DOP: / /2023 **DOS: / /2023**

Experiment No: 06

Title: Build Crud Operation using MongoDB.

Theory:

What is MongoDB:

- MongoDB is an open-source document database that provides high performance, high availability, and automatic scaling.
- In simple words, you can say that - Mongo DB is a document-oriented database. It is an open source product, developed and supported by a company named 10gen.
- MongoDB is available under General Public license for free, and it is also available under Commercial license from the manufacturer.
- The manufacturing company 10gen has defined MongoDB as:
- "MongoDB is a scalable, open source, high performance, document-oriented database." - 10gen

Features of MongoDB:

These are some important features of MongoDB:

1. Support ad hoc queries: In MongoDB, you can search by field, range query and it also supports regular expression searches.

2. Indexing: You can index any field in a document.

3. Replication: MongoDB supports Master Slave replication.

4. Duplication of data: MongoDB can run over multiple servers. The data is duplicated to keep the system up and also keep its running condition in case of hardware failure.

5. Load balancing: It has an automatic load balancing configuration because of data placed in shards.

6. Supports map reduce and aggregation tools.

7. Uses JavaScript instead of Procedures.

8. It is a schema-less database written in C++.

9. Provides high performance.

10. Stores files of any size easily without complicating your stack.

11. Easy to administer in the case of failures.

CRUD operations

CRUD (Create, Read, Update, Delete) operations allow you to work with the data stored in MongoDB. The CRUD operation documentation is categorized into two sections: Read Operations find and return documents stored within your MongoDB database. Write Operations to insert, modify, or delete documents in your MongoDB database.

Create Operations –

Method	Description
db.collection.insertOne()	It is used to insert a single document in the collection.
db.collection.insertMany()	It is used to insert multiple documents in the collection.
db.createCollection()	It is used to create an empty collection.

Read Operations –

Method	Description
db.collection.find()	It is used to retrieve documents from the collection.

Update Operations –

Method	Description
db.collection.updateOne()	It is used to update a single document in the collection that satisfy the given criteria.
db.collection.updateMany()	It is used to update multiple documents in the collection that satisfy the given criteria.
db.collection.replaceOne()	It is used to replace single document in the collection that satisfy the given criteria

Delete Operations –

Method	Description
db.collection.deleteOne()	It is used to delete a single document from the collection that satisfy the given criteria.
db.collection.deleteMany()	It is used to delete multiple documents from the collection that satisfy the given criteria.

```
const MongoClient = require("mongodb").MongoClient;
const url = "mongodb://localhost:27017";
MongoClient.connect(url, async function (err, db) {
    if(err) throw err;
    var dbo = db.db("Squad");
    console.log("successfully created database Squad");
    var insert = [
        { "name":"Mayur" },
        {"name ":"Ashish"},
        { "name": "Mitesh"},
        { "name":"kunal" },
        {"name":"Malvika"},
        { "name": "Anuj" }
    ]
    dbo.collection("names").insertMany(insert, async function (err) {
        if (err) throw err;
        console.log("Successfully Inserted data : ", insert)
    })
    dbo.collection("names").deleteOne({ "name": "Anuj" })
    console.log(await dbo.collection("names").find({}).toArray())
console.log("name deleted successfully ")
})
```

```
PS C:\data\db> node crud.js
successfully created database Squad
Successfully Inserted data :  [
PS C:\data\db> node crud.js
successfully created database Squad
Successfully Inserted data :  [
  { name: 'Mayur', _id: new ObjectId("6263a0a1f6ceadb265a07aec") },
  { 'name ': 'Ashish', _id: new ObjectId("6263a0a1f6ceadb265a07aed") },
  { name: 'Mitesh', _id: new ObjectId("6263a0a1f6ceadb265a07aee") },
  { name: 'kunal', _id: new ObjectId("6263a0a1f6ceadb265a07aef") },
  { name: 'Malvika', _id: new ObjectId("6263a0a1f6ceadb265a07af0") },
  { name: 'Anuj', _id: new ObjectId("6263a0a1f6ceadb265a07af1") }
]
[
  { _id: new ObjectId("6263a078c5a7d41e2289d9e8"), name: 'Mayur' },
  { _id: new ObjectId("6263a078c5a7d41e2289d9e9"), 'name ': 'Ashish' },
  { _id: new ObjectId("6263a078c5a7d41e2289d9ea"), name: 'Mitesh' },
  { _id: new ObjectId("6263a078c5a7d41e2289d9eb"), name: 'kunal' },
  { _id: new ObjectId("6263a078c5a7d41e2289d9ec"), name: 'Malvika' },
  { _id: new ObjectId("6263a0a1f6ceadb265a07aec"), name: 'Mayur' },
  { _id: new ObjectId("6263a0a1f6ceadb265a07aed"), 'name ': 'Ashish' },
  { _id: new ObjectId("6263a0a1f6ceadb265a07aee"), name: 'Mitesh' },
  { _id: new ObjectId("6263a0a1f6ceadb265a07aef"), name: 'kunal' },
  { _id: new ObjectId("6263a0a1f6ceadb265a07af0"), name: 'Malvika' }
]
name deleted successfully
```

Conclusion: - In this program, we learn the crud operation i.e. (create, read, update, delete) in MongoDB using NodeJS.