

# MongoDB Assignment 1

## 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.

```
Atlas atlas-2w6f2t-shard-0 [primary] test> use mongo_practice
```

```
switched to db mongo_practice
```

```
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.createCollection('movies');
```

```
{ ok: 1 }
```

```
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({title:"Fight Club",
writer:"Chuck Palahniuko", year:"1999", actors:["Brad Pitt", "Edward Norton"]})
```

```
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or
bulkWrite.
```

```
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("62033c4a6f4489cae2c8559f") }
}
```

```
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({title:"Fight Club",
writer:"Chuck Palahniuko", year:"1999", actors:["Brad Pitt, Edward Norton"]})
```

```
... )
```

```
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("62033d206f4489cae2c855a0") }
}
```

```
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({ title: "Pulp Fiction",
writer: "Quentin Tarantino", year: "1994", actors: "[John Travolta, Uma Thurman]" })
```

```
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("62033daa6f4489cae2c855a1") }
}
```

```
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({ title: "Inglorious
Basterds", writer: "Quentin Tarantino", year: "2009", actors: "[Brad Pitt, Diane Kruger, Eli
Roth]" })
```

```
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("62033e6c6f4489cae2c855a2") }
}
```

```
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({ title: "The Hobbit:
An Unexpected Journey", writer: "J.R.R Tolkein", year: "2012", franchise:"The Hobbit"})
```

```
{
```

```

    acknowledged: true,
    insertedIds: { '0': ObjectId("62033f686f4489cae2c855a3") }
  }
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({ title: "The Hobbit:
The Desolation of Smaug", writer: "J.R.R Tolkein", year: "2013", franchise:"The Hobbit"})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("62033fb66f4489cae2c855a4") }
}
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({ 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."})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("620340a56f4489cae2c855a6") }
}
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({title:"Pee Wee
Herman's Big Adventure"})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6203413f6f4489cae2c855a7") }
}
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.insert({title:"Avatar"})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("620341506f4489cae2c855a8") }
}

```

### **Query / Find Documents**

```

Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.find()
[
  {
    _id: ObjectId("62033d206f4489cae2c855a0"),
    title: 'Fight Club',
    writer: 'Chuck Palahniuko',
    year: '1999',
    actors: '[Brad Pitt, Edward Norton]'
  },
  {
    _id: ObjectId("62033daa6f4489cae2c855a1"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: '1994',
    actors: '[John Travolta, Uma Thurman]'
  },
  {

```

```

    _id: ObjectId("62033e6c6f4489cae2c855a2"),
    title: 'Inglorious Basterds',
    writer: 'Quentin Tarantino',
    year: '2009',
    actors: '[Brad Pitt, Diane Kruger, Eli Roth]'
  },
  {
    _id: ObjectId("62033f686f4489cae2c855a3"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolkein',
    year: '2012',
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("62033fb66f4489cae2c855a4"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R Tolkein',
    year: '2013',
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("620340a56f4489cae2c855a6"),
    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("6203413f6f4489cae2c855a7"),
    title: "Pee Wee Herman's Big Adventure"
  },
  { _id: ObjectId("620341506f4489cae2c855a8"), title: 'Avatar' }
]
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.find({writer:"Quentin
Tarantino"})
[
  {
    _id: ObjectId("62033daa6f4489cae2c855a1"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: '1994',
    actors: '[John Travolta, Uma Thurman]'
  },
  {
    _id: ObjectId("62033e6c6f4489cae2c855a2"),
    title: 'Inglorious Basterds',

```

```

    writer: 'Quentin Tarantino',
    year: '2009',
    actors: '[Brad Pitt, Diane Kruger, Eli Roth]'
  }
]
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.find({franchise:"The
Hobbit"})
[
  {
    _id: ObjectId("62033f686f4489cae2c855a3"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolkein',
    year: '2012',
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("62033fb66f4489cae2c855a4"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R Tolkein',
    year: '2013',
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("620340a56f4489cae2c855a6"),
    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-2w6f2t-shard-0 [primary] mongo_practice> db.movies.find({year: {$gt:"1990",
$lt:"2000"}})
[
  {
    _id: ObjectId("62033d206f4489cae2c855a0"),
    title: 'Fight Club',
    writer: 'Chuck Palahniuko',
    year: '1999',
    actors: '[Brad Pitt, Edward Norton]'
  },
  {
    _id: ObjectId("62033daa6f4489cae2c855a1"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: '1994',
    actors: '[John Travolta, Uma Thurman]'
  }
]

```

```

}
]
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.find({$or:[{year:
{$gt:"2010"}}, {year: {$lt:"2000"}}]})
[
{
  _id: ObjectId("62033d206f4489cae2c855a0"),
  title: 'Fight Club',
  writer: 'Chuck Palahniuko',
  year: '1999',
  actors: '[Brad Pitt, Edward Norton]'
},
{
  _id: ObjectId("62033daa6f4489cae2c855a1"),
  title: 'Pulp Fiction',
  writer: 'Quentin Tarantino',
  year: '1994',
  actors: '[John Travolta, Uma Thurman]'
},
{
  _id: ObjectId("62033f686f4489cae2c855a3"),
  title: 'The Hobbit: An Unexpected Journey',
  writer: 'J.R.R Tolkein',
  year: '2012',
  franchise: 'The Hobbit'
},
{
  _id: ObjectId("62033fb66f4489cae2c855a4"),
  title: 'The Hobbit: The Desolation of Smaug',
  writer: 'J.R.R Tolkein',
  year: '2013',
  franchise: 'The Hobbit'
},
{
  _id: ObjectId("620340a56f4489cae2c855a6"),
  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**

```

Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.movies.update({_id:
ObjectId("62033f686f4489cae2c855a3")}, {$set:{synopsis:"A reluctant Hobbit, Biblo Baggins,

```

sets out to the Lonely Mountains with a spirited group of dwarves to reclaim their mountain home - and the gold with 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-2w6f2t-shard-0 [primary] mongo\_practice> db.movies.update({\_id: ObjectId("62033fb66f4489cae2c855a4")}, {\$set:{synopsis:"The dwarves, along with Biblo Baggins, and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Biblo Baggins is in possession of a mysterious and magical ring."}})

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

### **Text Search**

Atlas atlas-2w6f2t-shard-0 [primary] mongo\_practice> db.movies.find({synopsis:{\$regex:"Biblo"}})

```
[
  {
    _id: ObjectId("62033f686f4489cae2c855a3"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolkein',
    year: '2012',
    franchise: 'The Hobbit',
    synopsis: 'A reluctant Hobbit, Biblo Baggins, sets out to the Lonely Mountains with a spirited group of dwarves to reclaim their mountain home - and the gold with it - from the dragon Smaug.'
  },
  {
    _id: ObjectId("62033fb66f4489cae2c855a4"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R Tolkein',
    year: '2013',
    franchise: 'The Hobbit',
    synopsis: 'The dwarves, along with Biblo Baggins, and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Biblo Baggins is in possession of a mysterious and magical ring.'
  }
]
```

```
]
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice>
db.movies.find({synopsis:{$regex:"Gandalf"}})
[
  {
    _id: ObjectId("62033fb66f4489cae2c855a4"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R Tolkein',
    year: '2013',
    franchise: 'The Hobbit',
    synopsis: 'The dwarves, along with Biblo Baggins, and Gandalf the Grey, continue their
quest to reclaim Erebor, their homeland, from Smaug. Biblo Baggins is in possession of a
mysterious and magical ring.'

```

```
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice>
db.movies.find({$and:[{synopsis:{$regex:"Biblo"}},{synopsis:{$not:/Gandalf/}}]})
[
  {
    _id: ObjectId("62033f686f4489cae2c855a3"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolkein',
    year: '2012',
    franchise: 'The Hobbit',
    synopsis: 'A reluctant Hobbit, Biblo Baggins, sets out to the Lonely Mountains with a
spirited group of dwarves to reclaim their mountain home - and the gold with it - from the
dragon Smaug.'

```

```
Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice>
db.movies.find({$or:[{synopsis:{$regex:"dwarves"}}, {synopsis:{$regex:"hobbit"}}]})
[
  {
    _id: ObjectId("62033f686f4489cae2c855a3"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R Tolkein',
    year: '2012',
    franchise: 'The Hobbit',
    synopsis: 'A reluctant Hobbit, Biblo Baggins, sets out to the Lonely Mountains with a
spirited group of dwarves to reclaim their mountain home - and the gold with it - from the
dragon Smaug.'
  },
  {
    _id: ObjectId("62033fb66f4489cae2c855a4"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R Tolkein',
    year: '2013',
    franchise: 'The Hobbit',

```

synopsis: 'The dwarves, along with Biblo Baggins, and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Biblo Baggins is in possession of a mysterious and magical ring.'

}

]

Atlas atlas-2w6f2t-shard-0 [primary] mongo\_practice>

db.movies.find({\$and:[{synopsis:{\$regex:"gold"}},{synopsis:{\$regex:"dragon"}}])

[

{

\_id: ObjectId("62033f686f4489cae2c855a3"),

title: 'The Hobbit: An Unexpected Journey',

writer: 'J.R.R Tolkein',

year: '2012',

franchise: 'The Hobbit',

synopsis: 'A reluctant Hobbit, Biblo Baggins, sets out to the Lonely Mountains with a spirited group of dwarves to reclaim their mountain home - and the gold with it - from the dragon Smaug.'

}

]

### **Delete Documents**

Atlas atlas-2w6f2t-shard-0 [primary] mongo\_practice> db.movies.remove({\_id:

ObjectId("6203413f6f4489cae2c855a7"))

DeprecationWarning: Collection.remove() is deprecated. Use deleteOne, deleteMany, findOneAndDelete, or bulkWrite.

{ acknowledged: true, deletedCount: 1 }

Atlas atlas-2w6f2t-shard-0 [primary] mongo\_practice> db.movies.remove({\_id:

ObjectId("620341506f4489cae2c855a8"))

{ acknowledged: true, deletedCount: 1 }

### **Relationships**

**1.** Insert the following documents into a **users** collection

Atlas atlas-2w6f2t-shard-0 [primary] mongo\_practice> db.users.insertMany([

... username:"GoodGuyGreg",

... first\_name:"Good Guy",

... last\_name:"Greg"

... }

... ,{

..... username:"ScumbagSteve",

..... first\_name:"Scumbag",

..... last\_name:"Steve",

..... }

... ]

... )

{



```

    acknowledged: true,
    insertedIds: {
      '0': ObjectId("620392042a7bcebdfcf30cd5"),
      '1': ObjectId("620392042a7bcebdfcf30cd6")
    }
  }
}

```

## 2. Insert the following documents into a **posts** collection

```

Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.posts.insertMany(
...  [
...    {
.....  username : "GoodGuyGreg",
.....  title : "Passes out at party",
.....  body : "Wakes up early and cleans house"
..... },
...  {
.....  username : "GoodGuyGreg",
.....  title : "Steals your identity",
.....  body : "Raises your credit score"
..... },
...  {
.....  username : "GoodGuyGreg",
.....  title : "Reports a bug in your code",
.....  body : "Sends you a Pull Request"
..... },
...  {
.....  username : "ScumbagSteve",
.....  title : "Borrows something",
.....  body : "Sells it"
..... },
...  {
.....  username : "ScumbagSteve",
.....  title : "Borrows everything",
.....  body : "The end"
..... },
...  {
.....  username : "ScumbagSteve",
.....  title : "Forks your repo on github",
.....  body : "Sets to private"
..... }])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("6203950a2a7bcebdfcf30cd8"),
    '1': ObjectId("6203950a2a7bcebdfcf30cd9"),
    '2': ObjectId("6203950a2a7bcebdfcf30cda"),
    '3': ObjectId("6203950a2a7bcebdfcf30cdb"),

```

```

'4': ObjectId("6203950a2a7bcebdfcf30cdc"),
'5': ObjectId("6203950a2a7bcebdfcf30cdd")
}
}

```

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

```

Atlas atlas-2w6f2t-shard-0 [primary] mongo_practice> db.comments.insertMany([
...   username : "GoodGuyGreg",
...   Comment : "Hope you got a good deal!",
...   post : ObjectId("6202ae744056de6680dc9427")
... },
... {
.....   username : "GoodGuyGreg",
.....   Comment : "What's mine is yours !",
.....   post : ObjectId("6202ae744056de6680dc9428")
..... },
... {
.....   username : "GoodGuyGreg",
.....   Comment : "Don't vioalte the licesing agreement!",
.....   post : ObjectId("6202ae744056de6680dc942a")
..... },
... {
.....   username : "ScumbagSteve",
.....   comment : "It still isn't clean",
.....   post : ObjectId("6202ae744056de6680dc9425")
..... }
... ])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("620395d12a7bcebdfcf30cde"),
    '1': ObjectId("620395d12a7bcebdfcf30cdf"),
    '2': ObjectId("620395d12a7bcebdfcf30ce0"),
    '3': ObjectId("620395d12a7bcebdfcf30ce1")
  }
}

```

### Querying related collections

```

db.users.find()
db.posts.find()
db.posts.find({username:"GoodGuyGreg"})
db.posts.find({username:"ScumbagSteve"})
db.comments.find()
db.comments.find({username:"GoodGuyGreg"})
db.comments.find({username:"ScumbagSteve"})
db.comments.find({post:ObjectId("62039aa72a7bcebdfcf30ce6")})

```