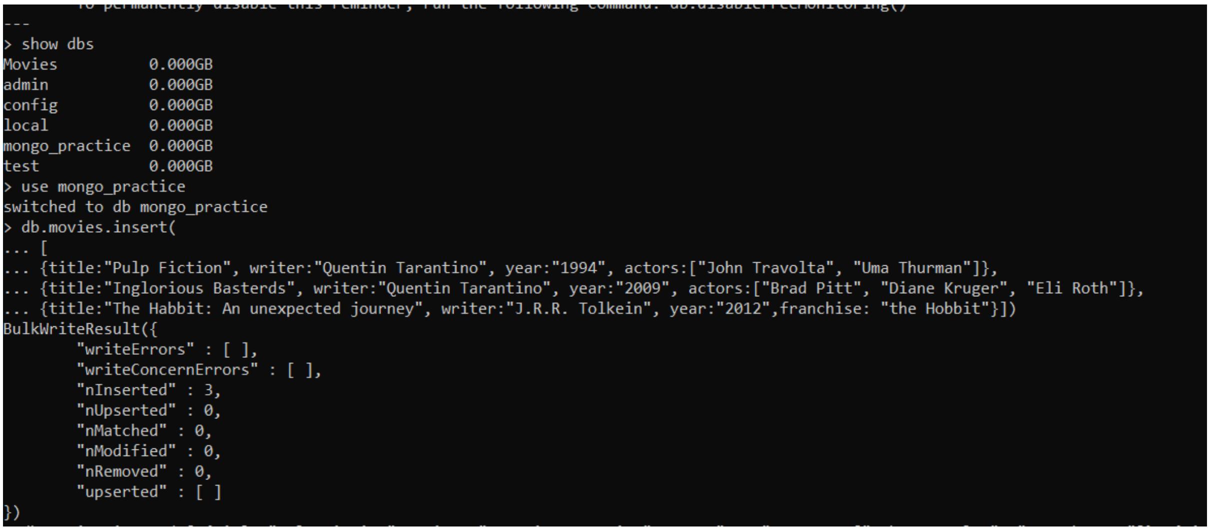
Assignment 1

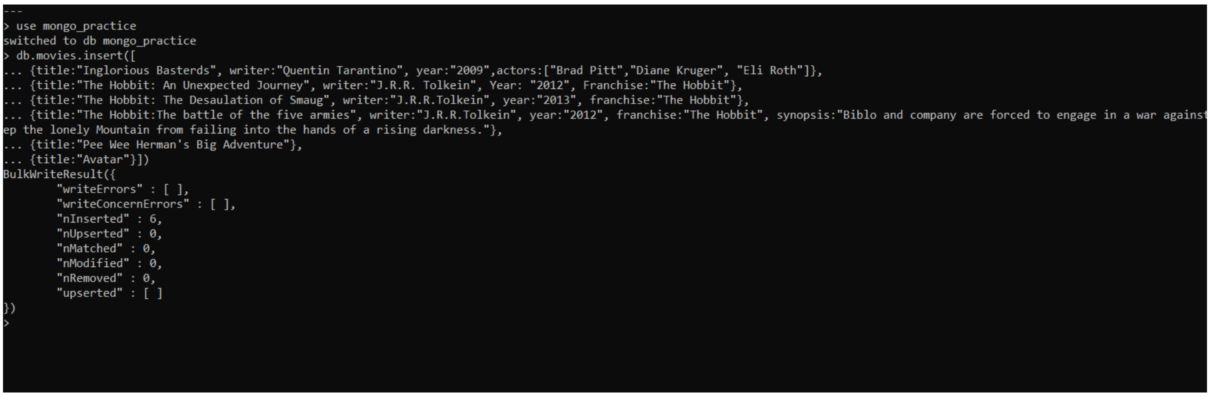
Queries on Find Document

1



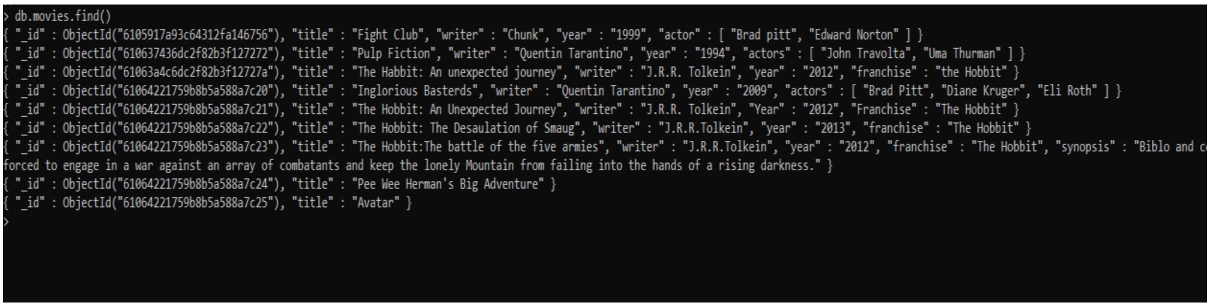
```
        To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
> show dbs
Movies          0.000GB
admin           0.000GB
config          0.000GB
local           0.000GB
mongo_practice  0.000GB
test            0.000GB
> use mongo_practice
switched to db mongo_practice
> db.movies.insert(
... [
... {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 Habbit: An unexpected journey", writer:"J.R.R. Tolkein", year:"2012",franchise: "the Hobbit"}])
BulkWriteResult({
        "writeErrors" : [ ],
        "writeConcernErrors" : [ ],
        "nInserted" : 3,
        "nUpserted" : 0,
        "nMatched" : 0,
        "nModified" : 0,
        "nRemoved" : 0,
        "upserted" : [ ]
})
```

2



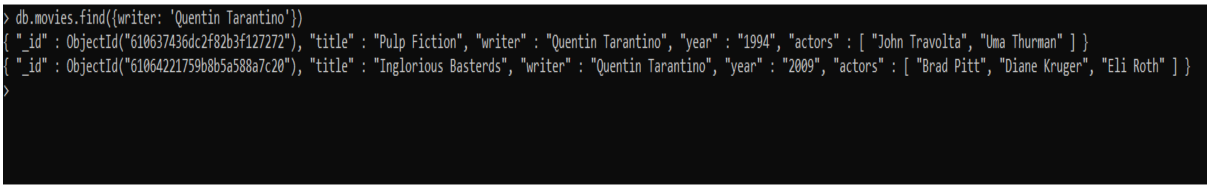
```
---
> use mongo_practice
switched to db mongo_practice
> db.movies.insert([
... {title:"Inglorious Basterds", writer:"Quentin Tarantino", year:"2009",actors:["Brad Pitt","Diane Kruger", "Eli Roth"]},
... {title:"The Hobbit: An Unexpected Journey", writer:"J.R.R. Tolkein", Year: "2012", Franchise:"The Hobbit"},
... {title:"The Hobbit: The Desaulation 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:"Biblo and company are forced to engage in a war against
ep the lonely Mountain from failing into the hands of a rising darkness."},
... {title:"Pee Wee Herman's Big Adventure"},
... {title:"Avatar"}])
BulkWriteResult({
        "writeErrors" : [ ],
        "writeConcernErrors" : [ ],
        "nInserted" : 6,
        "nUpserted" : 0,
        "nMatched" : 0,
        "nModified" : 0,
        "nRemoved" : 0,
        "upserted" : [ ]
})
>
```

3.Get All Documents



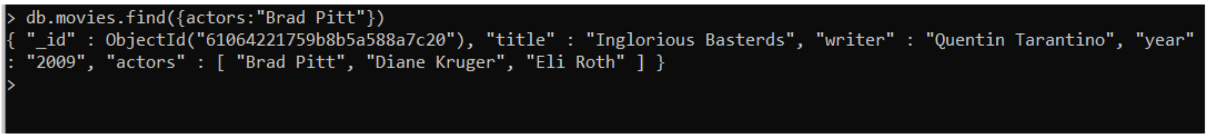
```
> db.movies.find()
{ "_id" : ObjectId("6105917a93c64312fa146756"), "title" : "Fight Club", "writer" : "Chunk", "year" : "1999", "actor" : [ "Brad pitt", "Edward Norton" ] }
{ "_id" : ObjectId("610637436dc2f82b3f127272"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : "1994", "actors" : [ "John Travolta", "Uma Thurman" ] }
{ "_id" : ObjectId("61063a4c6dc2f82b3f12727a"), "title" : "The Habbit: An unexpected journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "the Hobbit" }
{ "_id" : ObjectId("61064221759b8b5a588a7c20"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }
{ "_id" : ObjectId("61064221759b8b5a588a7c21"), "title" : "The Hobbit: An Unexpected Journey", "writer" : "J.R.R. Tolkein", "Year" : "2012", "Franchise" : "The Hobbit" }
{ "_id" : ObjectId("61064221759b8b5a588a7c22"), "title" : "The Hobbit: The Desaulation of Smaug", "writer" : "J.R.R.Tolkein", "year" : "2013", "franchise" : "The Hobbit" }
{ "_id" : ObjectId("61064221759b8b5a588a7c23"), "title" : "The Hobbit:The battle of the five armies", "writer" : "J.R.R.Tolkein", "year" : "2012", "franchise" : "The Hobbit", "synopsis" : "Biblo and c
forced to engage in a war against an array of combatants and keep the lonely Mountain from failing into the hands of a rising darkness." }
{ "_id" : ObjectId("61064221759b8b5a588a7c24"), "title" : "Pee Wee Herman's Big Adventure" }
{ "_id" : ObjectId("61064221759b8b5a588a7c25"), "title" : "Avatar" }
>
```

4.All documents of writer Quentin



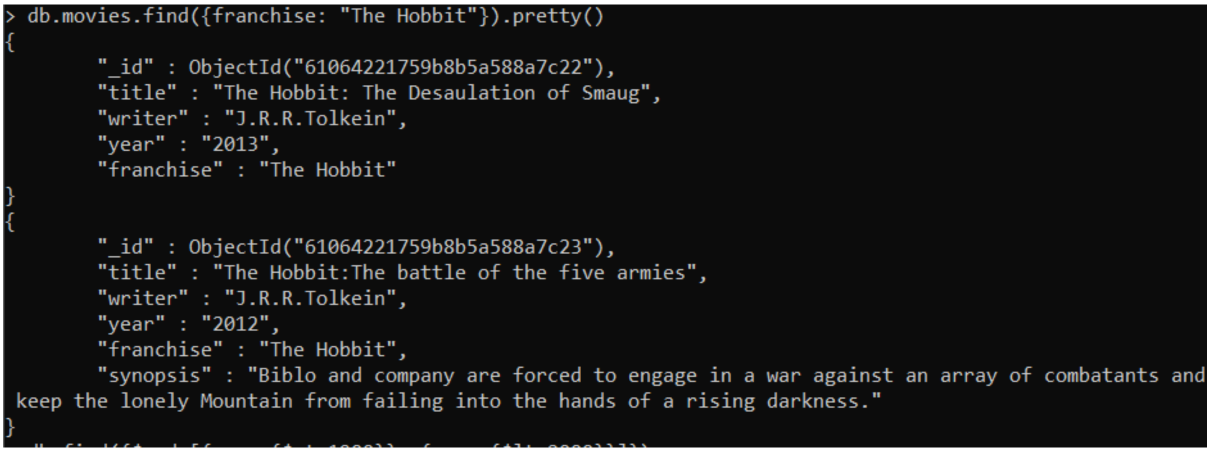
```
> db.movies.find({writer: 'Quentin Tarantino'})
{ "_id" : ObjectId("610637436dc2f82b3f127272"), "title" : "Pulp Fiction", "writer" : "Quentin Tarantino", "year" : "1994", "actors" : [ "John Travolta", "Uma Thurman" ] }
{ "_id" : ObjectId("61064221759b8b5a588a7c20"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year" : "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }
>
```

5.



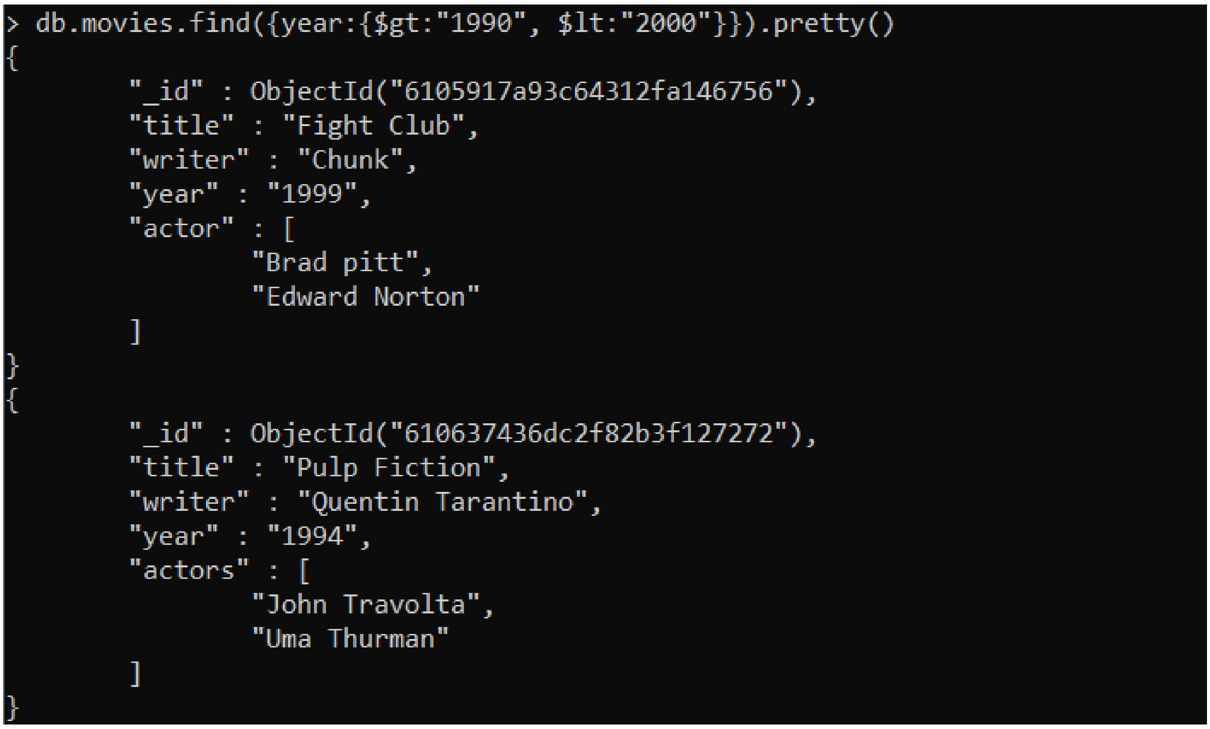
```
> db.movies.find({actors:"Brad Pitt"})
{ "_id" : ObjectId("61064221759b8b5a588a7c20"), "title" : "Inglorious Basterds", "writer" : "Quentin Tarantino", "year"
: "2009", "actors" : [ "Brad Pitt", "Diane Kruger", "Eli Roth" ] }
>
```

6.



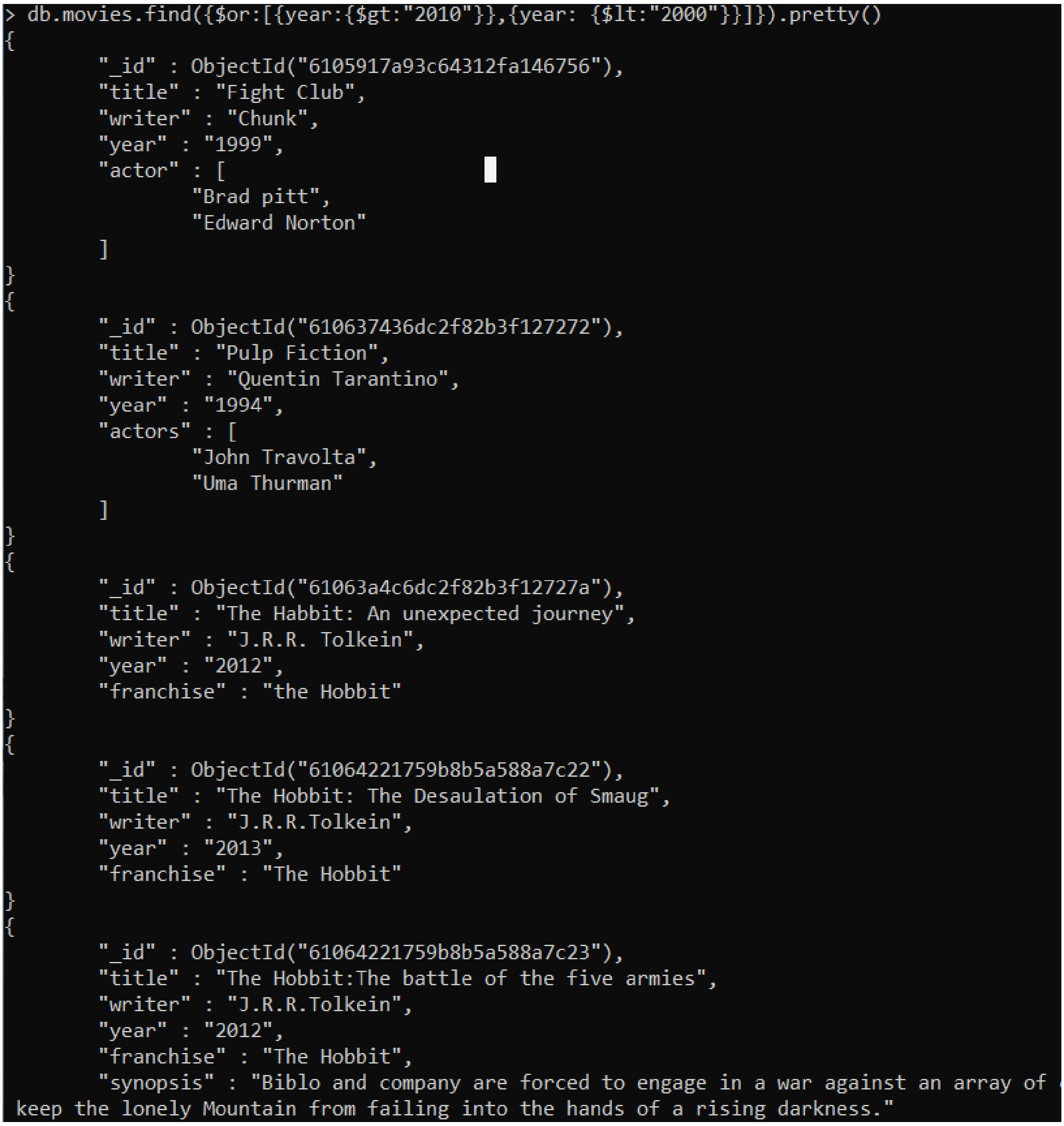
```
> db.movies.find({franchise: "The Hobbit"}).pretty()
{
        "_id" : ObjectId("61064221759b8b5a588a7c22"),
        "title" : "The Hobbit: The Desaulation of Smaug",
        "writer" : "J.R.R.Tolkein",
        "year" : "2013",
        "franchise" : "The Hobbit"
}
{
        "_id" : ObjectId("61064221759b8b5a588a7c23"),
        "title" : "The Hobbit:The battle of the five armies",
        "writer" : "J.R.R.Tolkein",
        "year" : "2012",
        "franchise" : "The Hobbit",
        "synopsis" : "Biblo and company are forced to engage in a war against an array of combatants and
 keep the lonely Mountain from failing into the hands of a rising darkness."
}
```

7.



```
> db.movies.find({year:{$gt:"1990", $lt:"2000"}}).pretty()
{
        "_id" : ObjectId("6105917a93c64312fa146756"),
        "title" : "Fight Club",
        "writer" : "Chunk",
        "year" : "1999",
        "actor" : [
                "Brad pitt",
                "Edward Norton"
        ]
}
{
        "_id" : ObjectId("610637436dc2f82b3f127272"),
        "title" : "Pulp Fiction",
        "writer" : "Quentin Tarantino",
        "year" : "1994",
        "actors" : [
                "John Travolta",
                "Uma Thurman"
        ]
}
```

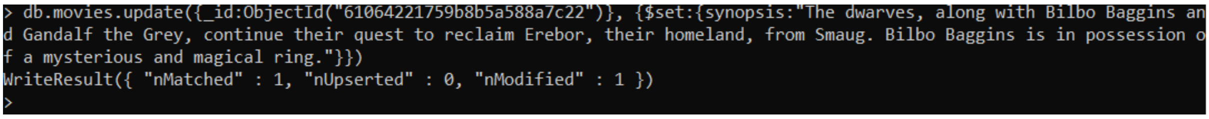
8.Movies before 2000 or after 2010



```
> db.movies.find({$or:[{year:{$gt:"2010"}},{year: {$lt:"2000"}}]}).pretty()
{
        "_id" : ObjectId("6105917a93c64312fa146756"),
        "title" : "Fight Club",
        "writer" : "Chunk",
        "year" : "1999",
        "actor" : [
                "Brad pitt",
                "Edward Norton"
        ]
}
{
        "_id" : ObjectId("610637436dc2f82b3f127272"),
        "title" : "Pulp Fiction",
        "writer" : "Quentin Tarantino",
        "year" : "1994",
        "actors" : [
                "John Travolta",
                "Uma Thurman"
        ]
}
{
        "_id" : ObjectId("61063a4c6dc2f82b3f12727a"),
        "title" : "The Habbit: An unexpected journey",
        "writer" : "J.R.R. Tolkein",
        "year" : "2012",
        "franchise" : "the Hobbit"
}
{
        "_id" : ObjectId("61064221759b8b5a588a7c22"),
        "title" : "The Hobbit: The Desaulation of Smaug",
        "writer" : "J.R.R.Tolkein",
        "year" : "2013",
        "franchise" : "The Hobbit"
}
{
        "_id" : ObjectId("61064221759b8b5a588a7c23"),
        "title" : "The Hobbit:The battle of the five armies",
        "writer" : "J.R.R.Tolkein",
        "year" : "2012",
        "franchise" : "The Hobbit",
        "synopsis" : "Biblo and company are forced to engage in a war against an array of
 keep the lonely Mountain from failing into the hands of a rising darkness."
```

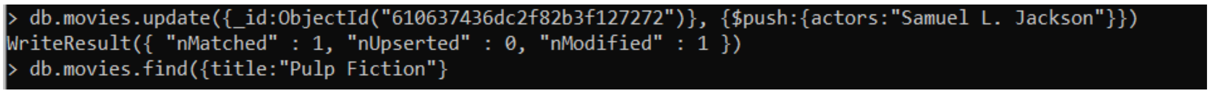
Update Documents

1



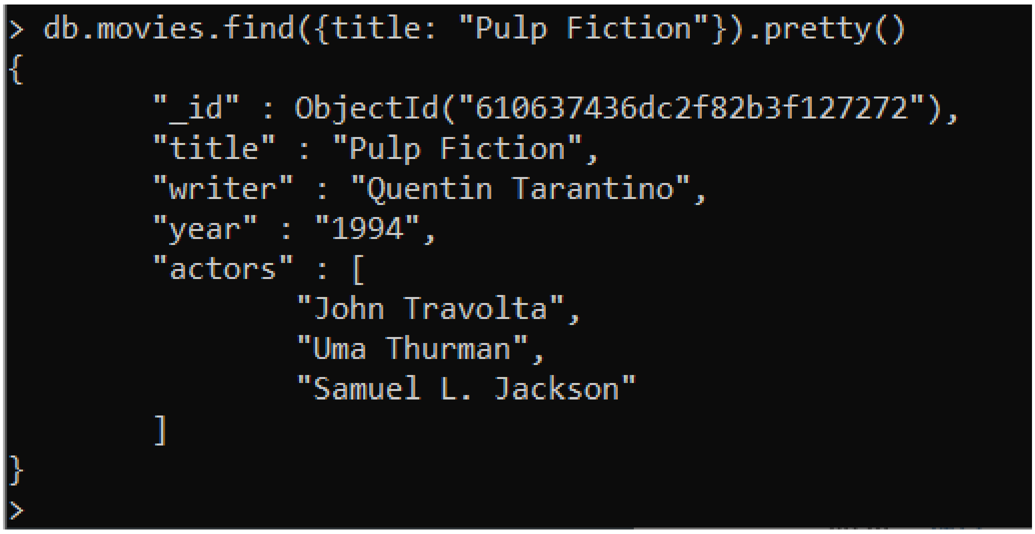
```
> db.movies.update({_id:ObjectId("61064221759b8b5a588a7c22")}, {$set:{synopsis:"The dwarves, along with Bilbo Baggins an
d Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession o
f a mysterious and magical ring."}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>
```

2.



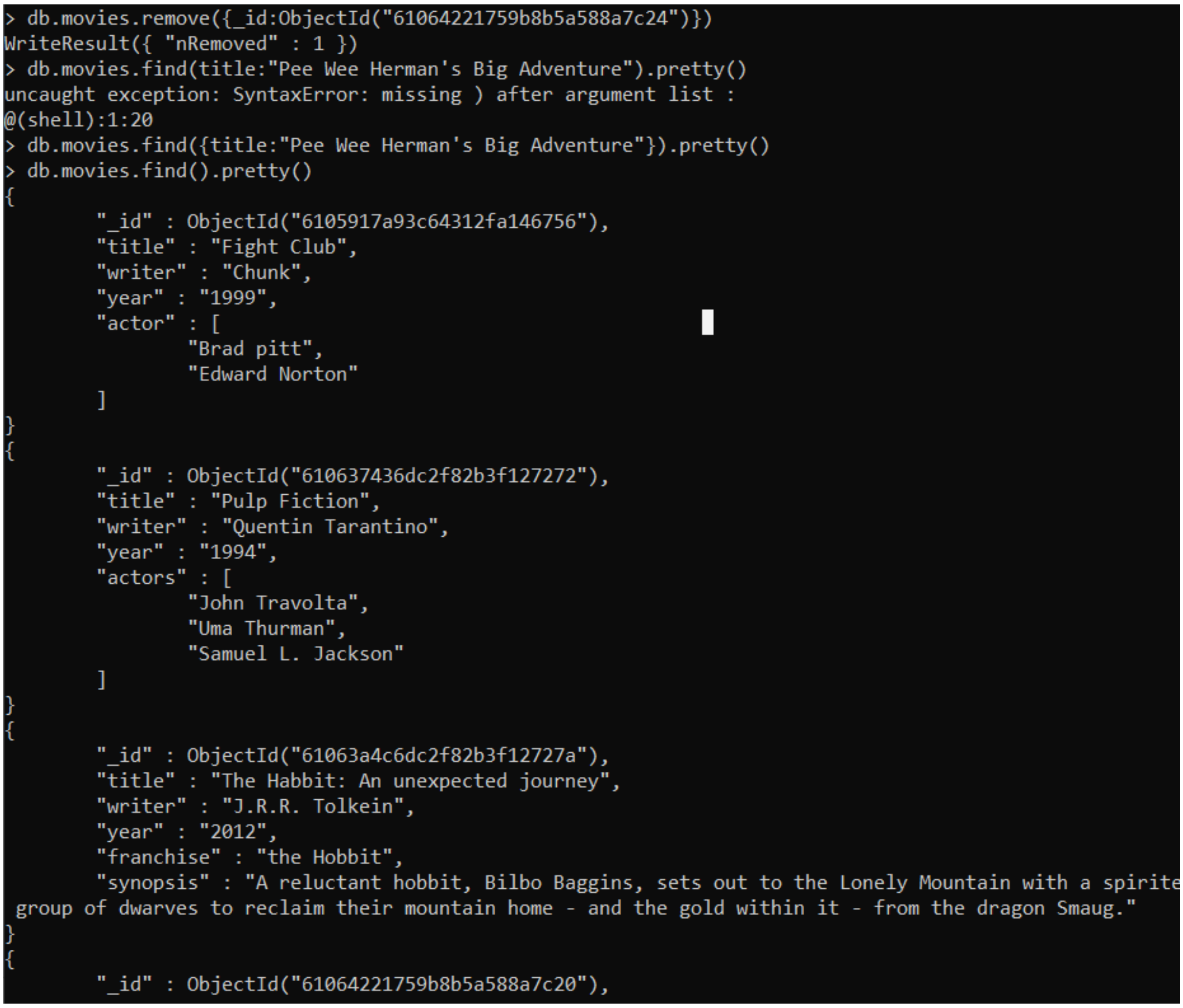
```
> db.movies.update({_id:ObjectId("610637436dc2f82b3f127272")}, {$push:{actors:"Samuel L. Jackson"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.movies.find({title:"Pulp Fiction"}
```

3.

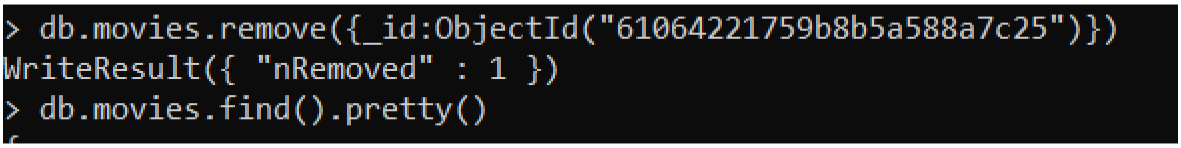


```
> db.movies.find({title: "Pulp Fiction"}).pretty()
{
        "_id" : ObjectId("610637436dc2f82b3f127272"),
        "title" : "Pulp Fiction",
        "writer" : "Quentin Tarantino",
        "year" : "1994",
        "actors" : [
                "John Travolta",
                "Uma Thurman",
                "Samuel L. Jackson"
        ]
}
>
```

Deleting documents



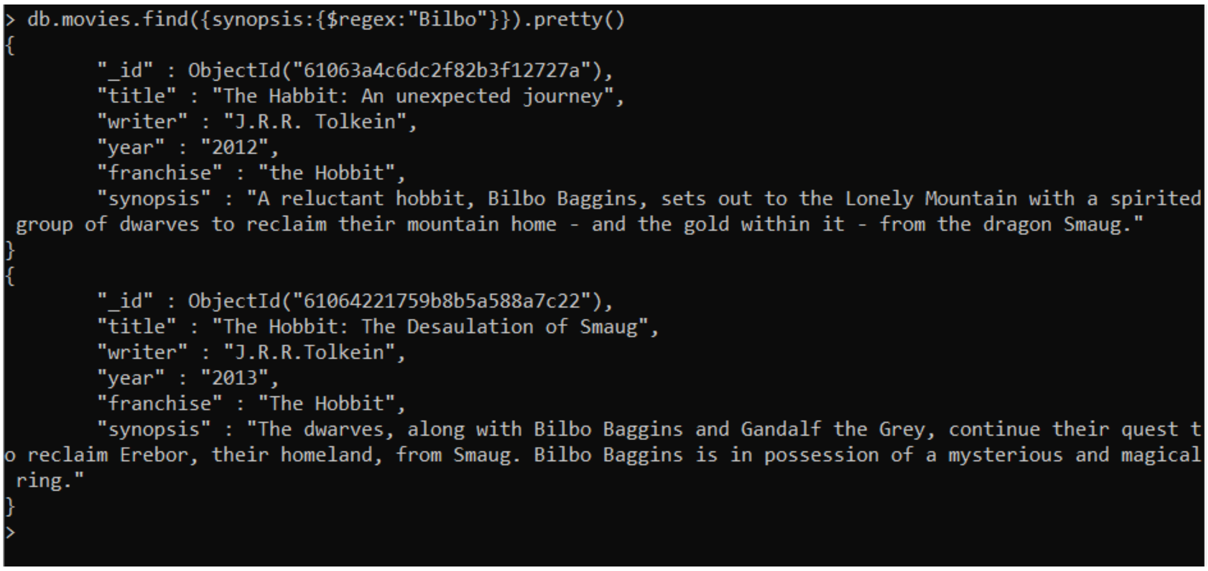
```
> db.movies.remove({_id:ObjectId("61064221759b8b5a588a7c24")})
WriteResult({ "nRemoved" : 1 })
> db.movies.find(title:"Pee Wee Herman's Big Adventure").pretty()
uncaught exception: SyntaxError: missing ) after argument list :
@(shell):1:20
> db.movies.find({title:"Pee Wee Herman's Big Adventure"}).pretty()
> db.movies.find().pretty()
{
        "_id" : ObjectId("6105917a93c64312fa146756"),
        "title" : "Fight Club",
        "writer" : "Chunk",
        "year" : "1999",
        "actor" : [
                "Brad pitt",
                "Edward Norton"
        ]
}
{
        "_id" : ObjectId("610637436dc2f82b3f127272"),
        "title" : "Pulp Fiction",
        "writer" : "Quentin Tarantino",
        "year" : "1994",
        "actors" : [
                "John Travolta",
                "Uma Thurman",
                "Samuel L. Jackson"
        ]
}
{
        "_id" : ObjectId("61063a4c6dc2f82b3f12727a"),
        "title" : "The Habbit: 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 spirite
 group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."
}
{
        "_id" : ObjectId("61064221759b8b5a588a7c20"),
```



```
> db.movies.remove({_id:ObjectId("61064221759b8b5a588a7c25")})
WriteResult({ "nRemoved" : 1 })
> db.movies.find().pretty()
```

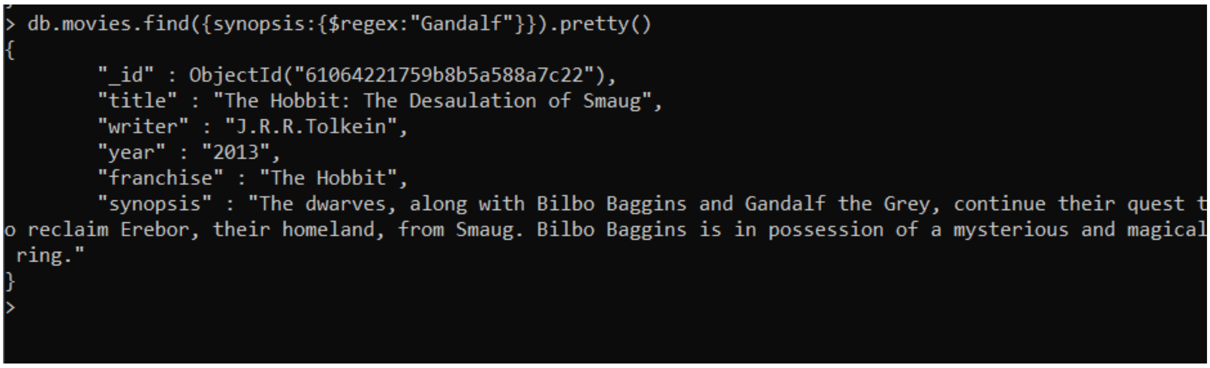
Text Search

1



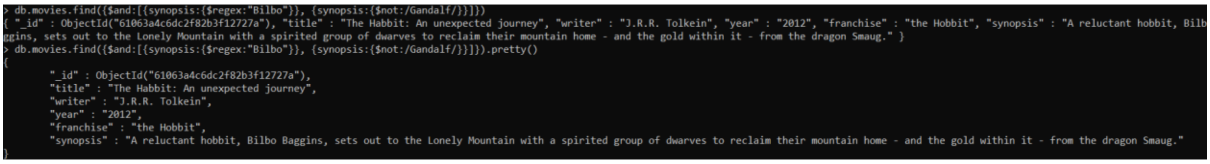
```
> db.movies.find({synopsis:{$regex:"Bilbo"}}).pretty()
{
        "_id" : ObjectId("61063a4c6dc2f82b3f12727a"),
        "title" : "The Habbit: An unexpected journey",
        "writer" : "J.R.R. Tolkein",
        "year" : "2012",
        "franchise" : "the Hobbit",
        "synopsis" : "A reluctant 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("61064221759b8b5a588a7c22"),
        "title" : "The Hobbit: The Desaulation 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 t
o reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and magical
 ring."
}
>
```

2.



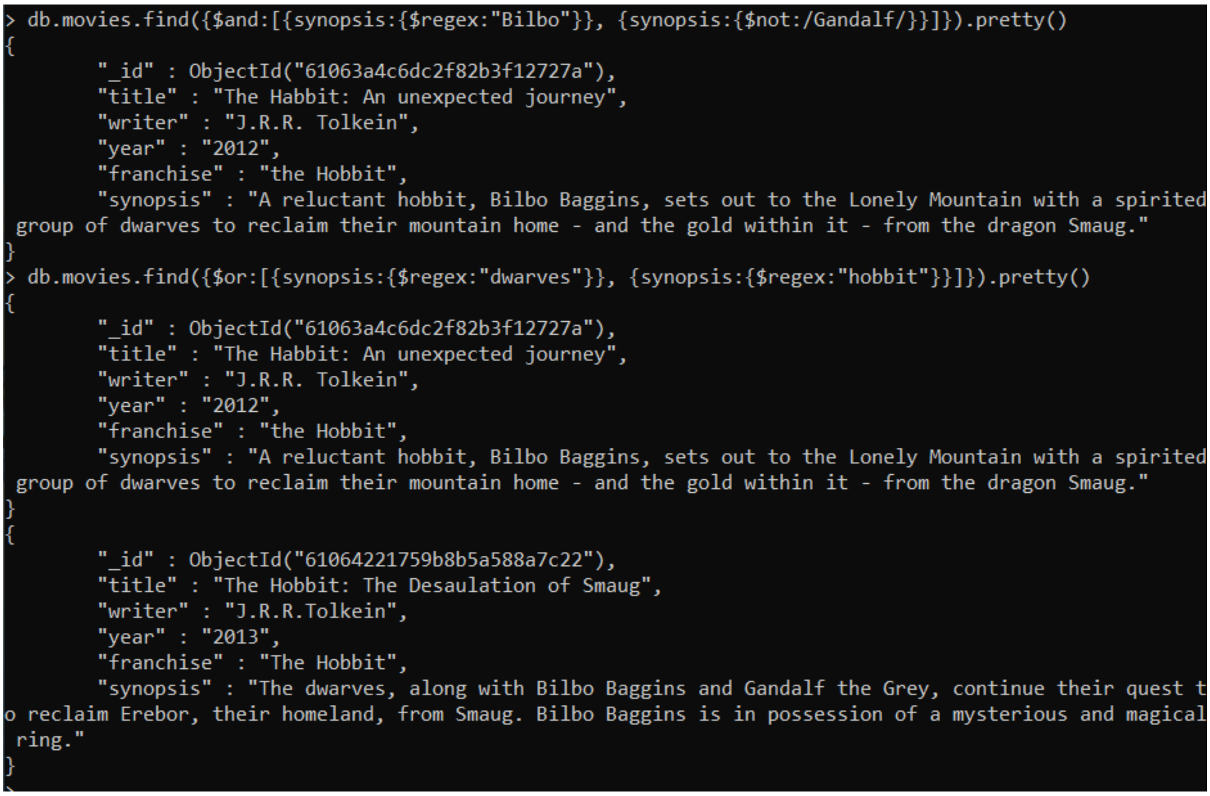
```
> db.movies.find({synopsis:{$regex:"Gandalf"}}).pretty()
{
        "_id" : ObjectId("61064221759b8b5a588a7c22"),
        "title" : "The Hobbit: The Desaulation of Smaug",
        "writer" : "J.R.R.Tolkein",
        "year" : "2013",
        "franchise" : "The Hobbit",
        "synopsis" : "The dwarves, along with Bilbo Baggins and Gandalf the Grey, continue their quest t
o reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and magical
 ring."
}
>
```

3.



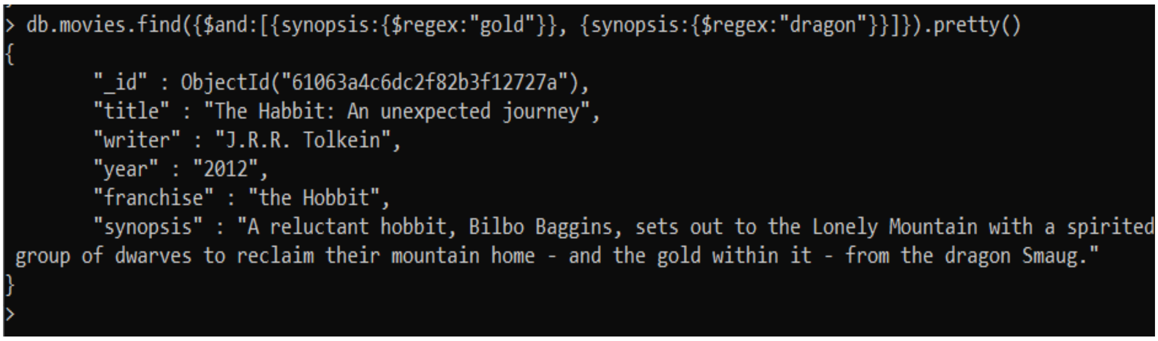
```
> db.movies.find({$and:[{synopsis:{$regex:"Bilbo"}}, {synopsis:{$not:/Gandalf/}}]})
{ "_id" : ObjectId("61063a4c6dc2f82b3f12727a"), "title" : "The Habbit: An unexpected journey", "writer" : "J.R.R. Tolkein", "year" : "2012", "franchise" : "the Hobbit", "synopsis" : "A reluctant hobbit, Bilb
ggins, sets out to the Lonely Mountain with a spirited group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug." }
> db.movies.find({$and:[{synopsis:{$regex:"Bilbo"}}, {synopsis:{$not:/Gandalf/}}]}).pretty()
{
        "_id" : ObjectId("61063a4c6dc2f82b3f12727a"),
        "title" : "The Habbit: 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."
}
```

4.



```
> db.movies.find({$and:[{synopsis:{$regex:"Bilbo"}}, {synopsis:{$not:/Gandalf/}}]}).pretty()
{
        "_id" : ObjectId("61063a4c6dc2f82b3f12727a"),
        "title" : "The Habbit: An unexpected journey",
        "writer" : "J.R.R. Tolkein",
        "year" : "2012",
        "franchise" : "the Hobbit",
        "synopsis" : "A reluctant hobbit, Bilbo Baggins, sets out to the Lonely Mountain with a spirited
 group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."
}
> db.movies.find({$or:[{synopsis:{$regex:"dwarves"}}, {synopsis:{$regex:"hobbit"}}]}).pretty()
{
        "_id" : ObjectId("61063a4c6dc2f82b3f12727a"),
        "title" : "The Habbit: 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("61064221759b8b5a588a7c22"),
        "title" : "The Hobbit: The Desaulation 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 t
o reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and magical
 ring."
}
```

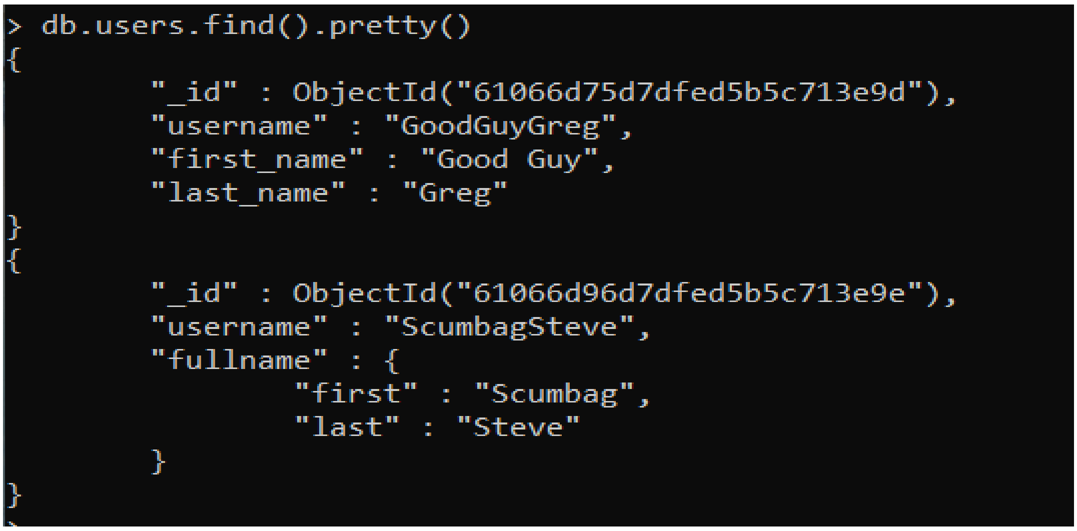
5.



```
> db.movies.find({$and:[{synopsis:{$regex:"gold"}}, {synopsis:{$regex:"dragon"}}]}).pretty()
{
        "_id" : ObjectId("61063a4c6dc2f82b3f12727a"),
        "title" : "The Habbit: 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."
}
>
```

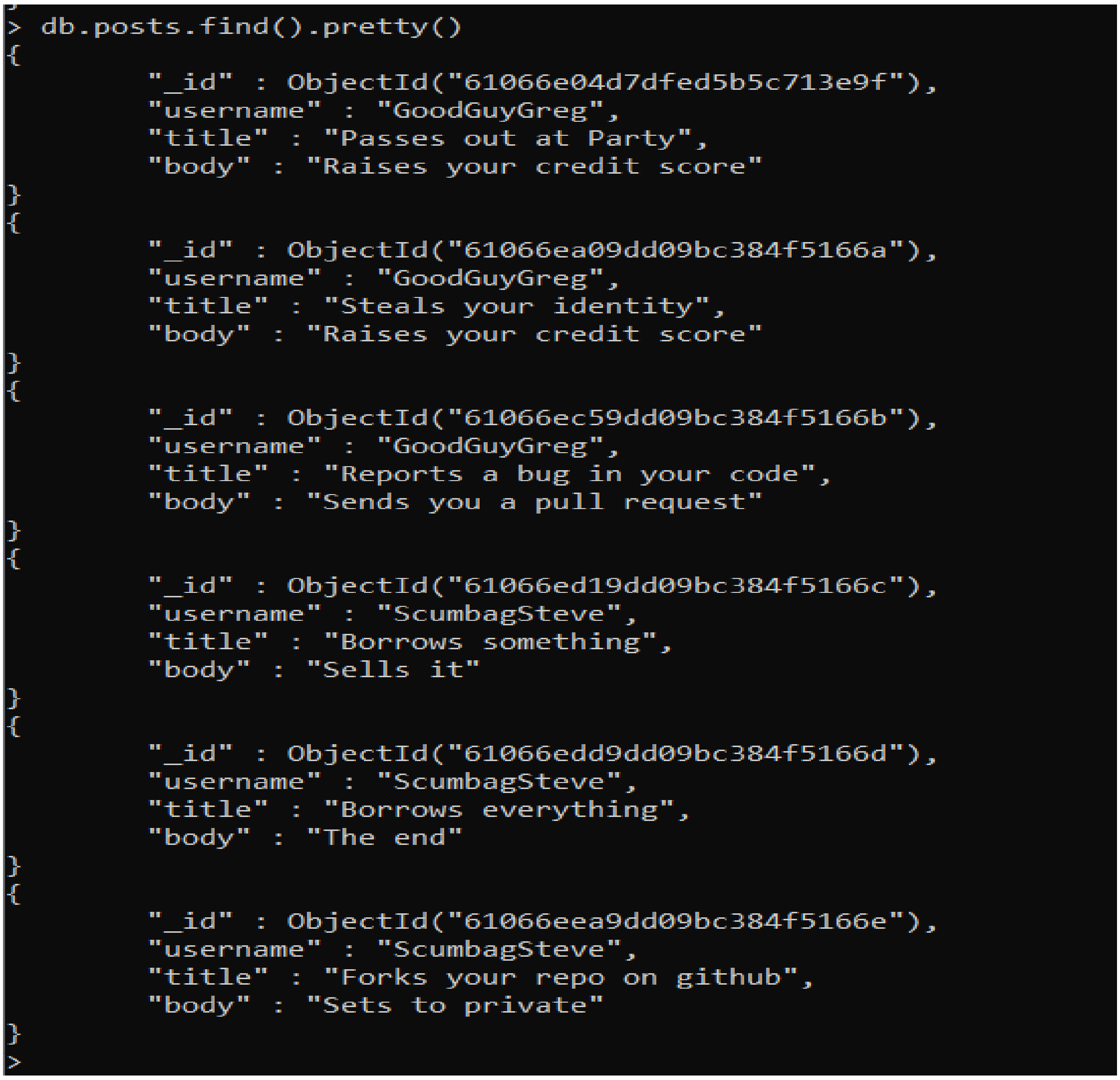
Relationships

1.



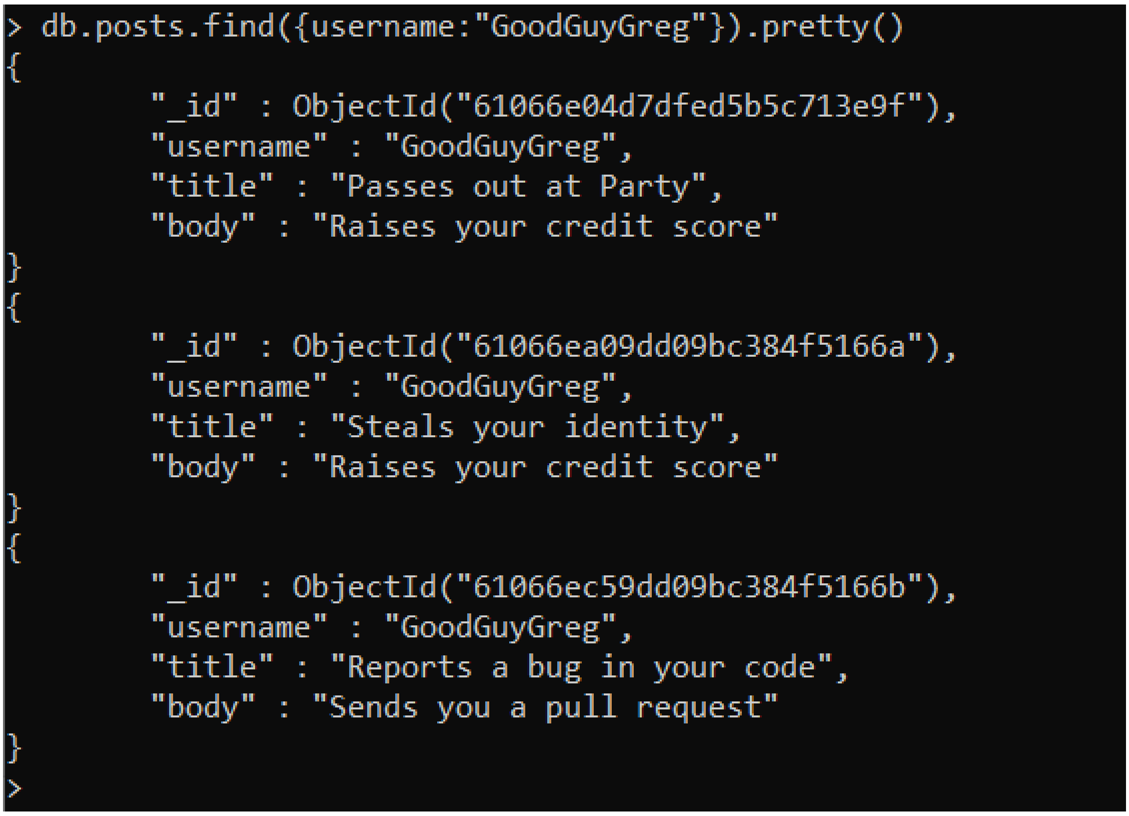
```
> db.users.find().pretty()
{
        "_id" : ObjectId("61066d75d7dfed5b5c713e9d"),
        "username" : "GoodGuyGreg",
        "first_name" : "Good Guy",
        "last_name" : "Greg"
}
{
        "_id" : ObjectId("61066d96d7dfed5b5c713e9e"),
        "username" : "ScumbagSteve",
        "fullname" : {
                "first" : "Scumbag",
                "last" : "Steve"
        }
}
```

2.



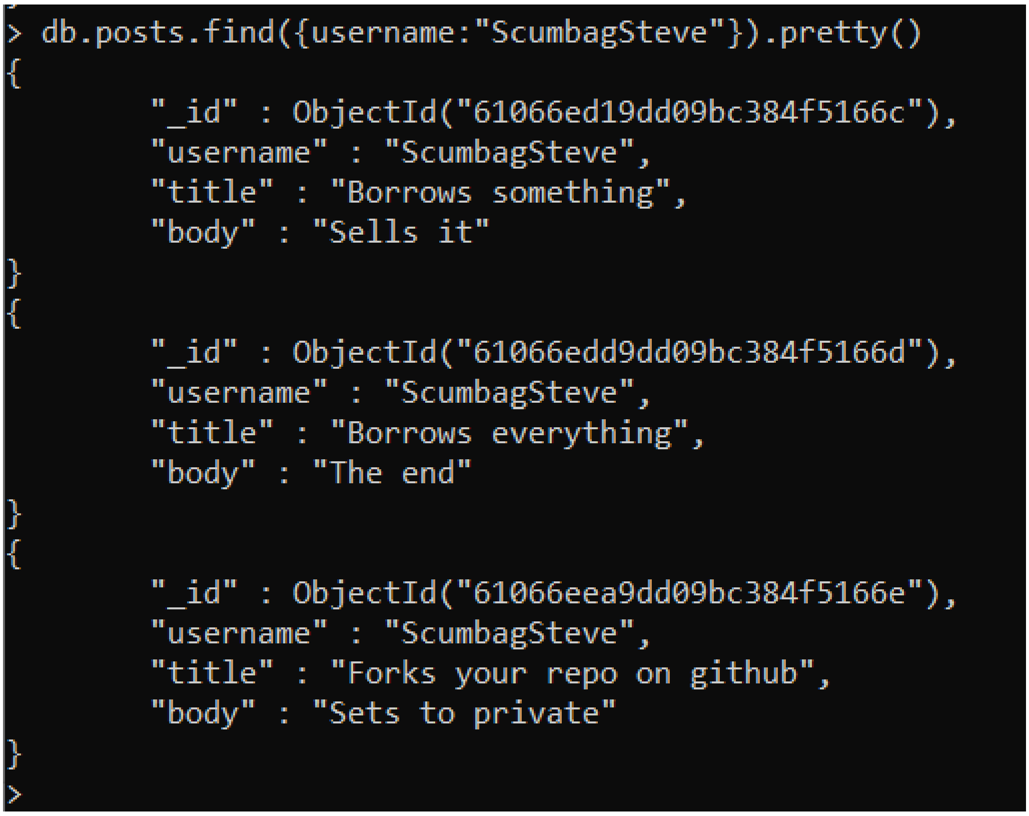
```
> db.posts.find().pretty()
{
        "_id" : ObjectId("61066e04d7dfed5b5c713e9f"),
        "username" : "GoodGuyGreg",
        "title" : "Passes out at Party",
        "body" : "Raises your credit score"
}
{
        "_id" : ObjectId("61066ea09dd09bc384f5166a"),
        "username" : "GoodGuyGreg",
        "title" : "Steals your identity",
        "body" : "Raises your credit score"
}
{
        "_id" : ObjectId("61066ec59dd09bc384f5166b"),
        "username" : "GoodGuyGreg",
        "title" : "Reports a bug in your code",
        "body" : "Sends you a pull request"
}
{
        "_id" : ObjectId("61066ed19dd09bc384f5166c"),
        "username" : "ScumbagSteve",
        "title" : "Borrows something",
        "body" : "Sells it"
}
{
        "_id" : ObjectId("61066edd9dd09bc384f5166d"),
        "username" : "ScumbagSteve",
        "title" : "Borrows everything",
        "body" : "The end"
}
{
        "_id" : ObjectId("61066eea9dd09bc384f5166e"),
        "username" : "ScumbagSteve",
        "title" : "Forks your repo on github",
        "body" : "Sets to private"
}
>
```

3.



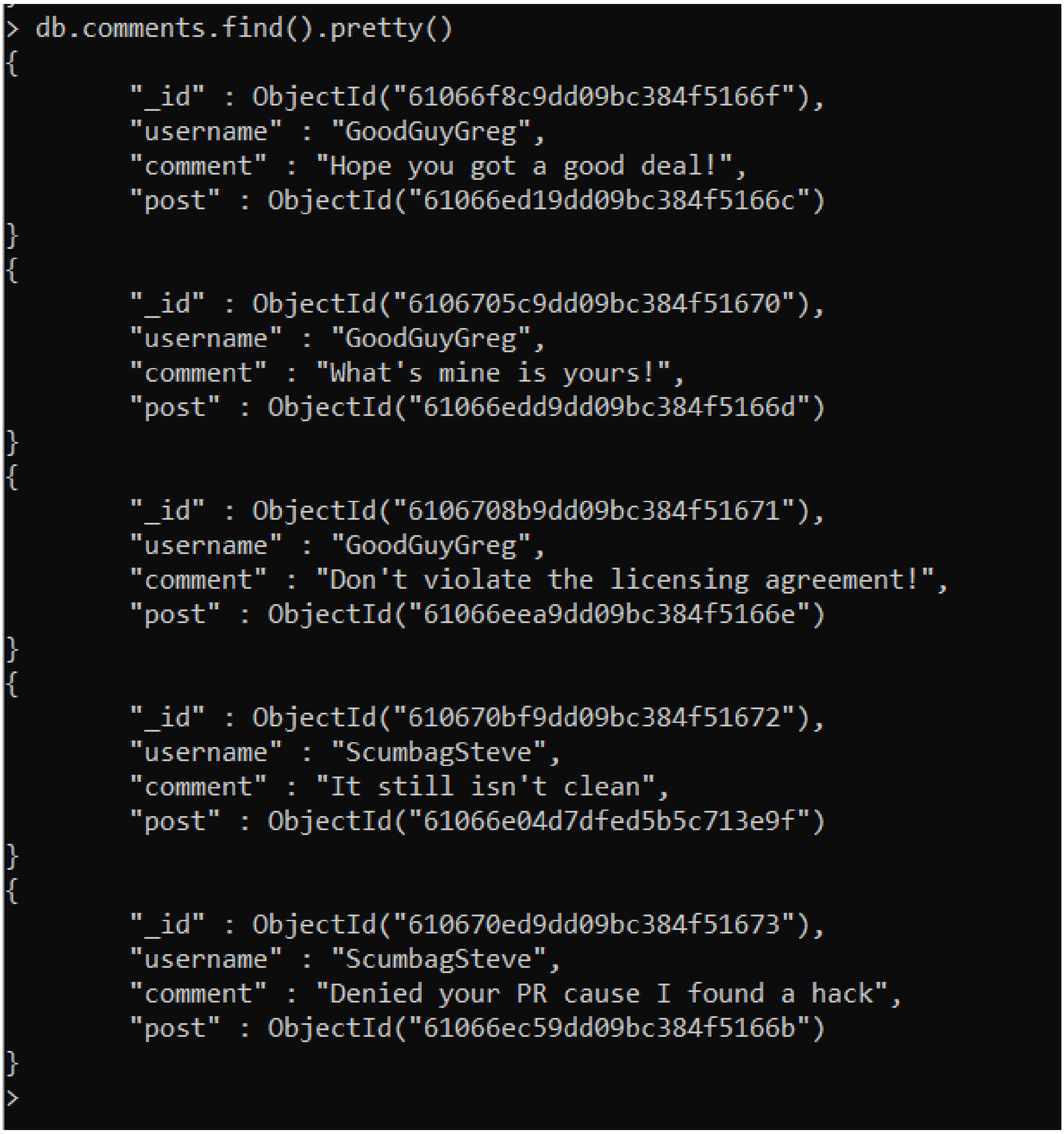
```
> db.posts.find({username:"GoodGuyGreg"}).pretty()
{
        "_id" : ObjectId("61066e04d7dfed5b5c713e9f"),
        "username" : "GoodGuyGreg",
        "title" : "Passes out at Party",
        "body" : "Raises your credit score"
}
{
        "_id" : ObjectId("61066ea09dd09bc384f5166a"),
        "username" : "GoodGuyGreg",
        "title" : "Steals your identity",
        "body" : "Raises your credit score"
}
{
        "_id" : ObjectId("61066ec59dd09bc384f5166b"),
        "username" : "GoodGuyGreg",
        "title" : "Reports a bug in your code",
        "body" : "Sends you a pull request"
}
>
```

4.



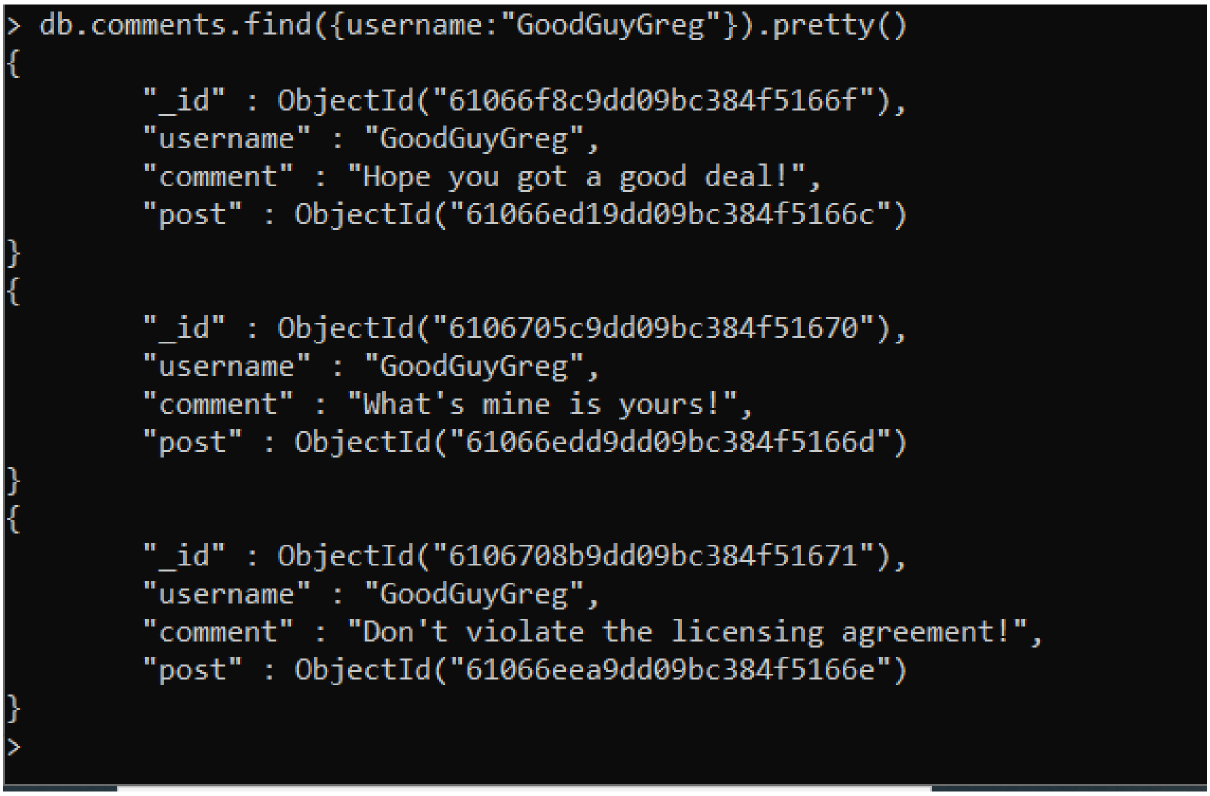
```
> db.posts.find({username:"ScumbagSteve"}).pretty()
{
        "_id" : ObjectId("61066ed19dd09bc384f5166c"),
        "username" : "ScumbagSteve",
        "title" : "Borrows something",
        "body" : "Sells it"
}
{
        "_id" : ObjectId("61066edd9dd09bc384f5166d"),
        "username" : "ScumbagSteve",
        "title" : "Borrows everything",
        "body" : "The end"
}
{
        "_id" : ObjectId("61066eea9dd09bc384f5166e"),
        "username" : "ScumbagSteve",
        "title" : "Forks your repo on github",
        "body" : "Sets to private"
}
>
```

4.



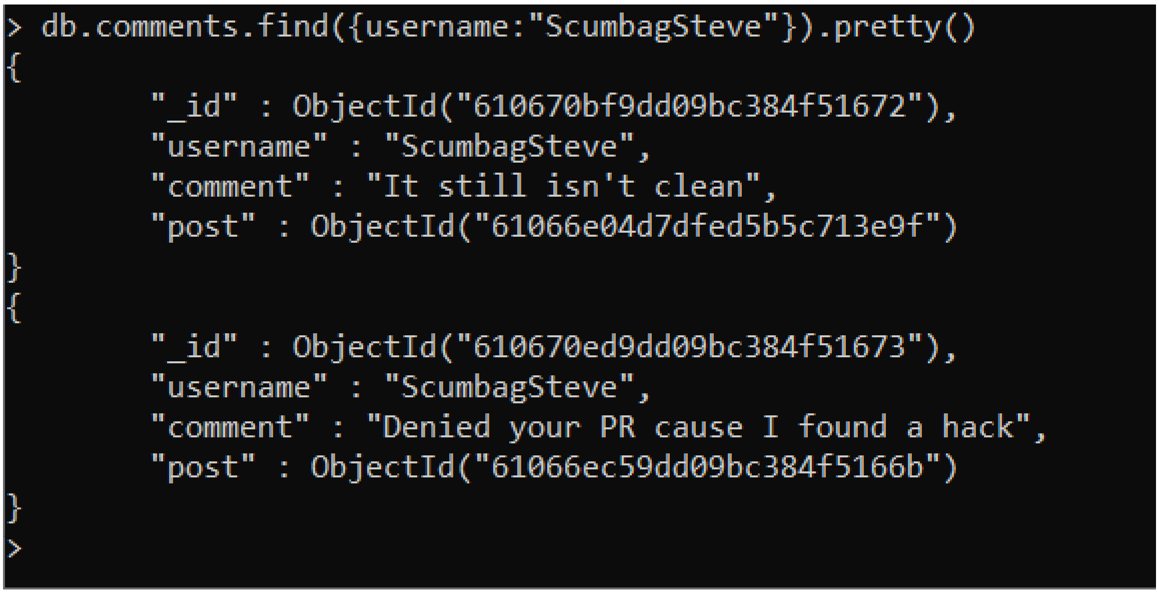
```
> db.comments.find().pretty()
{
        "_id" : ObjectId("61066f8c9dd09bc384f5166f"),
        "username" : "GoodGuyGreg",
        "comment" : "Hope you got a good deal!",
        "post" : ObjectId("61066ed19dd09bc384f5166c")
}
{
        "_id" : ObjectId("6106705c9dd09bc384f51670"),
        "username" : "GoodGuyGreg",
        "comment" : "What's mine is yours!",
        "post" : ObjectId("61066edd9dd09bc384f5166d")
}
{
        "_id" : ObjectId("6106708b9dd09bc384f51671"),
        "username" : "GoodGuyGreg",
        "comment" : "Don't violate the licensing agreement!",
        "post" : ObjectId("61066eea9dd09bc384f5166e")
}
{
        "_id" : ObjectId("610670bf9dd09bc384f51672"),
        "username" : "ScumbagSteve",
        "comment" : "It still isn't clean",
        "post" : ObjectId("61066e04d7dfed5b5c713e9f")
}
{
        "_id" : ObjectId("610670ed9dd09bc384f51673"),
        "username" : "ScumbagSteve",
        "comment" : "Denied your PR cause I found a hack",
        "post" : ObjectId("61066ec59dd09bc384f5166b")
}
>
```

5.



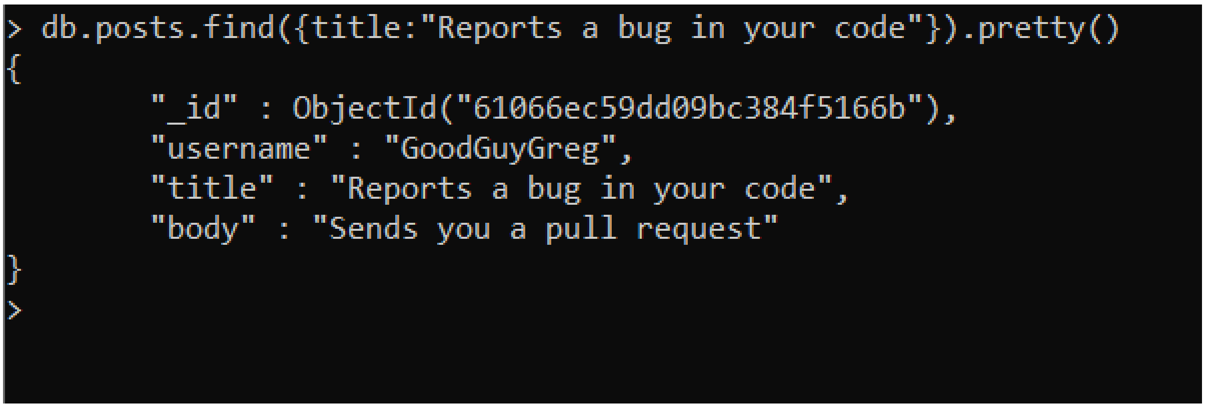
```
> db.comments.find({username:"GoodGuyGreg"}).pretty()
{
        "_id" : ObjectId("61066f8c9dd09bc384f5166f"),
        "username" : "GoodGuyGreg",
        "comment" : "Hope you got a good deal!",
        "post" : ObjectId("61066ed19dd09bc384f5166c")
}
{
        "_id" : ObjectId("6106705c9dd09bc384f51670"),
        "username" : "GoodGuyGreg",
        "comment" : "What's mine is yours!",
        "post" : ObjectId("61066edd9dd09bc384f5166d")
}
{
        "_id" : ObjectId("6106708b9dd09bc384f51671"),
        "username" : "GoodGuyGreg",
        "comment" : "Don't violate the licensing agreement!",
        "post" : ObjectId("61066eea9dd09bc384f5166e")
}
>
```

6.



```
> db.comments.find({username:"ScumbagSteve"}).pretty()
{
        "_id" : ObjectId("610670bf9dd09bc384f51672"),
        "username" : "ScumbagSteve",
        "comment" : "It still isn't clean",
        "post" : ObjectId("61066e04d7dfed5b5c713e9f")
}
{
        "_id" : ObjectId("610670ed9dd09bc384f51673"),
        "username" : "ScumbagSteve",
        "comment" : "Denied your PR cause I found a hack",
        "post" : ObjectId("61066ec59dd09bc384f5166b")
}
>
```

7.



```
> db.posts.find({title:"Reports a bug in your code"}).pretty()
{
        "_id" : ObjectId("61066ec59dd09bc384f5166b"),
        "username" : "GoodGuyGreg",
        "title" : "Reports a bug in your code",
        "body" : "Sends you a pull request"
}
>
```