

**Experiment No. 04**

Semester	B.E. Semester VII – Computer Engineering
Subject	Big Data Analysis
Subject Professor In-charge	Prof. Pankaj Vanvari
Lab Professor In-charge	Dr. Umesh Kulkarni
Academic Year	2024-25
Student Name	Deep Salunkhe
Roll Number	21102A0014

**Title:** To install and configure MongoDB to execute NoSQL commands

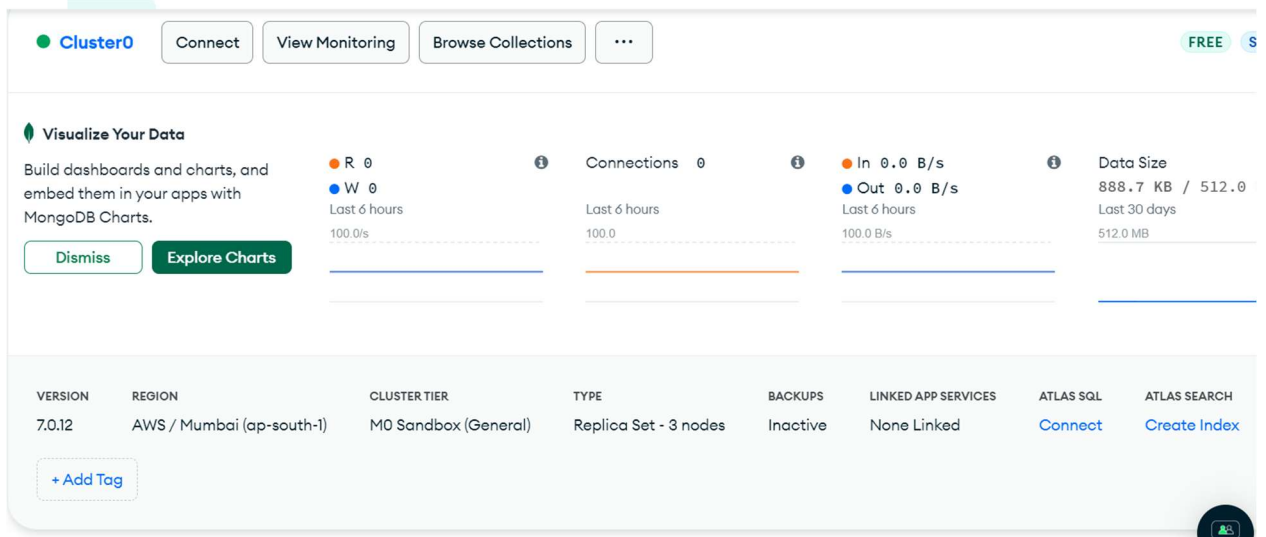
```

1  const mongoose=require('mongoose');
2  const env=require('dotenv');
3  env.config();
4  const connectDB=async()=>{
5    try{
6      await mongoose.connect(process.env.MONGOKEY,{
7        useNewUrlParser:true,
8        useUnifiedTopology:true
9      });
10     console.log('MongoDB connection SUCCESS');
11   }catch(error){
12     console.error(error);
13   }
14 }
15
16 module.exports=connectDB;
  
```

client  
server  
db  
 connections.js  
 schema.js  
 middleware  
routes  
 index.js  
 package-lock.json  
 package.json  
ws\_server  
 index.js  
 index1.js  
 index3.js  
 package-lock.json  
 package.json

NORMAL NvimTree\_1  
E21: Cannot make changes, 'modifiable' is off

app 12 %



### Connect to Cluster0

Set up connection security Choose a connection method Connect

#### Connecting with MongoDB Driver

- Select your driver and version**

We recommend installing and using the latest driver version.

Driver: C++ Version: 3.7.x or later
- Install your driver**

[View MongoDB C++ Driver installation instructions.](#)
- Add your connection string into your application code**

Use this connection string in your application

☐ View full code sample

```
mongodb+srv://<db_username>:<db_password>@cluster0.nowues7.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0
```

Replace <db\_password> with the password for the <db\_username> database user. Ensure any option params are URL encoded.

#### RESOURCES

[Get started with the C++ Driver](#) [Access your Database Users](#) [Troubleshoot Connections](#)

```

  client
  server
  db
    connections.js
    schema.js
  middleware
  routes
    conference.js
    users.js
    index.js
  package-lock.json
  package.json
  ws_server
    index.js
    index1.js
    index3.js
    package-lock.json
    package.json
  connections.js
  users.js
37 } catch (error) {
38   return res.status(400).json({ error: "Invalid data" });
39 }
40
41 // Check if the user already exists
42 const existingUser = await User.findOne({ email: email });
43
44 if (existingUser) {
45   console.log("User already exists");
46   return res.status(400).json({ error: "User already exists" });
47 }
48
49 // Hash the password
50 const hashedPassword = bcrypt.hashSync(password, salt);
51
52 // Create a new user
53 const newUser = new User({
54   email: email,
55   passwordHash: hashedPassword,
56 });
57
58 // Save the user to the database
59 try {
60   const savedUser = await newUser.save();
61   res.status(200).json({ message: "User created successfully" });
62 } catch (error) {
63   res.status(500).json({ error: "Internal server error" });
64 }
65 }
66
  NORMAL users.js deepsalunkhe
  F21: Cannot make changes 'modifiable' is off
  LSP ~ GitHub Copilot app 30 %
```

**Data Services** Charts

+ Create Database

Search Namespaces

- 0\_100
- AKCRH
- AngleEye
- BlockMeet**
  - conferences
  - users**
- BroFood
- OCL-Viva
- HasherDemo
- Health\_check
- Hostel\_bank

### BlockMeet.users

STORAGE SIZE: 36KB LOGICAL DATA SIZE: 5.16KB TOTAL DOCUMENTS: 9 INDEXES TOTAL SIZE: 36KB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

Generate queries from natural language in Compass

INSERT DOCUMENT

Filter Type a query: { field: 'value' } Reset Apply Options

QUERY RESULTS: 1-9 OF 9

```
{
  "_id": ObjectId("66dfc588ca6df55e629cb370"),
  "email": "deep1@gmail.com",
  "passwordHash": "$2b$13$2oQymjzBQ80.ZMBH7vXWk.1LyAB/81MqWEIFhLzUn0b3GULwH.",
  "conferences": Array (3)
  __v: 3
}
```

```
{
  "_id": ObjectId("66dfc5a2ca6df55e629cb374"),
  "email": "deep2@gmail.com",
  "passwordHash": "$2b$13$2oQymjzBQ80.ZMBH7vXWk.mrucJ4aSpEGdJtYbSUJIB1GABoaKNvu"
}
```