TASK 9

TASK 10: CRUD operations in Document databases

Perform Mongoose using NPM design on MongoDB designing document database and performing CRUD operations like creating, inserting, querying, finding, removing operations.

AIM:

To Perform Mongoose using NPM design on MongoDB designing document database and performing CRUD operations like creating, inserting, querying, finding, removing operations

STEPS:

Step 1)install Mongo db using following link

https://www.mongodb.com/try/download/community

Step 2)install Mongosh using the below link

https://www.mongodb.com/docs/mongodb-shell/#download-and-install-mongosh

Step 3)To add the MongoDB Shell binary's location to your PATH environment variable:

Open the Control Panel.

In the System and Security category, click System.

Click Advanced system settings. The System Properties modal displays.

Click Environment Variables.

In the System variables section, select path and click Edit. The Edit environment variable modal displays.

Click New and add the filepath to your mongosh binary.

Click OK to confirm your changes. On each other modal, click OK to confirm your changes.

To confirm that your PATH environment variable is correctly configured to find mongosh, open a command prompt and enter the mongosh --help command.

If your PATH is configured correctly, a list of valid commands displays.

Step 4)Open mongo shell 4.0 from c:\programfiles\mongoDB\server\bin\mongod.exe

Step 5)Type the CRUD(CREATE READ UPDATE DELETE) COMMANDS GIVEN IN TEXT FILE.

CRUD OPERATIONS:

```
 db.createCollection("mylab3")
{ "ok" : 1 }
>
db.mylab3.insertOne({item:"canvas",qty:100,tags:["cotton"],size:{h:28,w:35.5,uom:"cm"}})
{
"acknowledged" : true,
"insertedId" : ObjectId("627d13acc73990c074e6397c")
}
> db.mylab3.find({item:"canvas"})
{ "_id" : ObjectId("627d13acc73990c074e6397c"), "item" : "canvas", "qty" : 100, "tags" : [
"cotton" ], "size" : { "h" : 28, "w" : 35.5, "uom" : "cm" } }
>
db.mylab3.insertMany([{item:"journal",qty:25,tags:["blank","red"],size:{h:14,w:21,uom:"cm
"}},{item:"mat",qty:85,tags:["gray"],size:{h:27.9,w:35.5,uom:"cm"}},{item:"mousepad",qty:
25,tags:["gel","blue"],size:{h:19,w:22.85,uom:"cm"}}])
{
"acknowledged" : true,
"insertedIds" : [
        ObjectId("627d1598c73990c074e6397d"),
        ObjectId("627d1598c73990c074e6397e"),
        ObjectId("627d1598c73990c074e6397f")
    ]
}
> db.mylab3.find({},{item:1,qty:1})
{ "_id" : ObjectId("627d13acc73990c074e6397c"), "item" : "canvas", "qty" : 100 }
{ "_id" : ObjectId("627d1598c73990c074e6397d"), "item" : "journal", "qty" : 25 }
{ "_id" : ObjectId("627d1598c73990c074e6397e"), "item" : "mat", "qty" : 85 }
{ "_id" : ObjectId("627d1598c73990c074e6397f"), "item" : "mousepad", "qty" : 25 }
> db.mylab.find({},{item:1,qty:1}).pretty()
{
"_id" : ObjectId("627d13acc73990c074e6397c"),
"item" : "canvas",
```

```
"qty" : 100
}
{
"_id" : ObjectId("627d1598c73990c074e6397d"),
"item" : "journal",
"qty" : 25
}
{ "_id" : ObjectId("627d1598c73990c074e6397e"), "item" : "mat", "qty" : 85 }
{
"_id" : ObjectId("627d1598c73990c074e6397f"),
"item" : "mousepad",
"qty" : 25
}
> db.mylab3.find({item:"canvas"}).pretty().sort({item:-1})
{
"_id" : ObjectId("627d13acc73990c074e6397c"),
"item" : "canvas",
"qty" : 100,
"tags" : [
"cotton"
    ],
"size" : {
"h" : 28,
"w" : 35.5,
"uom" : "cm"
    }
}
> db.myLab3.deleteOne({ item: "journal" });
...
...
> db.mylab3.find({},{item:1,qty:1}).pretty()
{
"_id" : ObjectId("627d13acc73990c074e6397c"),
"item" : "canvas",
```

```
"qty" : 100
}
{
"_id" : ObjectId("627d1598c73990c074e6397d"),
"item" : "journal",
"qty" : 25
}
{ "_id" : ObjectId("627d1598c73990c074e6397e"), "item" : "mat", "qty" : 85 }
{
"_id" : ObjectId("627d1598c73990c074e6397f"),
"item" : "mousepad",
"qty" : 25
}
>
```

Result:

Thus CRUD using NPM design on MongoDB designing document database and performing CRUD operations like creating, inserting, querying, finding, removing operations are performed.