

MONGO DB ASSINGMENTS

ASSIGNMENT-1

```
mongosh mongodb+srv://cluster0.ab4al.mongodb.net/myFirstDatabase
Microsoft Windows [Version 10.0.19042.1466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Vamsipolisetty\Downloads\mongosh-1.1.9-win32-x64\bin>mongosh "mongodb+srv://cluster0.ab4al.mongodb.net/myFirstDatabase" --username Vamsi
Enter password: *****
Current Mongosh Log ID: 620272e9f0661cdfaf9c1dda
Connecting to:      mongodb+srv://cluster0.ab4al.mongodb.net/myFirstDatabase?appName=mongosh+1.1.9
Using MongoDB:      4.4.12
Using Mongosh:      1.1.9

For mongosh info see: https://docs.mongodb.com/mongod-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-z7l76e-shard-0 [primary] myFirstDatabase> use mongo_practice
switched to db mongo_practice
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> show dbs
```

```
local 4.25 GB
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.createCollection("movies")
{ ok: 1 }
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.insertMany([
... title: "Fight Club",
... writer: "Chuck Palahniuk",
... 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"
... },
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice>
... {
... 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."
... },
... ]
}
```

```

Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find()
[
  {
    _id: ObjectId("620274473a049acdb3bf52c2"),
    title: 'Fight Club',
    writer: 'Chuck Palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c3"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c4"),
    title: 'Inglorious Basterds',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diane Kruger', 'Eli Roth' ]
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c5"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R. Tolkien',
    year: 2012,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c6"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R. Tolkien',
    year: 2013,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c7"),
    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("620274473a049acdb3bf52c8"),
    title: 'Pee Wee Herman's Big Adventure'
  }
]

```

```

null
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({"writer" : "Quentin Tarantino"})
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({"writer" : "Quentin Tarantino"})
[null]
[
  {
    _id: ObjectId("620274473a049acdb3bf52c3"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c4"),
    title: 'Inglorious Basterds',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diane Kruger', 'Eli Roth' ]
  }
]
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({"actor" : "Brad Pitt"})
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({"actors" : "Brad Pitt"})
[
  {
    _id: ObjectId("620274473a049acdb3bf52c2"),
    title: 'Fight Club',
    writer: 'Chuck Palahniuko',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c4"),
    title: 'Inglorious Basterds',
    writer: 'Quentin Tarantino',
    year: 2009,
    actors: [ 'Brad Pitt', 'Diane Kruger', 'Eli Roth' ]
  }
]
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({"title" : "The Hobbit"})
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({"franchise" : "The Hobbit"})
[
  {
    _id: ObjectId("620274473a049acdb3bf52c5"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R. Tolkien',

```

```

    {
      _id: ObjectId("620274473a049acdb3bf52c4"),
      title: 'Inglorious Basterds',
      writer: 'Quentin Tarantino',
      year: 2009,
      actors: [ 'Brad Pitt', 'Diane Kruger', 'Eli Roth' ]
    }
  ]
}
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({title : "The Hobbit"})
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({franchise : "The Hobbit"})
[
  {
    _id: ObjectId("620274473a049acdb3bf52c5"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R. Tolkien',
    year: 2012,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c6"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R. Tolkien',
    year: 2013,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c7"),
    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.'
  }
]
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.({"year" : {$in: ["1990", "1999"]}})
Uncaught:
SyntaxError: Unexpected token (1:10)
> 1 | db.movies.({"year" : {$in: ["1990", "1999"]}})
    |                                     ^
    |
    |
    2 |

```

```

Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({$or: [{year : {$lt: 2000, $gt: 2010}}]})
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.find({$or: [{year : {$lt: 2000}}, {year: {$gt: 2010}}]})
[
  {
    _id: ObjectId("620274473a049acdb3bf52c2"),
    title: 'Fight Club',
    writers: 'Chuck Palahniuk',
    year: 1999,
    actors: [ 'Brad Pitt', 'Edward Norton' ]
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c3"),
    title: 'Pulp Fiction',
    writer: 'Quentin Tarantino',
    year: 1994,
    actors: [ 'John Travolta', 'Uma Thurman' ]
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c5"),
    title: 'The Hobbit: An Unexpected Journey',
    writer: 'J.R.R. Tolkien',
    year: 2012,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c6"),
    title: 'The Hobbit: The Desolation of Smaug',
    writer: 'J.R.R. Tolkien',
    year: 2013,
    franchise: 'The Hobbit'
  },
  {
    _id: ObjectId("620274473a049acdb3bf52c7"),
    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.'
  }
]

```

```

2 |
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.update((title:"The Hobbit:An Unexpected Journey"),{$set:{synopsis:"A reluctant hobbit,Bilbo Baggins, sets out to the Lonely Mountain with a spirit
ed group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."}})
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.update((title:"The Hobbit:An Unexpected Journey"),{$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: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.update((title:"Pulp Fiction"),{$push:{actors:"Samuel L. Atlas atlas-z7l76e-shard-0 [primary] mongo_practice>
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.movies.update((title:"Pulp Fiction"),{$push:{actors:"Samuel L. Jackson"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

```

Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.users.find()
{
  _id: ObjectId("620357343efb0a7969fafa73"),
  username: 'GoodGuyGreg',
  first_name: 'Good Guy',
  last_name: 'Greg'
},
{
  _id: ObjectId("620357343efb0a7969fafa74"),
  username: 'ScumbagSteve',
  full_name: [ { first: 'Scumbag' }, { last: 'Steve' } ]
}
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.posts.find()
{
  _id: ObjectId("620357563efb0a7969fafa75"),
  username: 'GoodGuyGreg',
  title: 'Passes out at party',
  body: 'Wakes up early and cleans house'
},
{
  _id: ObjectId("620357563efb0a7969fafa76"),
  username: 'GoodGuyGreg',
  title: 'Steals your identity',
  body: 'Raises your credit score'
},
{
  _id: ObjectId("620357563efb0a7969fafa77"),
  username: 'GoodGuyGreg',
  title: 'Reports a bug in your code',
  body: 'Sends you a Pull Request'
},
{
  _id: ObjectId("620357563efb0a7969fafa78"),
  username: 'ScumbagSteve',
  title: 'Borrows something',
  body: 'Sells it'
},
{
  _id: ObjectId("620357563efb0a7969fafa79"),
  username: 'ScumbagSteve',
  title: 'Borrows everything',
  body: 'The end'
},
{

```

```

    username: 'ScumbagSteve',
    title: 'Forks your repo on github',
    body: 'Sets to private'
  }
}
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.posts.find({"username": 'GoodGuyGreg'})
[
  {
    _id: ObjectId("620357563efb0a7969fala75"),
    username: 'GoodGuyGreg',
    title: 'Passes out at party',
    body: 'Makes up early and cleans house'
  },
  {
    _id: ObjectId("620357563efb0a7969fala76"),
    username: 'GoodGuyGreg',
    title: 'Steals your identity',
    body: 'Raises your credit score'
  },
  {
    _id: ObjectId("620357563efb0a7969fala77"),
    username: 'GoodGuyGreg',
    title: 'Reports a bug in your code',
    body: 'Sends you a Pull Request'
  }
]
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.posts.find({"username": 'ScumbagSteve'})
[
  {
    _id: ObjectId("620357563efb0a7969fala78"),
    username: 'ScumbagSteve',
    title: 'Borrows something',
    body: 'Sells it'
  },
  {
    _id: ObjectId("620357563efb0a7969fala79"),
    username: 'ScumbagSteve',
    title: 'Borrows everything',
    body: 'The end'
  },
  {
    _id: ObjectId("620357563efb0a7969fala7a"),
    username: 'ScumbagSteve',
    title: 'Forks your repo on github',
    body: 'Sets to private'
  }
]

```

```

    body: 'Sets to private'
  }
}
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.comments.find()
[
  {
    _id: ObjectId("620357e23efb0a7969fala7b"),
    username: 'GoodGuyGreg',
    comment: 'Hope you got a good deal!',
    post: ObjectId("620357563efb0a7969fala78")
  },
  {
    _id: ObjectId("620357e23efb0a7969fala7c"),
    username: 'GoodGuyGreg',
    comment: 'What's mine is yours!',
    post: ObjectId("620357563efb0a7969fala79")
  },
  {
    _id: ObjectId("620357e23efb0a7969fala7d"),
    username: 'GoodGuyGreg',
    comment: "Don't violate the licensing agreement!",
    post: ObjectId("620357563efb0a7969fala7a")
  },
  {
    _id: ObjectId("620357e23efb0a7969fala7e"),
    username: 'ScumbagSteve',
    comment: "It still isn't clean",
    post: ObjectId("620357563efb0a7969fala75")
  },
  {
    _id: ObjectId("620357e23efb0a7969fala7f"),
    username: 'ScumbagSteve',
    comment: 'Denied your PR cause I found a hack',
    post: ObjectId("620357563efb0a7969fala77")
  }
]

```

```

}
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.comments.find({"username": 'GoodGuyGreg'})
[
  {
    _id: ObjectId("620357e23efb0a7969fafa7b"),
    username: 'GoodGuyGreg',
    comment: 'Hope you got a good deal!',
    post: ObjectId("620357563efb0a7969fafa78")
  },
  {
    _id: ObjectId("620357e23efb0a7969fafa7c"),
    username: 'GoodGuyGreg',
    comment: 'What's mine is yours!',
    post: ObjectId("620357563efb0a7969fafa79")
  },
  {
    _id: ObjectId("620357e23efb0a7969fafa7d"),
    username: 'GoodGuyGreg',
    comment: 'Don't violate the licensing agreement!',
    post: ObjectId("620357563efb0a7969fafa7a")
  }
]
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.comments.find({"username": 'ScumbagSteve'})
[
  {
    _id: ObjectId("620357e23efb0a7969fafa7e"),
    username: 'ScumbagSteve',
    comment: 'It still isn't clean',
    post: ObjectId("620357563efb0a7969fafa75")
  },
  {
    _id: ObjectId("620357e23efb0a7969fafa7f"),
    username: 'ScumbagSteve',
    comment: 'Denied your PR cause I found a hack',
    post: ObjectId("620357563efb0a7969fafa77")
  }
]
Atlas atlas-z7l76e-shard-0 [primary] mongo_practice> db.comments.find((post: db.posts.findOne({title: "Reports a bug in your code"}))._id)
[
  {
    _id: ObjectId("620357e23efb0a7969fafa7f"),
    username: 'ScumbagSteve',
    comment: 'Denied your PR cause I found a hack',
    post: ObjectId("620357563efb0a7969fafa77")
  }
]

```

```

Atlas atlas-z7l76e-shard-0 [primary] restaurants> db.addresses.find({name: {$regex: /^Mad.*$/}}, {_id:0, name:1, borough:1, "address.coord":1, cuisine:1})
[
  {
    address: { coord: [ -73.9868597, 40.7431194 ] },
    borough: 'Manhattan',
    cuisine: 'American',
    name: 'Madison Square'
  },
  {
    address: { coord: [ -73.98302199999999, 40.742313 ] },
    borough: 'Manhattan',
    cuisine: 'Indian',
    name: 'Madras Mahal'
  },
  {
    address: { coord: [ -74.000002, 40.72735 ] },
    borough: 'Manhattan',
    cuisine: 'American',
    name: 'Madame X'
  },
  {
    address: { coord: [ -73.98171959999999, 40.7499406 ] },
    borough: 'Manhattan',
    cuisine: 'French',
    name: 'Madison Bistro'
  },
  {
    address: { coord: [ -73.9717845, 40.6897199 ] },
    borough: 'Brooklyn',
    cuisine: 'African',
    name: 'Madiha'
  },
  {
    address: { coord: [ -73.9040753, 40.9069011 ] },
    borough: 'Bronx',
    cuisine: 'Italian',
    name: 'Madison S'
  },
  {
    address: { coord: [ -73.9886598, 40.7565811 ] },
    borough: 'Manhattan',
    cuisine: 'Hotdogs',
    name: 'Madame Tussaud's'
  },
  {
    address: { coord: [ -73.95623719999999, 40.7761697 ] },
    borough: 'Manhattan',
    cuisine: 'American',
    name: 'Mad River Bar & Grille'
  }
]

```