

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

```
Atlas atlas-pjfls6-shard-0 [primary] person> use ashu
switched to db ashu
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.createCollection("movies")
{ ok: 1 }
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.insert([
... {
..... title:"fight club",
..... writer: "chuck palahniuko",
..... year:1999,
..... actors:["Brad Pitt","Edward Norton"],
..... },
... {
..... title:"Pulp Club",
..... writer: "chuck palahniuko",
..... year :1994,
..... actors:["John Travolta","Uma Thurman"],
..... },
... {
..... title:"Inglorious Bastard",
..... writer:"Quentin Tarantino",
..... year:2009,
..... actors:["Brad Pitt", "Diana Kruger","Eli Roth"],
..... },
... {
..... title:"The Hobbit: An Unexpected Journey",
..... writer: "J.R.R Tolken",
..... year:2012,
..... franchise:"The Hobbit",
..... }
])
```

```

..... title:"The Hobbit: The Battle of the Five Armies",
..... writer: "J.R.R Tolken",
..... year:2013,
..... franchise:"The Hobbit",
..... },
..... {
..... title:"The Hobbit: The Desolation of Smaug",
..... writer: "J.R.R Tolken",
..... year:2013,
..... franchise:"The Hobbit",
..... },
..... {
..... title:"Pee Wee Herman's Big Adventure",
..... },
..... {
..... title:"Avtar",
..... })
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("61eee1888565a0f0188d453f"),
    '1': ObjectId("61eee1888565a0f0188d4540"),
    '2': ObjectId("61eee1888565a0f0188d4541"),
    '3': ObjectId("61eee1888565a0f0188d4542"),
    '4': ObjectId("61eee1888565a0f0188d4543"),
    '5': ObjectId("61eee1888565a0f0188d4544"),
    '6': ObjectId("61eee1888565a0f0188d4545"),
    '7': ObjectId("61eee1888565a0f0188d4546")
  }
}

```

1. get all documents

```
Atlas atlas-pjf1s6-shard-0 [primary] mongo_practice> db.movies.find().pretty()
[
  {
    _id: ObjectId("61eee1888565a0f0188d453f"),
    title: 'fight club',
    writer: 'chuck palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4540"),
    title: 'Pulp Club',
    writer: 'chuck palahniuko',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4541"),
    title: 'Inglorious Bastard',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diana Kruger', 'Eli Roth' ]
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4542"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolken',
    year: 2012,
    franchise: 'The Hobbit'
  },
]
```

```

    },
    {
      _id: ObjectId("61eee1888565a0f0188d4543"),
      title: 'The Hobbit: The Battle of the Five Armies',
      writer: 'J.R.R Tolken',
      year: 2013,
      franchise: 'The Hobbit'
    },
    {
      _id: ObjectId("61eee1888565a0f0188d4544"),
      title: 'The Hobbit: The Desolation of Smaug',
      writer: 'J.R.R Tolken',
      year: 2013,
      franchise: 'The Hobbit'
    },
    {
      _id: ObjectId("61eee1888565a0f0188d4545"),
      title: "Pee Wee Herman's Big Adventure"
    },
    { _id: ObjectId("61eee1888565a0f0188d4546"), title: 'Avatar' }
  ]
Atlas atlas-pjf1s6-shard-0 [primary] mongo_practice>

```

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

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

```

Atlas atlas-pjf1s6-shard-0 [primary] mongo_practice> db.movies.find({writer: "Quentin Tarantino"})
[
  {
    _id: ObjectId("61eee1888565a0f0188d4541"),
    title: 'Inglorious Bastard',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diana Kruger', 'Eli Roth' ]
  }
]
Atlas atlas-pjf1s6-shard-0 [primary] mongo_practice> db.movies.find({actors: "Brad Pitt"})
[
  {
    _id: ObjectId("61eee1888565a0f0188d453f"),
    title: 'fight club',
    writer: 'chuck palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4541"),
    title: 'Inglorious Bastard',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diana Kruger', 'Eli Roth' ]
  }
]

```

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

```
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({franchise: "The Hobbit"})
[
  {
    _id: ObjectId("61eee1888565a0f0188d4542"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolken',
    year: 2012,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4543"),
    title: 'The Hobbit: The Battle of the Five Armies',
    writer: 'J.R.R Tolken',
    year: 2013,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4544"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R Tolken',
    year: 2013,
    franchise: 'The Hobbit'
  }
]
```

5. get all movies released in the 90s

```
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({$and:[{year:{$gte:1990}},{year:{$lt:2000}]}})
[
  {
    _id: ObjectId("61eee1888565a0f0188d453f"),
    title: 'fight club',
    writer: 'chuck palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4540"),
    title: 'Pulp Club',
    writer: 'chuck palahniuko',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  }
]
```

```
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice>
```

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

```
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({$or:[{year:{$lt:2000}},{year:{$gt:2010}}]})
[
  {
    _id: ObjectId("61eee1888565a0f0188d453f"),
    title: 'fight club',
    writer: 'chuck palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4540"),
    title: 'Pulp Club',
    writer: 'chuck palahniuko',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4542"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolken',
    year: 2012,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4543"),
    title: 'The Hobbit: The Battle of the Five Armies',
    writer: 'J.R.R Tolken',
    year: 2013,
    franchise: 'The Hobbit'
  }
]
```

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."
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-pjfls6-shard-0 [primary] mongo_practice> db.movies.update({ 'title' : "The Hobbit: An Unexpected Journey"},{$set:{'synopsis':"A reluctant hobbit,Bilbo Baggins,sets out to the lonely Mountain with a spirited group of dwarves to reclaim their mountain home -and the gold within it -from the dragon Smaug"}})
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.update({ 'title' : "The Hobbit: The Desolation of Smaug"},{$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 and magical ring."}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.update({ 'title' : "Pulp Club"},{$set:{'title':"Pulp Fiction"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

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

```

Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.update({title:"Pulp Fiction"},{$push:{actors:"Samuel L. Jackson"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({'title':"Pulp Fiction"})
[
  {
    _id: ObjectId("61eee1888565a0f0188d4540"),
    title: 'Pulp Fiction',
    writer: 'chuck palahniuko',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman', 'Samuel L. Jackson' ]
  }
]
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice>

```

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"


```

Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({$text:{$search:"Bilbo"}}).pretty()
[
  {
    _id: ObjectId("61eee1888565a0f0188d4544"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R Tolken',
    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("61eee1888565a0f0188d4542"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolken',
    year: 2012,
    franchise: 'The Hobbit',
    synopsis: 'A reluctant hobbit,Bilbo Baggins,sets out to the lonely Mountain with a spirited ground of dwarves to reclaim their mountain home -and the gold within it -from the dragon Smaug'
  }
]
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.createIndex({synopsis:"text2"})
MongoServerError: Error in specification { key: { synopsis: "text2" }, name: "synopsis_text2" } :: caused by :: Unknown index plugin 'text2'
Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({$text:{$search:"Gandalf"}}).pretty()
[
  {
    _id: ObjectId("61eee1888565a0f0188d4544"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R Tolken',
    year: 2013,
    franchise: 'The Hobbit',
    synopsis: 'The dwarves, along with Bilbo Baggins and Gandalf the Grey,continue their quest to reclaim Erebor , their homeland, from Smaug.Bilbo Baggins is in possession of a mysterious and magical ring.'
  }
]

```

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

```

Atlas atlas-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({$text:{$search:"Bilbo -Gandalf"}}).pretty()
[
  {
    _id: ObjectId("61eee1888565a0f0188d4542"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolken',
    year: 2012,
    franchise: 'The Hobbit',
    synopsis: 'A reluctant hobbit,Bilbo Baggins,sets out to the lonely Mountain with a spirited ground of dwarves to reclaim their mountain home -and the gold within it -from the dragon Smaug'
  }
]

```

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-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({$text:{$search:"dwarves hobbit"}}).pretty()
[
  {
    _id: ObjectId("61eee1888565a0f0188d4542"),
    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 with
thin it -from the dragon Smaug'
  },
  {
    _id: ObjectId("61eee1888565a0f0188d4544"),
    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-pjfls6-shard-0 [primary] mongo_practice> db.movies.find({$text:{$search:"gold dragon"}}).pretty()
[
  {
    _id: ObjectId("61eee1888565a0f0188d4542"),
    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 with
thin it -from the dragon Smaug'
  }
]
```

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

```
Atlas atlas-pjfls6-shard-0 [primary] myFirstDatabase> db.movies.deleteOne({title:"Pee Wee Herman's Big Adventure"})
{ acknowledged: true, deletedCount: 0 }
Atlas atlas-pjfls6-shard-0 [primary] myFirstDatabase> db.movies.deleteOne({title:"Avatar"})
{ acknowledged: true, deletedCount: 0 }
Atlas atlas-pjfls6-shard-0 [primary] myFirstDatabase>
```

```
switched to db users
Atlas atlas-pjfls6-shard-0 [primary] users> db.users.insert({ username:"GoodGuyGerg", first_name : "Good Guy", last_name : "Greg", username: "ScumbagSteve", full_name:[{first:"Scumbag",last:"Steve"}]})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61f0184a95c222b6b7abc86a") }
}
```

```
switched to db posts
Atlas atlas-pjfls6-shard-0 [primary] posts> db.post.insertMany([{ username: "GoodGuyGerg", title: "Passes out at party", body: "Wakes up early and cleans house" }, { username: "GoodGuyGerg", title: "Steals your identity", body: "Raise your credit score" }, { username: "GoodGuyGerg", title: "Reports a bug in your code", body: "Sends you a Pull Request" }, { username: "ScumbagSteve", title: "Borrows Something", body: "Sells it" }, { username: "ScumbagSteve", title: "Forks Your repo on github", body: "Sets to private" }, { username: "ScumbagSteve", title: "Borrows everything", body: "The end" }])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("61f018b595c222b6b7abc86b"),
    '1': ObjectId("61f018b595c222b6b7abc86c"),
    '2': ObjectId("61f018b595c222b6b7abc86d"),
    '3': ObjectId("61f018b595c222b6b7abc86e"),
    '4': ObjectId("61f018b595c222b6b7abc86f"),
    '5': ObjectId("61f018b595c222b6b7abc870")
  }
}
```

```

Atlas atlas-pjf1s6-shard-0 [primary] comments> db.comments.insertMany([
  { username: "GoodGuyGreg", comments: "Hope you got a good deal!", post: [ObjectId("61f018b595c222b6b7abc86e")] },
  { username: "GoodGuyGreg", comments: "Whats's mine is yours", post: [ObjectId("61f018b595c222b6b7abc870")] },
  { username: "GoodGuyGreg", comments: "Dont't violate the licensing agreement!", post: [ObjectId("61f018b595c222b6b7abc86f")] },
  { username: "ScumbagSteve", comments: "It still isn't clean", post: [ObjectId("61f018b595c222b6b7abc86b")] },
  { username: "ScumbagSteve", comments: "Denied your PR cause i found a hack", post: [ObjectId("61f018b595c222b6b7abc86d")] }])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("61f01ecb95c222b6b7abc871"),
    '1': ObjectId("61f01ecb95c222b6b7abc872"),
    '2': ObjectId("61f01ecb95c222b6b7abc873"),
    '3': ObjectId("61f01ecb95c222b6b7abc874"),
    '4': ObjectId("61f01ecb95c222b6b7abc875")
  }
}

```

1. find all users

```

Atlas atlas-pjf1s6-shard-0 [primary] post> use users
switched to db users
Atlas atlas-pjf1s6-shard-0 [primary] users> db.users.find()
[
  {
    _id: ObjectId("61f001550639626311bacef9"),
    username: 'ScumbagSteve',
    first_name: 'Good Guy',
    last_name: 'Greg',
    full_name: [ { first: 'Scumbag', last: 'Steve' } ]
  },
  {

```

2. find all posts

```
Atlas atlas-pjfls6-shard-0 [primary] post> db.post.find()
[
  {
    _id: ObjectId("61f006b30639626311bacefa"),
    username: 'GoodGuyGerg',
    title: 'Passes out at party',
    body: 'Wakes up early and cleans house'
  },
  {
    _id: ObjectId("61f006b30639626311bacefb"),
    username: 'GoodGuyGerg',
    title: 'Steals your identity',
    body: 'Raise your credit score'
  },
  {
    _id: ObjectId("61f006b30639626311bacefc"),
    username: 'GoodGuyGerg',
    title: 'Reports a bug in your code',
    body: 'Sends you a Pull Request'
  },
  {
    _id: ObjectId("61f006b30639626311bacefd"),
    username: 'ScumbagSteve',
    title: 'Borrows Something',
    body: 'Sells it'
  }
]
```

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

```
{
  _id: ObjectId("61f006b30639626311bacefe"),
  username: 'ScumbagSteve',
  title: 'Forks Your repo on github',
  body: 'Sets to private'
},
{
  _id: ObjectId("61f006b30639626311baceff"),
  username: 'ScumbagSteve',
  title: 'Borrows everything',
  body: 'The end'
}
}

Atlas atlas-pjfls6-shard-0 [primary] post> db.post.find({username:"GoodGuyGerg"})
[
  {
    _id: ObjectId("61f006b30639626311bacefb"),
    username: 'GoodGuyGerg',
    title: 'Steals your identity',
    body: 'Raise your credit score'
  },
  {
    _id: ObjectId("61f006b30639626311bacefc"),
    username: 'GoodGuyGerg',
    title: 'Reports a bug in your code',
    body: 'Sends you a Pull Request'
  }
]
```

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

```
Atlas atlas-pjf1s6-shard-0 [primary] post> db.post.find({username:"ScumbagSteve"})
[
  {
    _id: ObjectId("61f006b30639626311bacefd"),
    username: 'ScumbagSteve',
    title: 'Borrows Something',
    body: 'Sells it'
  },
  {
    _id: ObjectId("61f006b30639626311bacefe"),
    username: 'ScumbagSteve',
    title: 'Forks Your repo on github',
    body: 'Sets to private'
  },
  {
    _id: ObjectId("61f006b30639626311baceff"),
    username: 'ScumbagSteve',
    title: 'Borrows everything',
    body: 'The end'
  }
]
```

5. find all comments

```
Atlas atlas-pjf1s6-shard-0 [primary] post> use comments
switched to db comments
Atlas atlas-pjf1s6-shard-0 [primary] comments> db.comments.find()
[
  {
    _id: ObjectId("61f01ecb95c222b6b7abc871"),
    username: 'GoodGuyGreg',
    comments: 'Hope you got a good deal!',
    post: [ ObjectId("61f018b595c222b6b7abc86e") ]
  },
  {
    _id: ObjectId("61f01ecb95c222b6b7abc872"),
    username: 'GoodGuyGreg',
    comments: "Whats's mine is yours",
    post: [ ObjectId("61f018b595c222b6b7abc870") ]
  },
  {
    _id: ObjectId("61f01ecb95c222b6b7abc873"),
    username: 'GoodGuyGreg',
    comments: "Dont't violate the licensing agreement!",
    post: [ ObjectId("61f018b595c222b6b7abc86f") ]
  },
  {
    _id: ObjectId("61f01ecb95c222b6b7abc874"),
    username: 'ScumbagSteve',
    comments: "It still isn't clean",
    post: [ ObjectId("61f018b595c222b6b7abc86b") ]
  },
]
```

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

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

```
_id: ObjectId("61f01ecb95c222b6b7abc875"),
username: 'ScumbagSteve',
comments: 'Denied your PR cause i found a hack',
post: [ ObjectId("61f018b595c222b6b7abc86d") ]
}
]
Atlas atlas-pjfls6-shard-0 [primary] comments> db.comments.find({username:"ScumbagSteve"})
[
  {
    _id: ObjectId("61f01ecb95c222b6b7abc874"),
    username: 'ScumbagSteve',
    comments: "It still isn't clean",
    post: [ ObjectId("61f018b595c222b6b7abc86b") ]
  },
  {
    _id: ObjectId("61f01ecb95c222b6b7abc875"),
    username: 'ScumbagSteve',
    comments: 'Denied your PR cause i found a hack',
    post: [ ObjectId("61f018b595c222b6b7abc86d") ]
  }
]
```

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

```
Atlas atlas-pjfls6-shard-0 [primary] comments> db.comments.find({post:db.post.findOne({title:"Reports a bug in your code"})._id})
```