## Worksheet- 10

Name: Sohit Kumar                          UID: 23BCS12492

Branch: CSE                                Section/Group: KRG 3-A

Semester: 5th                              Date of Performance:30/10/2025

Subject Name: ADBMS                        Subject Code: 23CSP-333

1. **Aim:** Perform CRUD Operations using No-SQL Database:MongoDB

## 2. Objective:

- Create and manage databases and collections.
- Perform **CRUD operations** — Create, Read, Update, and Delete documents.
- Use **aggregation** for grouping and summarizing data.
- Understand how to modify, query, and manage collections efficiently in MongoDB.
- Gain practical experience with real-world data handling through command-line operations.

## 3. Code:

```
// 1. Show all databases
show dbs

// 2. Create or switch to your database
use db_ravi_prince

// 3. Check current database
db

// 4. Create a collection (optional)
db.createCollection("movies")

// 5. Insert Operations
db.movies.insertOne({
title: "Dream Code",
director: "Ravi",
 year: 2010,
 rating: 8.8,
 genre: "Sci-Fi",
 features: ["Mind-bending", "Action"],
 added_by: "Ravi"
})
```

```
db.movies.insertMany([
  {
    title: "Galaxy Journey",
    director: "Prince",
    year: 2014,
    rating: 8.6,
    genre: "Sci-Fi",
    features: ["Space", "Drama"],
    added_by: "Prince"
  },
  {
    title: "Ocean World",
    director: "Prince",
    year: 2009,
    rating: 7.8,
    genre: "Adventure",
    features: ["3D", "Action"],
    added_by: "Prince"
  },
  {
    title: "The Shadow Hero",
    director: "Ravi",
    year: 2008,
    rating: 9.0,
    genre: "Action",
    features: ["Hero", "Crime"],
    added_by: "Ravi"
  }
])

// 6. Show all collections
show collections

// 7. Read Operations
db.movies.find().pretty()
db.movies.findOne()
db.movies.find({}, { title: 1, year: 1, _id: 0 })
db.movies.find({ genre: "Action" })
```

```
db.movies.find({ "details.language": "English" })

// 8. Update Operations
db.movies.updateOne(
  { title: "Ocean World" },
  { $set: { rating: 8.1, genre: "Sci-Fi Adventure" } }
)

db.movies.updateOne(
  { title: "Dream Code" },
  { $push: { features: "Thriller" } }
)

db.movies.updateOne(
  { title: "Dream Code" },
  { $pull: { features: "Action" } }
)

db.movies.updateMany(
  { director: { $in: ["Ravi", "Prince"] } },
  { $set: { language: "English" } }
)

db.movies.updateMany({}, { $unset: { added_by: "" } })
db.movies.updateMany({}, { $set: { quality: "Full HD" } })

db.movies.updateOne(
  { title: "Time Reversal" },
  { $set: { director: "Ravi", year: 2020, rating: 7.5 } },
  { upsert: true }
)

// 9. Delete Operations
db.movies.deleteOne({ title: "Ocean World" })
db.movies.deleteMany({ director: "Ravi" })
db.movies.deleteMany({})

// 10. Grouping (Aggregation)
db.movies.aggregate([
```

```
  { $group: { _id: "$director", total_movies: { $sum: 1 } } }
])


db.movies.aggregate([
  { $group: { _id: "$genre", total_movies: { $sum: 1 } } }
])


// 11. Drop collection
db.movies.drop()


// 12. Drop database
    db.dropDatabase()
```

**Output:**

```
admin           0.000GB
college         0.000GB
collegedb       0.000GB
config          0.000GB
db_company      0.000GB
education_db  0.000GB
library         0.000GB
local           0.000GB
switched to db db_ravi_prince
db_ravi_prince
{ "ok" : 1 }
```

```
{
        "acknowledged" : true,
        "insertedId" : ObjectId("690b89ab8090b47abf5502b5")
}
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("690b89ab8090b47abf5502b6"),
                ObjectId("690b89ab8090b47abf5502b7"),
                ObjectId("690b89ab8090b47abf5502b8")
        ]
}
movies
{
        "_id" : ObjectId("690b89ab8090b47abf5502b5"),
        "title" : "Dream Code",
        "director" : "Ravi",
        "year" : 2010,
        "rating" : 8.8,
        "genre" : "Sci-Fi",
        "features" : [
                "Mind-bending",
                "Action"
        ],
        "added_by" : "Ravi"
}
{
        "_id" : ObjectId("690b89ab8090b47abf5502b6"),
        "title" : "Galaxy Journey",
        "director" : "Prince",
        "year" : 2014,
        "rating" : 8.6,
        "genre" : "Sci-Fi",
        "features" : [
                "Space",
                "Drama"
        ],
        "added_by" : "Prince"
}
```

```
{
        "_id" : ObjectId("690b8ae81aec4a29ccce5507"),
        "title" : "Ocean World",
        "director" : "Prince",
        "year" : 2009,
        "rating" : 7.8,
        "genre" : "Adventure",
        "features" : [
                "3D",
                "Action"
        ],
        "added_by" : "Prince"
}
{
        "_id" : ObjectId("690b8ae81aec4a29ccce5508"),
        "title" : "The Shadow Hero",
        "director" : "Ravi",
        "year" : 2008,
        "rating" : 9,
        "genre" : "Action",
        "features" : [
                "Hero",
                "Crime"
        ],
        "added_by" : "Ravi"
}
{
        "_id" : ObjectId("690b8ae81aec4a29ccce5505"),
        "title" : "Dream Code",
        "director" : "Ravi",
        "year" : 2010,
        "rating" : 8.8,
        "genre" : "Sci-Fi",
        "features" : [
                "Mind-bending",
                "Action"
        ],
        "added_by" : "Ravi"
}
```

```
{ "title" : "Dream Code", "year" : 2010 }
{ "title" : "Galaxy Journey", "year" : 2014 }
{ "title" : "Ocean World", "year" : 2009 }
{ "title" : "The Shadow Hero", "year" : 2008 }
{ "_id" : ObjectId("690b8ae81aec4a29ccce5508"), "title" : "The Shadow Hero", "director" : "Ravi", "year" : 2008, "rating" : 9, "genre" : "Action", "features" : [ "Hero", "Crime" ], "added_by" : "Ravi" }
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
{ "acknowledged" : true, "matchedCount" : 4, "modifiedCount" : 4 }
{ "acknowledged" : true, "matchedCount" : 4, "modifiedCount" : 4 }
{ "acknowledged" : true, "matchedCount" : 4, "modifiedCount" : 4 }
{
        "acknowledged" : true,
        "matchedCount" : 0,
        "modifiedCount" : 0,
        "upsertedId" : ObjectId("690b8ae83b50a1c716af1df5")
}
{ "acknowledged" : true, "deletedCount" : 1 }
{ "acknowledged" : true, "deletedCount" : 3 }
{ "acknowledged" : true, "deletedCount" : 1 }
true
{ "ok" : 1 }
```

## 4. Learning Outcomes:

- Understand the structure and working of MongoDB as a NoSQL database.
- Create and manage databases and collections using MongoDB commands.
- Perform CRUD (Create, Read, Update, Delete) operations on documents.
- Apply aggregation operations for data grouping and analysis.
- Gain practical skills in handling and modifying data using MongoDB shell.