Basic Assignment -Mongo DB

**Query / Find Documents**

query the movies collection to

1. get all documents

🡪db.movies.find().pretty()

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

🡪db.movies.find({writer: {$eq: {“Quentin Tarantino”}}}).pretty()

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

🡪db.movies.find({actors: {$eq: {“Brad Pitt”}}}).pretty()

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

🡪db.movies.find({franchise: {$eq: {“The Hobbit”}}}).pretty()

5. get all movies released in the 90s

🡪db.movies.find({year: {$gte:1990,$lte:2000}}).pretty()

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

🡪db.movies.find({$or:[{year:{$eq:2000}},{year:{$gt:2010}}]}).pretty()

**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."

🡪 db.movies.update({\_id:4},{$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."})

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."

🡪db.movies.update({\_id:5},{$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."}})

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

🡪db.movies.update({\_id:2},{$push:{actor:”Samuel L.Jackson”}})

**Text Search**

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

🡪db.movies.find(synopsis: {$regex:”Bilbo”}).pretty()

2. find all movies that have a synsopsis that contains the word "Gandalf"

🡪db.movies.find(synopsis:{$regex:”Gandalf”}).pretty()

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

🡪db.movies.find({$and:[{synopsis:{$regex:"Bilbo"}},{synopsis:{$not:/Gandalf/}}]})

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

🡪db.movies.find({$or:[{synopsis:{$regex:"dwarves"}},{synopsis:{$regex:"hobbit"}}]})

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

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

**Delete Documents**

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

🡪db.movies.delete({\_id: "5fff09f9f0a42c1a185b000c"})

2. delete the movie "Avatar”

🡪db.movies.delete({\_id:7})

**Querying related collections**

1. find all users

🡪db.users.find().pretty()

2. find all posts

🡪db.posts.find().pretty()

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

🡪db.posts.find({username:”GoodGuyGerg”}).pretty()

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

🡪db.posts.find({username:”GoodGuyGerg”}).pretty()

5. find all comments

🡪db.comments.find().pretty()

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

🡪db.comments.find(username:”GoodGuyGerg”)

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

🡪db.comments.find({username:”ScumbagSteve”})

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