

MONGODB BASIC ASSIGNMENTS

Lovely Professional ...

Access Manager

Billing

All Clusters

Get Help

UPENDRA

Project 0

Atlas

Realm

Charts

DEPLOYMENT

Databases

Data Lake

DATA SERVICES

Triggers

Data API

SECURITY

Quickstart

Database Access

Network Access

Advanced

LOVELY PROFESSIONAL UNIVERSITY > PROJECT 0 > DATABASES

Test

VERSION: 4.4.11

REGION: AWS Mumbai (ap-south-1)

OverviewMetricsCollectionsSearchCmd Line ToolsReal TimeProfilerPerformance AdvisorOnline Archive

DATABASES: 1COLLECTIONS: 4

+ Create Database

NAMESPACES

mongo_practise

comments

movies

posts

users

mongo_practise

DATABASE SIZE: 277KB

INDEX SIZE: 116KB

TOTAL COLLECTIONS: 4

CREATE COLLECTION

Collection Name	Documents	Documents Size	Documents Avg	Indexes	Index Size	Index Avg
comments	5	541B	109B	1	20KB	20KB
movies	6	1.4KB	240B	2	56KB	28KB
posts	6	663B	111B	1	20KB	20KB
users	2	193B	97B	1	20KB	20KB

query the movies collection to

1. get all documents

```
Select mongosh mongodb+srv://test:hfd7.mongodb.net/myFirstDatabase
true
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> 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.Tolkien",year:2012,franchise:"The Hobbit"},{title:"The Hobbit:The Desolation of Smaug",writer:"J.R.R.Tolkien",year:2013,franchise:"The Hobbit"},{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."},{title:"Pee Wee Herman's Big Adventure"},{title:"Avatar"}],)
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("61ec466c3e5bb1eddd12a7ce"),
    '1': ObjectId("61ec466c3e5bb1eddd12a7cf"),
    '2': ObjectId("61ec466c3e5bb1eddd12a7d0"),
    '3': ObjectId("61ec466c3e5bb1eddd12a7d1"),
    '4': ObjectId("61ec466c3e5bb1eddd12a7d2"),
    '5': ObjectId("61ec466c3e5bb1eddd12a7d3"),
    '6': ObjectId("61ec466c3e5bb1eddd12a7d4"),
    '7': ObjectId("61ec466c3e5bb1eddd12a7d5")
  }
}
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find()
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7ce"),
    title: 'Fight Club',
    writer: 'Chuck Palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7cf"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d0"),
    title: 'Inglorious Basterds',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diane Kruger', 'eli Roth' ]
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d1"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R.Tolkien',
    year: 2012,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d2"),
    title: 'The Hobbit:The Desolation of Smaug',
    writer: 'J.R.R.Tolkien',
    year: 2013,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d3"),
    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.'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d4"),
    title: 'Pee Wee Herman's Big Adventure'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d5"), title: 'Avatar' }
]
```

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

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

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

```

Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({writer:"Quentin Tarantino"})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7cf"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d0"),
    title: 'Inglorious Basterds',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diane Kruger', 'eli Roth' ]
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({actors:"Brad Pitt"})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7ce"),
    title: 'Fight Club',
    writer: 'Chuck Palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d0"),
    title: 'Inglorious Basterds',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diane Kruger', 'eli Roth' ]
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({franchise:"The Hobbit"})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d1"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R.Tolkein',
    year: 2012,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d2"),
    title: 'The Hobbit:The Desolation of Smaug',

```

5. get all movies released in the 90s

```
mongosh mongodb+srv://test.hftd7.mongodb.net/myFirstDatabase
```

```
[
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({year:{$gte:1900,$lt:2000}})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7ce"),
    title: 'Fight Club',
    writer: 'Chuck Palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7cf"),
    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

```
mongosh mongodb+srv://test.hftd7.mongodb.net/myFirstDatabase
```

```
[
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({$or:[{year:{$lte:2000}},{year:{$gte:2010}]}})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7ce"),
    title: 'Fight Club',
    writer: 'Chuck Palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7cf"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d1"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R.Tolkein',
    year: 2012,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d2"),
    title: 'The Hobbit:The Desolation of Smaug',
    writer: 'J.R.R.Tolkein',
    year: 2013,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d3"),
    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.'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
franchise: 'The Hobbit'
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d3"),
    title: 'The Hobbit:The Battle of the Five Armies',
    writer: 'J.R.R.Tolkein',
    year: 2012,
    franchise: 'The Hobbit',
    Synopsis: 'Bilbo and Company are forced to engage in a war against an array of combatants and keep the Lonely Mountain from falling into the hands of a rising darkness.'
  }
]
```

Update 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."

```
mongosh mongodb+srv://test.hfd7.mongodb.net/myFirstDatabase
Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

C:\Users\my pc>mongosh "mongodb+srv://test.hfd7.mongodb.net/myFirstDatabase" --username upendrarambarki --password Haveenup143
Current Mongosh log ID: 61ed343362681edcd183568
Connecting to:      mongodb+srv://test.hfd7.mongodb.net/myFirstDatabase?appName=mongosh+1.1.9
Using MongoDB:      4.4.11
Using Mongosh:      1.1.9

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

Warning: Found ~/.mongorc.js, but not ~/.mongoshrc.js. ~/.mongorc.js will not be loaded.
You may want to copy or rename ~/.mongorc.js to ~/.mongoshrc.js.
Atlas atlas-w4167a-shard-0 [primary] myFirstDatabase> use mongo_practise
switched to db mongo_practise
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> 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."}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
```

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."

```
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> 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 and magical ring."}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
```

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

```
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.update((title:"Pulp Fiction"),{$push:{actor:"Samuel L.Jackson"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
```

Text Search 1.

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

```
mongosh mongodb+srv://test:hfd7.mongodb.net/myFirstDatabase
> 1 | db.movies.update([{$title:"Pulp Fiction"},{$push:{actor:"Samuel L.Jackson"}})
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.createIndex({Synopsis:"text"})
Synopsis.text
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({$text:{$search:"Bilbo"}})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d2"),
    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("61ec466c3e5bb1eddd12a7d3"),
    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.'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d1"),
    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.'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({$text:{$search:"Gandalf"}})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d2"),
    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.'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
```

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

```
mongosh mongodb+srv://test:hfd7.mongodb.net/myFirstDatabase
title: 'The Hobbit:The Desolation of Smaug',
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({$text:{$search:"Bilbo -Gandalf"}})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d3"),
    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.'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d1"),
    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.'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
```

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

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

```
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({$text:{$search:"dwarves hobbit"}})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d1"),
    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.'
  },
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d2"),
    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.'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.find({$text:{$search:"gold""dragon"}})
[
  {
    _id: ObjectId("61ec466c3e5bb1eddd12a7d1"),
    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.'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
```

Delete Documents

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

2. delete the movie "Avatar"

```
mongosh mongodb+srv://test.hftd7.mongodb.net/myFirstDatabase
```

```
}  
]  
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.remove({title:"Pee Wee Herman's Big Adventure"})  
DeprecationWarning: Collection.remove() is deprecated. Use deleteOne, deleteMany, findOneAndDelete, or bulkWrite.  
{ acknowledged: true, deletedCount: 1 }  
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.movies.remove({title:"Avatar"})  
{ acknowledged: true, deletedCount: 1 }  
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> _
```

Querying related collections

1. find all users

```
mongosh mongodb+srv://test.hftd7.mongodb.net/myFirstDatabase
```

```
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.users.find()  
[  
  {  
    _id: ObjectId("61ed58932df233a9a2194300"),  
    username: 'GoodGuyGreg',  
    first_name: 'Good Guy',  
    last_name: 'Greg'  
  },  
  {  
    _id: ObjectId("61ed58932df233a9a2194301"),  
    username: 'ScumbagSteve',  
    full_name: { first: 'Scumbag', last: 'Steve' }  
  }  
]  
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> _
```

2. find all posts

```

Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.posts.find()
[
  {
    _id: ObjectId("61ed4eb02df233a9a21942f5"),
    username: 'GoodGuyGreg',
    title: 'Passes out at party',
    body: 'Wakes up early and cleans house'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942f6"),
    username: 'GoodGuyGreg',
    title: 'Steals your identity',
    body: 'Raises your credit score'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942f7"),
    username: 'GoodGuyGreg',
    title: 'Reports a bug in your code',
    body: 'Sends you a Pull Request'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942f8"),
    username: 'ScumbagSteve',
    title: 'Borrows something',
    body: 'Sells it'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942f9"),
    username: 'ScumbagSteve',
    title: 'Borrows everything',
    body: 'The end'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942fa"),
    username: 'ScumbagSteve',
    title: 'Forks your repo on github',
    body: 'Sets to private'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>

```

3. find all posts that was authored by "GoodGuyGreg"

mongosh mongodb+srv://test.hftd7.mongodb.net/myFirstDatabase

```
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.posts.find({username:"GoodGuyGreg"})
[
  {
    _id: ObjectId("61ed4eb02df233a9a21942f5"),
    username: 'GoodGuyGreg',
    title: 'Passes out at party',
    body: 'Wakes up early and cleans house'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942f6"),
    username: 'GoodGuyGreg',
    title: 'Steals your identity',
    body: 'Raises your credit score'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942f7"),
    username: 'GoodGuyGreg',
    title: 'Reports a bug in your code',
    body: 'Sends you a Pull Request'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
```

4. find all posts that was authored by "ScumbagSteve"

mongosh mongodb+srv://test.hftd7.mongodb.net/myFirstDatabase

```
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.posts.find({username:"ScumbagSteve"})
[
  {
    _id: ObjectId("61ed4eb02df233a9a21942f8"),
    username: 'ScumbagSteve',
    title: 'Borrows something',
    body: 'Sells it'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942f9"),
    username: 'ScumbagSteve',
    title: 'Borrows everything',
    body: 'The end'
  },
  {
    _id: ObjectId("61ed4eb02df233a9a21942fa"),
    username: 'ScumbagSteve',
    title: 'Forks your repo on github',
    body: 'Sets to private'
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> █
```

5. find all comments

 mongosh mongodb+srv://test.hftd7.mongodb.net/myFirstDatabase

```
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.comments.find()
[
  {
    _id: ObjectId("61ed65392df233a9a2194302"),
    username: 'GoodGuyGreg',
    comment: 'Hope you got a good deal!',
    post: ObjectId("61ed4eb02df233a9a21942f8")
  },
  {
    _id: ObjectId("61ed65392df233a9a2194303"),
    username: 'GoodGuyGreg',
    comment: "What's mine is yours!",
    post: ObjectId("61ed4eb02df233a9a21942f9")
  },
  {
    _id: ObjectId("61ed65392df233a9a2194304"),
    username: 'GoodGuyGreg',
    comment: "Don't violate the licensing aggrement!",
    post: ObjectId("61ed4eb02df233a9a21942fa")
  },
  {
    _id: ObjectId("61ed65392df233a9a2194305"),
    username: 'ScumbagSteve',
    comment: "it still isn't clean",
    post: ObjectId("61ed4eb02df233a9a21942f5")
  },
  {
    _id: ObjectId("61ed65392df233a9a2194306"),
    username: 'ScumbagSteve',
    comment: 'Denied your PR cause I found a hack',
    post: ObjectId("61ed4eb02df233a9a21942f7")
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>
```

6. find all comments that was authored by "GoodGuyGreg"
7. find all comments that was authored by "ScumbagSteve"

```

Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.comments.find({username:"GoodGuyGreg"})
[
  {
    _id: ObjectId("61ed65392df233a9a2194302"),
    username: 'GoodGuyGreg',
    comment: 'Hope you got a good deal!',
    post: ObjectId("61ed4eb02df233a9a21942f8")
  },
  {
    _id: ObjectId("61ed65392df233a9a2194303"),
    username: 'GoodGuyGreg',
    comment: "What's mine is yours!",
    post: ObjectId("61ed4eb02df233a9a21942f9")
  },
  {
    _id: ObjectId("61ed65392df233a9a2194304"),
    username: 'GoodGuyGreg',
    comment: "Don't violate the licensing agreement!",
    post: ObjectId("61ed4eb02df233a9a21942fa")
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.comments.find({username:"ScumbagSteve"})
[
  {
    _id: ObjectId("61ed65392df233a9a2194305"),
    username: 'ScumbagSteve',
    comment: "it still isn't clean",
    post: ObjectId("61ed4eb02df233a9a21942f5")
  },
  {
    _id: ObjectId("61ed65392df233a9a2194306"),
    username: 'ScumbagSteve',
    comment: 'Denied your PR cause I found a hack',
    post: ObjectId("61ed4eb02df233a9a21942f7")
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise> _

```

8. find all comments belonging to the post "Reports a bug in your code"

```

Atlas atlas-w4167a-shard-0 [primary] mongo_practise> db.comments.find({post: db.posts.findOne({title: "Reports a bug in your code"})._id})
[
  {
    _id: ObjectId("61ed65392df233a9a2194306"),
    username: 'ScumbagSteve',
    comment: 'Denied your PR cause I found a hack',
    post: ObjectId("61ed4eb02df233a9a21942f7")
  }
]
Atlas atlas-w4167a-shard-0 [primary] mongo_practise>

```