# TASK – 8

## CRUD Operations in Document Databases

### AIM:

To perform Mongoose using NPM design on MongoDB by designing a University document database and performing CRUD operations such as creating, inserting, querying, finding, updating, and deleting records.

### STEPS:

Step 1: Install MongoDB using the following link: https://www.mongodb.com/try/download/community

Step 2: Install Mongosh using the following link: https://www.mongodb.com/docs/mongodb-shell/#download-and-install-mongosh

Step 3: Add the MongoDB Shell binary’s location to your PATH environment variable:

3.1 Open the Control Panel.

3.2 In the System and Security category, click System.

3.3 Click Advanced system settings → The System Properties modal displays.

3.4 Click Environment Variables.

3.5 In the System variables section, select Path and click Edit.

3.6 Click New and add the file path to your mongosh binary.

3.7 Click OK to confirm your changes.

Step 4: To confirm that your PATH is correctly configured, open Command Prompt and type: mongosh --help

Step 5: Open the Mongo Shell from the following location: C:\Program Files\MongoDB\Server\bin\mongod.exe

Step 6: Type and execute the CRUD commands given below.

### CRUD OPERATIONS:

#### Creating a Collection

> use UniversityDB  
switched to db UniversityDB  
  
> db.createCollection("University")  
{ "ok" : 1 }

#### Inserting a Single Document

> db.University.insertOne({  
 UniversityID: "U001",  
 Name: "Anna University",  
 Location: "Chennai",  
 Established\_Year: 1978,  
 Chancellor: "Hon. R. N. Ravi",  
 Vice\_Chancellor: "Dr. R. Velraj",  
 Contact\_Email: "info@annauniv.edu",  
 Phone: 04422357000  
});  
Output:  
{  
 "acknowledged" : true,  
 "insertedId" : ObjectId("651cf1726ebbfe7993adf909")  
}

#### Inserting Multiple Documents

> db.University.insertMany([  
 {  
 UniversityID: "U002",  
 Name: "University of Madras",  
 Location: "Chennai",  
 Established\_Year: 1857,  
 Chancellor: "Hon. R. N. Ravi",  
 Vice\_Chancellor: "Dr. S. Gowri",  
 Contact\_Email: "registrar@unom.ac.in",  
 Phone: 04425361073  
 },  
 {  
 UniversityID: "U003",  
 Name: "Bharathiar University",  
 Location: "Coimbatore",  
 Established\_Year: 1982,  
 Chancellor: "Hon. R. N. Ravi",  
 Vice\_Chancellor: "Dr. P. Kaliraj",  
 Contact\_Email: "contact@buc.edu.in",  
 Phone: 04222422222  
 },  
 {  
 UniversityID: "U004",  
 Name: "Manonmaniam Sundaranar University",  
 Location: "Tirunelveli",  
 Established\_Year: 1990,  
 Chancellor: "Hon. R. N. Ravi",  
 Vice\_Chancellor: "Dr. P. Sabitha",  
 Contact\_Email: "info@msuniv.ac.in",  
 Phone: 04622233114  
 },  
 {  
 UniversityID: "U005",  
 Name: "Periyar University",  
 Location: "Salem",  
 Established\_Year: 1997,  
 Chancellor: "Hon. R. N. Ravi",  
 Vice\_Chancellor: "Dr. R. Jagannathan",  
 Contact\_Email: "registrar@periyaruniv.ac.in",  
 Phone: 04272234000  
 }  
]);  
Output:  
{  
 "acknowledged" : true,  
 "insertedIds" : [ObjectId("651ceee36ebbfe7993adf904"), ObjectId("651ceee36ebbfe7993adf905"), ObjectId("651ceee36ebbfe7993adf906"), ObjectId("651ceee36ebbfe7993adf907")]  
}

#### Displaying All Documents

> db.University.find()  
Output:  
{ "\_id" : ObjectId("651ceee36ebbfe7993adf904"), "UniversityID" : "U002", "Name" : "University of Madras", "Location" : "Chennai", "Established\_Year" : 1857, "Chancellor" : "Hon. R. N. Ravi", "Vice\_Chancellor" : "Dr. S. Gowri", "Contact\_Email" : "registrar@unom.ac.in", "Phone" : 04425361073 }

#### Displaying Documents in Readable Format

> db.University.find().pretty()  
Output:  
{  
 "\_id" : ObjectId("651ceee36ebbfe7993adf904"),  
 "UniversityID" : "U002",  
 "Name" : "University of Madras",  
 "Location" : "Chennai",  
 "Established\_Year" : 1857,  
 "Chancellor" : "Hon. R. N. Ravi",  
 "Vice\_Chancellor" : "Dr. S. Gowri",  
 "Contact\_Email" : "registrar@unom.ac.in",  
 "Phone" : 04425361073  
}

#### Updating a Document

> db.University.updateOne(  
 { UniversityID: "U005" },  
 { $set: { Vice\_Chancellor: "Dr. S. Kannan", Phone: 04272234011 } }  
);  
Output:  
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }

#### Deleting a Document

> db.University.deleteOne({ UniversityID: "U003" })  
Output:  
{ "acknowledged" : true, "deletedCount" : 1 }

#### Display Remaining Documents

> db.University.find().pretty()  
Output:  
{  
 "\_id" : ObjectId("651ceee36ebbfe7993adf904"),  
 "UniversityID" : "U002",  
 "Name" : "University of Madras",  
 "Location" : "Chennai",  
 "Established\_Year" : 1857,  
 "Chancellor" : "Hon. R. N. Ravi",  
 "Vice\_Chancellor" : "Dr. S. Gowri",  
 "Contact\_Email" : "registrar@unom.ac.in",  
 "Phone" : 04425361073  
},  
{  
 "\_id" : ObjectId("651ceee36ebbfe7993adf907"),  
 "UniversityID" : "U005",  
 "Name" : "Periyar University",  
 "Location" : "Salem",  
 "Established\_Year" : 1997,  
 "Chancellor" : "Hon. R. N. Ravi",  
 "Vice\_Chancellor" : "Dr. S. Kannan",  
 "Contact\_Email" : "registrar@periyaruniv.ac.in",  
 "Phone" : 04272234011  
}

### RESULT:

Thus, CRUD operations using NPM design on MongoDB for a University document database — including creation, insertion, querying, updating, and deletion — were successfully performed.