg", "first_name" : "GoodGuy", "last_name" : "Greg"})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6388d1c2f9c0eb4c3db9b")
}
```

username: ScumbagSteve

full_name:

first: "Scumbag"

last: "Steve"

Ans:- db.users.insertOne({"username" : "SchumbagSteve" , "full_name" : {"first" : "Scumbag" , "last" : "Steve"}})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a639451c2f9c0eb4c3db9c")
}
```

Insert the following documents into a posts collection

username: GoodGuyGreg

title: Passes out at party

body: Wakes up early and cleans house

Ans:- db.posts.insertOne({"username" : "GoodGuyGreg" , "title": " Passes out at a party" , "Body": "Wakes up early and cleans house"})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a650261c2f9c0eb4c3db9f")
}
```

username: GoodGuyGreg

title: Steals your identity

body: Raises your credit score

Ans:- > db.posts.insertOne({"username" : "GoodGuyGreg" , "title": "Steals your identity" , "Body": "Raises your credit score"})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a650691c2f9c0eb4c3dba0")
}
```

username: GoodGuyGreg

title: Reports a bug in your code

body: Sends you a Pull Request

Ans:- > db.posts.insertOne({"username" : "GoodGuyGreg" , "title":"Reports a bug in your code" , "Body": "Sends you a Pull Request"})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a650a91c2f9c0eb4c3dba1")
}
```

username: ScumbagSteve

title: Borrows something

body: Sells it

Ans:- > db.posts.insertOne({"username" : "ScumbagSteve" , "title":"Borrows Something" , "Body": "Sells it"})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6511a1c2f9c0eb4c3dba2")
}
```

username: ScumbagSteve

title: Borrows everything

body: The end

Ans:- > db.posts.insertOne({"username" : "ScumbagSteve" , "title":"Borrows Everything" , "Body": "The End"})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6513f1c2f9c0eb4c3dba3")
}
```

username: ScumbagSteve

title: Forks your repo on github

body: Sets to private

Ans:- > db.posts.insertOne({"username" : "ScumbagSteve" , "title":"Forks your repo on github" , "Body": "Sets to private"})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6516c1c2f9c0eb4c3dba4")
}
```

Insert the following documents into a comments collection

username: GoodGuyGreg

comment: Hope you got a good deal!

post [post_obj_id] where [post_obj_id] is the Objectid of the posts document: "Borrows something"

Ans:- atlas-b9aweb-shard-0:PRIMARY> db.comments.insertOne({"username" : "GoodGuyGreg" ,"comment": "Hope you got a deal!" , "post": ObjectId("60a6511a1c2f9c0eb4c3dba2")})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a656221c2f9c0eb4c3dba5")
}
```

username: GoodGuyGreg

comment : What's mine is yours!

post [post_obj_id] where [post_obj_id] is the Objectid of the posts document: "Borrows everything"

Ans:- > db.comments.insertOne({"username" : "GoodGuyGreg" ,"comment": "What's mine is yours!" , "post": ObjectId("60a6513f1c2f9c0eb4c3dba3")})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6567e1c2f9c0eb4c3dba6")
}
```

username: GoodGuyGreg

comment : Don't violate the licensing agreement!

post [post_obj_id] where [post_obj_id] is the Objectid of the posts document: "Forks your repo on github"

Ans:- > db.comments.insertOne({"username" : "GoodGuyGreg" ,"comment": "Don't voilate licensing agreement!" , "post": ObjectId("60a6516c1c2f9c0eb4c3dba4")})

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a656e11c2f9c0eb4c3dba7")
}
```

```
}
```

```
username: ScumbagSteve
comment: It still isn't clean
post [post_obj_id] where [post_obj_id] is the Objectid of the posts document: "Passes out at party"
Ans:- > db.comments.insertOne({"username" : "GoodGuyGreg" ,"comment": "It still insn't clean!", "post": ObjectId("60a650261c2f9c0eb4c3db9f")})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a657b21c2f9c0eb4c3dba8")
}
```

```
username: ScumbagSteve
comment : Denied your PR cause I found a hack
post [post_obj_id] where [post_obj_id] is the Objectid of the posts document: "Reports a bug in your code"
Ans:- > db.comments.insertOne({"username" : "ScumbagSteve" ,"comment": "Denied your PR cause I found a hack", "post":
ObjectId("60a650a91c2f9c0eb4c3dba1")})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a658621c2f9c0eb4c3dbaa")
}
```

Querying related collections

1. find all users
Ans:- > db.users.find().pretty()
2. find all posts
Ans:- > db.posts.find().pretty()
3. find all posts that was authored by "GoodGuyGreg"
Ans:- > db.posts.find({username : "GoodGuyGreg"}).pretty()
4. find all posts that was authored by "ScumbagSteve"
Ans:- > > db.posts.find({"username" : "ScumbagSteve"}).pretty()
5. find all comments
Ans:- > db.comments.find().pretty()
6. find all comments that was authored by "GoodGuyGreg"
Ans:- db.comments.find({"username" : "GoodGuyGreg"}).pretty()
7. find all comments that was authored by "ScumbagSteve"
Ans:- > db.comments.find({"username" : "ScumbagSteve"}).pretty()
8. find all comments belonging to the post "Reports a bug in your code"
Ans:- > db.comments.find({"post" : ObjectId("60a650a91c2f9c0eb4c3dba1")}).pretty()

References:

<https://docs.mongodb.com/manual/reference/method/db.collection.find/>

ScreenShots:-


```

    "comment" : "Denied your PR cause I found a hack",
    "post" : ObjectId("60a650a91c2f9c0eb4c3dba1")
  }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"username" : "GoodGuyGreg"}).pretty(
)
{
  "_id" : ObjectId("60a656221c2f9c0eb4c3dba5"),
  "username" : "GoodGuyGreg",
  "comment" : "Hope you got a deal!",
  "post" : ObjectId("60a6511a1c2f9c0eb4c3dba2")
}
{
  "_id" : ObjectId("60a6567e1c2f9c0eb4c3dba6"),
  "username" : "GoodGuyGreg",
  "comment" : "What's mine is yours!",
  "post" : ObjectId("60a6513f1c2f9c0eb4c3dba3")
}
{
  "_id" : ObjectId("60a656e11c2f9c0eb4c3dba7"),
  "username" : "GoodGuyGreg",
  "comment" : "Don't violate licensing agreement!",
  "post" : ObjectId("60a6516c1c2f9c0eb4c3dba4")
}
{
  "_id" : ObjectId("60a657b21c2f9c0eb4c3dba8"),
  "username" : "GoodGuyGreg",
  "comment" : "It still insn't clean!",
  "post" : ObjectId("60a650261c2f9c0eb4c3db9f")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"username" : "ScumbagSteve"}).pretty(
)
{
  "_id" : ObjectId("60a657f61c2f9c0eb4c3dba9"),
  "username" : "ScumbagSteve",
  "comment" : "It still insn't clean",
  "post" : ObjectId("60a650261c2f9c0eb4c3db9f")
}
{
  "_id" : ObjectId("60a658621c2f9c0eb4c3dbaa"),
  "username" : "ScumbagSteve",
  "comment" : "Denied your PR cause I found a hack",
  "post" : ObjectId("60a650a91c2f9c0eb4c3dba1")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"title" : "Reports a bug in your code
"}).pretty()
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"title" : "Reports a bug in your cod
e"}).pretty()
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"title" : "Reports a bug in your cod
e"}).pretty()

```

```
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.find({"username" : "ScumbagSteve"}).pretty()
{
  "_id" : ObjectId("60a6511a1c2f9c0eb4c3dba2"),
  "username" : "ScumbagSteve",
  "title" : "Borrows Something",
  "Body" : "Sells it"
}
{
  "_id" : ObjectId("60a6513f1c2f9c0eb4c3dba3"),
  "username" : "ScumbagSteve",
  "title" : "Borrows Everything",
  "Body" : "The End"
}
{
  "_id" : ObjectId("60a6516c1c2f9c0eb4c3dba4"),
  "username" : "ScumbagSteve",
  "title" : "Forks your repo on github",
  "Body" : "Sets to private"
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find().pretty()
{
  "_id" : ObjectId("60a656221c2f9c0eb4c3dba5"),
  "username" : "GoodGuyGreg",
  "comment" : "Hope you got a deal!",
  "post" : ObjectId("60a6511a1c2f9c0eb4c3dba2")
}
{
  "_id" : ObjectId("60a6567e1c2f9c0eb4c3dba6"),
  "username" : "GoodGuyGreg",
  "comment" : "What's mine is yours!",
  "post" : ObjectId("60a6513f1c2f9c0eb4c3dba3")
}
{
  "_id" : ObjectId("60a656e11c2f9c0eb4c3dba7"),
  "username" : "GoodGuyGreg",
  "comment" : "Don't violate licensing agreement!",
  "post" : ObjectId("60a6516c1c2f9c0eb4c3dba4")
}
{
  "_id" : ObjectId("60a657b21c2f9c0eb4c3dba8"),
  "username" : "GoodGuyGreg",
  "comment" : "It still insn't clean!",
  "post" : ObjectId("60a650261c2f9c0eb4c3db9f")
}
{
  "_id" : ObjectId("60a657f61c2f9c0eb4c3dba9"),
  "username" : "ScumbagSteve",
  "comment" : "It still insn't clean",
}
```



```
    "Body" : "Sets to private"
  }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.find({username : "GoodGuyGreg"}).pretty()
{
  "_id" : ObjectId("60a64f841c2f9c0eb4c3db9e"),
  "username" : "GoodGuyGreg",
  "title" : " Passes out at a party",
  "Body" : "Wakes up early and cleans house"
}
{
  "_id" : ObjectId("60a650261c2f9c0eb4c3db9f"),
  "username" : "GoodGuyGreg",
  "title" : " Passes out at a party",
  "Body" : "Wakes up early and cleans house"
}
{
  "_id" : ObjectId("60a650691c2f9c0eb4c3dba0"),
  "username" : "GoodGuyGreg",
  "title" : "Steals your identity",
  "Body" : "Raises your credit score"
}
{
  "_id" : ObjectId("60a650a91c2f9c0eb4c3dba1"),
  "username" : "GoodGuyGreg",
  "title" : "Reports a bug in your code",
  "Body" : "Sends you a Pull Request"
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.find({"username" : "GoodGuyGreg"}).pretty()
{
  "_id" : ObjectId("60a64f841c2f9c0eb4c3db9e"),
  "username" : "GoodGuyGreg",
  "title" : " Passes out at a party",
  "Body" : "Wakes up early and cleans house"
}
{
  "_id" : ObjectId("60a650261c2f9c0eb4c3db9f"),
  "username" : "GoodGuyGreg",
  "title" : " Passes out at a party",
  "Body" : "Wakes up early and cleans house"
}
{
  "_id" : ObjectId("60a650691c2f9c0eb4c3dba0"),
  "username" : "GoodGuyGreg",
  "title" : "Steals your identity",
  "Body" : "Raises your credit score"
}
{
  "_id" : ObjectId("60a650a91c2f9c0eb4c3dba1"),
  "username" : "GoodGuyGreg",
```

```
    "title" : "Borrows Something",
    "Body" : "Sells it"
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.insertOne({"username" : "GoodGuyGreg" ,"comment": "Hope you got a deal!", "post": ObjectId("60a6511a1c2f9c0eb4c3dba2")})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a656221c2f9c0eb4c3dba5")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.insertOne({"username" : "GoodGuyGreg" ,"comment": "What's mine is yours!", "post": ObjectId("60a6513f1c2f9c0eb4c3dba3")})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6567e1c2f9c0eb4c3dba6")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.insertOne({"username" : "GoodGuyGreg" ,"comment": "Don't violate licensing agreement!", "post": ObjectId("60a6516c1c2f9c0eb4c3dba4")})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a656e11c2f9c0eb4c3dba7")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.insertOne({"username" : "GoodGuyGreg" ,"comment": "It still isn't clean!", "post": ObjectId("60a650261c2f9c0eb4c3db9f")})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a657b21c2f9c0eb4c3dba8")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.insertOne({"username" : "ScumbagSteve" ,"comment": "It still isn't clean", "post": ObjectId("60a650261c2f9c0eb4c3db9f")})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a657f61c2f9c0eb4c3dba9")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.insertOne({"username" : "ScumbagSteve" ,"comment": "Denied your PR cause I found a hack", "post": ObjectId("60a650a91c2f9c0eb4c3dba1")})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a658621c2f9c0eb4c3dbaa")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.users.find().pretty()
{
  "_id" : ObjectId("60a4a60d4a251f11e4193da7"),
  "name" : "Bhushan Bire",
  "age" : 23
}
{
  "_id" : ObjectId("60a6388d1c2f9c0eb4c3db9b"),
  "username" : "GoodGuyGreg",
  "first_name" : "GoodGuy",
}
```



```

    },
    "operationTime" : Timestamp(1621511843, 7)
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insert({"username" : "GoodGuyGreg" ,"title":"
Passes out at a party", "Body": "Wakes up early and cleans house"})
WriteResult({ "nInserted" : 1 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insertOne({"username" : "GoodGuyGreg" ,"title
":" Passes out at a party", "Body": "Wakes up early and cleans house"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a650261c2f9c0eb4c3db9f")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insertOne({"username" : "GoodGuyGreg" ,"title
":"Steals your identity", "Body": "Raises your credit score"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a650691c2f9c0eb4c3dba0")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insertOne({"username" : "GoodGuyGreg" ,"title
":"Reports a bug in your code", "Body": "Sends you a Pull Request"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a650a91c2f9c0eb4c3dba1")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insertOne({"username" : "ScumbagSteve" ,"titl
e":"Borrows Something" , "Body": "Sells it"})
uncaught exception: SyntaxError: missing } after property list :
@(shell):1:50
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insertOne({"username" : "ScumbagSteve" , "tit
le":"Borrows Something" , "Body": "Sells it"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6511a1c2f9c0eb4c3dba2")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insertOne({"username" : "ScumbagSteve" , "tit
le":"Borrows Everything" , "Body": "The End"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6513f1c2f9c0eb4c3dba3")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insertOne({"username" : "ScumbagSteve" , "tit
le":"Forks your repo on github" , "Body": "Sets to private"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6516c1c2f9c0eb4c3dba4")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.find().pretty()
{
  "_id" : ObjectId("60a64f841c2f9c0eb4c3db9e"),

```



```

... )
uncaught exception: SyntaxError: expected property name, got '{' :
@(shell):1:20
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.users.insertOne({"username" : "GoodGuyGreg", "first_
_name" : "GoodGuy", "last_name" : "Greg"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a6388d1c2f9c0eb4c3db9b")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.users.insertOne({"username" : "SchumbagSteve" , "fu
ll_name" : {"first" : "Scumbag" , "last" : "Steve"}})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60a639451c2f9c0eb4c3db9c")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.users.insert({"username" : "SchumbagSteve" , "full_
name" : {"first" : "Scumbag" , "last" : "Steve"}})
WriteResult({ "nInserted" : 1 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> show dbs
admin          0.000GB
local          4.032GB
mongo_practice 0.000GB
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> use mongo_practice
switched to db mongo_practice
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.createCollection('posts')
{
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621511827, 7),
    "signature" : {
      "hash" : BinData(0,"0xzw7qviOXAMV1KBfQgV8xs6Zok="),
      "keyId" : NumberLong("6963585860331634691")
    }
  },
  "operationTime" : Timestamp(1621511827, 7)
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.createCollection('comments')
{
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621511843, 7),
    "signature" : {
      "hash" : BinData(0,"0vAE+qExwEEjxGQr1x8fxpq1Y8c="),
      "keyId" : NumberLong("6963585860331634691")
    }
  },
  "operationTime" : Timestamp(1621511843, 7)
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.posts.insert({"username" : "GoodGuyGreg" ,"title":

```

```
{ "_id" : ObjectId("60a4cade4a251f11e4193da7"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.remove({Title: "Avatar"})
WriteResult({ "nRemoved" : 1 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find().pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : "Samuel L. Jackson"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "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("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation 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 magic
```



```

    "Title" : "Pee Wee Herman's Big Adventure"
  }
}
{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.remove({Title: "Pee Wee Herman's Big Adventure"})
WriteResult({ "nRemoved" : 1 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find().pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : "Samuel L. Jackson"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "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("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkein",
  "year" : 2013,

```



```

    "_id" : ObjectId("60a4cade4a251f11e4193dac"),
    "Title" : "The Hobbit:The Dessolation 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 magic
al ring."
}
{
    "_id" : ObjectId("60a4cade4a251f11e4193dad"),
    "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 combantants a
nd keep the Lonely Mountain form falling into hands of a rising darkness."
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "gold" $and "dragon"}
})
uncaught exception: SyntaxError: missing } after property list :
@(shell):1:40
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find($and: [{ $text: { $search: "gold" $and "d
ragon" } } ] )
uncaught exception: SyntaxError: missing ) after argument list :
@(shell):1:19
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$and: [{ $text: { $search: "gold" $and "
dragon" } } ] } )
uncaught exception: SyntaxError: missing } after property list :
@(shell):1:48
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "gold","dragon"}})
uncaught exception: SyntaxError: missing : after property id :
@(shell):1:48
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "gold"},{$search: "dr
agon"}})
uncaught exception: SyntaxError: expected property name, got '{' :
@(shell):1:41
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "gold dragon"}}).pret
ty()
{
    "_id" : ObjectId("60a4cade4a251f11e4193dab"),
    "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 sprited
group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY>

```



```

    "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."
  }
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "dwarves hobbit"}}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "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("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation 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("60a4cade4a251f11e4193dad"),
  "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 hands of a rising darkness."
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "gold" $and "dragon"}})
uncaught exception: SyntaxError: missing } after property list :
@(shell):1:40
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find($and: [{ $text: { $search: "gold" $and "dragon" } }])
uncaught exception: SyntaxError: missing ) after argument list :
@(shell):1:19
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$and: [{ $text: { $search: "gold" $and "dragon" } } ]})
uncaught exception: SyntaxError: missing } after property list :
@(shell):1:48
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "gold","dragon"}})
uncaught exception: SyntaxError: missing : after property id :
@(shell):1:48
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "gold"},{$search: "dr

```



```

    "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."
  }
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "Gandalf"}}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation 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."
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "Bilbo -Gandalf"}}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193dad"),
  "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("60a4cade4a251f11e4193dab"),
  "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."
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "dwarves hobbit"}}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "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("60a4cade4a251f11e4193dac"),

```



```

}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "Bilbo"}})
{ "_id" : ObjectId("60a4cade4a251f11e4193dac"), "Title" : "The Hobbit:The Dessolation 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("60a4cade4a251f11e4193dad"), "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 hands of a rising darkness." }
{ "_id" : ObjectId("60a4cade4a251f11e4193dab"), "Title" : "The Hobbit:An Unexpected Journey", "Writer" : "J.R.R. Tolkien", "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." }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "Bilbo"}}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation 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("60a4cade4a251f11e4193dad"),
  "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 hands of a rising darkness."
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "Title" : "The Hobbit:An Unexpected Journey",
  "Writer" : "J.R.R. Tolkien",
  "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."
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text: {$search: "Gandalf"}}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkien",

```



```

    }
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.createIndex({ Title: "text", Synopses: "text" })
{
  "numIndexesBefore" : 2,
  "numIndexesAfter" : 2,
  "note" : "all indexes already exist",
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621500353, 6),
    "signature" : {
      "hash" : BinData(0,"4rqLZtQ1iT+KP5HnPv4xafuL/w8="),
      "keyId" : NumberLong("6963585860331634691")
    }
  },
  "operationTime" : Timestamp(1621500353, 6)
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.createIndex({ Title: "text"}, {Synopses: "text"})
{
  "operationTime" : Timestamp(1621500379, 4),
  "ok" : 0,
  "errmsg" : "Error in specification { key: { Title: \"text\" }, name: \"Title_text\", Synopses: \"text\" } :: caused by :: The field 'Synopses' is not valid for an index specification. Specification: { key: { Title: \"text\" }, name: \"Title_text\", Synopses: \"text\" }",
  "code" : 197,
  "codeName" : "InvalidIndexSpecificationOption",
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621500379, 4),
    "signature" : {
      "hash" : BinData(0,"D2HcFPy4gDYniDmTIR/rHv+WizU="),
      "keyId" : NumberLong("6963585860331634691")
    }
  }
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.createIndex({ Title: "text", Synopses: "text" })
{
  "numIndexesBefore" : 2,
  "numIndexesAfter" : 2,
  "note" : "all indexes already exist",
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621500386, 6),
    "signature" : {
      "hash" : BinData(0,"0dhGFny/5yxKr6jhPX0XQd29Wms="),
      "keyId" : NumberLong("6963585860331634691")
    }
  }
}

```



```

}
{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.update({Title: "Pulp Fiction"},{$set:{actors
:"Samuel L. Jackson"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find()
{ "_id" : ObjectId("60a4cade4a251f11e4193da8"), "Title" : "Fight Club", "Writer" : "Chuck Palahniuko",
"Year" : 1999, "actors" : [ "Brad Pitt", "Edward Norton" ] }
{ "_id" : ObjectId("60a4cade4a251f11e4193da9"), "Title" : "Pulp Fiction", "Writer" : "Quentin Tarantino
", "Year" : 1994, "actors" : "Samuel L. Jackson" }
{ "_id" : ObjectId("60a4cade4a251f11e4193daa"), "Title" : "Inglorious Basterds", "Writer" : "Quentin Ta
rantino", "Year" : 2009, "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }
{ "_id" : ObjectId("60a4cade4a251f11e4193dab"), "Title" : "The Hobbit:An Unexpected Journey", "Writer"
: "J.R.R. Tolkein", "Year" : 2012, "Franchise" : "The Hobbit", "Synopsis" : "A reluctant hobbit, Bilbo
Baggins,sets out to the Lonely Mountain with a spirited group of dwarves to reclaim their mountain home
- and the gold within it - from the dragon Smaug." }
{ "_id" : ObjectId("60a4cade4a251f11e4193dac"), "Title" : "The Hobbit:The Dessolation of Smaug", "Write
r" : "J.R.R. Tolkein", "year" : 2013, "Franchise" : "The Hobbit", "Synopsis" : "The dwarves, along with
Bilbo Baggins and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug
. Bilbo Baggins is in possession of a mysterious and magical ring." }
{ "_id" : ObjectId("60a4cade4a251f11e4193dad"), "Title" : "The Hobbit:The Battle of the Five Armies", "
Writer" : "J.R.R. Tolkein", "Year" : 2012, "Franchise" : "The Hobbit", "Synopsis" : "Bilbo and Company
are forced to engage in a war against an array of combantants and keep the Lonely Mountain form falling
into hands of a rising darkness." }
{ "_id" : ObjectId("60a4cade4a251f11e4193dae"), "Title" : "Pee Wee Herman's BIg Adventure" }
{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({$text:{$search: "Bilbo"}}).pretty()
Error: error: {
  "operationTime" : Timestamp(1621498530, 2),
  "ok" : 0,
  "errmsg" : "text index required for $text query",
  "code" : 27,
  "codeName" : "IndexNotFound",
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621498530, 2),
    "signature" : {
      "hash" : BinData(0,"QLuyKgRTLTtONPZfhW39cmX6J0c="),
      "keyId" : NumberLong("6963585860331634691")
    }
  }
}
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.createIndex({ Title: "text", Synopsis: "test
"})
{
  "operationTime" : Timestamp(1621499256, 6),
  "ok" : 0,
  "errmsg" : "Error in specification { key: { Title: \"text\", Synopsis: \"test\" }, name: \"Titl
e_text_Synopses_test\" } :: caused by :: Can't use more than one index plugin for a single index.",
  "code" : 67,

```



```

{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation 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("60a4cade4a251f11e4193dad"),
  "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 combantants and keep the Lonely Mountain form falling into hands of a rising darkness."
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dae"),
  "Title" : "Pee Wee Herman's BIG Adventure"
}
{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.update({Title: "Pulp Fiction"},{$set:{actors: "Samuel L. Jackson"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find()
{ "_id" : ObjectId("60a4cade4a251f11e4193da8"), "Title" : "Fight Club", "Writer" : "Chuck Palahniuko", "Year" : 1999, "actors" : [ "Brad Pitt", "Edward Norton" ] }
{ "_id" : ObjectId("60a4cade4a251f11e4193da9"), "Title" : "Pulp Fiction", "Writer" : "Quentin Tarantino", "Year" : 1994, "actors" : "Samuel L. Jackson" }
{ "_id" : ObjectId("60a4cade4a251f11e4193daa"), "Title" : "Inglorious Basterds", "Writer" : "Quentin Tarantino", "Year" : 2009, "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }
{ "_id" : ObjectId("60a4cade4a251f11e4193dab"), "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("60a4cade4a251f11e4193dac"), "Title" : "The Hobbit:The Dessolation 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("60a4cade4a251f11e4193dad"), "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 combantants and keep the Lonely Mountain form falling into hands of a rising darkness." }
{ "_id" : ObjectId("60a4cade4a251f11e4193dae"), "Title" : "Pee Wee Herman's BIG Adventure" }
{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY>

```



```
movies
users
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.update({Title: "The Hobbit:The Dessolation of
Smaug"},{$set:{Synopsis: "The dwarves, along with Bilbo Baggins and Gandalf the Grey, continue their qu
est to reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and ma
gical ring."}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find().pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "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 sprited g
roup of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."
}
{
```



```
movies
users
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.update({Title: "The Hobbit:The Dessolation of
Smaug"},{$set:{Synopsis: "The dwarves, along with Bilbo Baggins and Gandalf the Grey, continue their qu
est to reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and ma
gical ring."}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find().pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "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 sprited g
roup of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."
}
{
```

```
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> 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 group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 0 })
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find().pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "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("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkein",
}
```



```

    "Eli Roth"
  ]
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({Franchise: "The Hobbit"}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "Title" : "The Hobbit:An Unexpected Journey",
  "Writer" : "J.R.R. Tolkein",
  "Year" : 2012,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkein",
  "year" : 2013,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dad"),
  "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 combantants an
d keep the Lonely Mountain form falling into hands of a rising darkness."
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({"Year":{$gte:1990, $lt:2000}}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}

```

```

    ]
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({actors: "Brad Pitt"}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({Franchise: "The Hobbit"}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "Title" : "The Hobbit:An Unexpected Journey",
  "Writer" : "J.R.R. Tolkein",
  "Year" : 2012,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkein",
  "year" : 2013,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dad"),
  "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 combantants an
d keep the Lonely Mountain form falling into hands of a rising darkness."
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({"Year":{$gte:1990, $lt:2000}}).pretty()

```



```
{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({Writer: "Quentin Tarantino"})
{ "_id" : ObjectId("60a4cade4a251f11e4193da9"), "Title" : "Pulp Fiction", "Writer" : "Quentin Tarantino",
  "Year" : 1994, "actors" : [ "John Travolta", "Uma Thurman" ] }
{ "_id" : ObjectId("60a4cade4a251f11e4193daa"), "Title" : "Inglorious Basterds", "Writer" : "Quentin Tarantino",
  "Year" : 2009, "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({Writer: "Quentin Tarantino"}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({actors: "Brad Pitt"}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
```

```

{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "Title" : "The Hobbit:An Unexpected Journey",
  "Writer" : "J.R.R. Tolkein",
  "Year" : 2012,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkein",
  "year" : 2013,
  "franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dad"),
  "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 combantants and keep the Lonely Mountain form falling into hands of a rising darkness."
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dae"),
  "Title" : "Pee Wee Herman's BIG Adventure"
}
{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({Writer: "Quentin Tarantino"})
{ "_id" : ObjectId("60a4cade4a251f11e4193da9"), "Title" : "Pulp Fiction", "Writer" : "Quentin Tarantino",
  "Year" : 1994, "actors" : [ "John Travolta", "Uma Thurman" ] }
{ "_id" : ObjectId("60a4cade4a251f11e4193daa"), "Title" : "Inglorious Basterds", "Writer" : "Quentin Tarantino", "Year" : 2009, "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find({Writer: "Quentin Tarantino"}).pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,

```



```

{ "_id" : ObjectId("60a4cade4a251f11e4193daf"), "Title" : "Avatar" }
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.find().pretty()
{
  "_id" : ObjectId("60a4cade4a251f11e4193da8"),
  "Title" : "Fight Club",
  "Writer" : "Chuck Palahniuko",
  "Year" : 1999,
  "actors" : [
    "Brad Pitt",
    "Edward Norton"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193da9"),
  "Title" : "Pulp Fiction",
  "Writer" : "Quentin Tarantino",
  "Year" : 1994,
  "actors" : [
    "John Travolta",
    "Uma Thurman"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193daa"),
  "Title" : "Inglorious Basterds",
  "Writer" : "Quentin Tarantino",
  "Year" : 2009,
  "actors" : [
    "Brad Pitt",
    "Diane Kruger",
    "Eli Roth"
  ]
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dab"),
  "Title" : "The Hobbit:An Unexpected Journey",
  "Writer" : "J.R.R. Tolkein",
  "Year" : 2012,
  "Franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dac"),
  "Title" : "The Hobbit:The Dessolation of Smaug",
  "Writer" : "J.R.R. Tolkein",
  "year" : 2013,
  "franchise" : "The Hobbit"
}
{
  "_id" : ObjectId("60a4cade4a251f11e4193dad"),

```

```

MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.movies.insertMany([ {Title: "Fight Club", Writer: "C
huck Palahniuko", year: 1999, actors: ["Brad Pitt", "Edward Norton"]},{Title: "Pulp Fiction", Writer: "Q
uentin Tarantino", year: 1994, actors: ["John Travolta", "Uma Thurman"]}, {Title: "Inglorious Basterds",
  Writer: "Quentin Tarantino", year: 2009, actors: ["Brad Pitt", "Diane Kruger", "Eli Roth"]}, {Title: "T
he Hobbit:An Unexpected Journey", Writer: "J.R.R. Tolkein", year: 2012, franchise: "The Hobbit"}, {Title
: "The Hobbit:The Dessolation 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 combantants
and keep the Lonely Mountain form falling into hands of a rising darkness."}, {Title: "Pee Wee Herman's
Big Adventure"}, {Title: "Avatar"}])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("60a4cade4a251f11e4193da8"),
    ObjectId("60a4cade4a251f11e4193da9"),
    ObjectId("60a4cade4a251f11e4193daa"),
    ObjectId("60a4cade4a251f11e4193dab"),
    ObjectId("60a4cade4a251f11e4193dac"),
    ObjectId("60a4cade4a251f11e4193dad"),
    ObjectId("60a4cade4a251f11e4193dae"),
    ObjectId("60a4cade4a251f11e4193daf")
  ]
}

```



```

    "post" : ObjectId("60a6516c1c2f9c0eb4c3dba4")
  }
}
  "_id" : ObjectId("60a657b21c2f9c0eb4c3dba8"),
  "username" : "GoodGuyGreg",
  "comment" : "It still insn't clean!",
  "post" : ObjectId("60a650261c2f9c0eb4c3db9f")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"username" : "ScumbagSteve"}).pretty
()
{
  "_id" : ObjectId("60a657f61c2f9c0eb4c3dba9"),
  "username" : "ScumbagSteve",
  "comment" : "It still insn't clean",
  "post" : ObjectId("60a650261c2f9c0eb4c3db9f")
}
{
  "_id" : ObjectId("60a658621c2f9c0eb4c3dbaa"),
  "username" : "ScumbagSteve",
  "comment" : "Denied your PR cause I found a hack",
  "post" : ObjectId("60a650a91c2f9c0eb4c3dba1")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"title" : "Reports a bug in your code
"}).pretty()
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"title" : "Reports a bug in your cod
e"}).pretty()
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"title" : "Reports a bug in your cod
e"}).pretty()
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> db.comments.find({"post" : ObjectId("60a650a91c2f9c0eb
4c3dba1")})).pretty()
{
  "_id" : ObjectId("60a658621c2f9c0eb4c3dbaa"),
  "username" : "ScumbagSteve",
  "comment" : "Denied your PR cause I found a hack",
  "post" : ObjectId("60a650a91c2f9c0eb4c3dba1")
}
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY> show dbs
mongo_practice
admin
local
MongoDB Enterprise atlas-b9aweb-shard-0:PRIMARY>

```