

Task 8: Using MongoDB with Node.js

Objective:

Connect a Node.js application to a MongoDB database and perform basic CRUD operations. Ensure data is correctly inserted, read, updated, and deleted in the database.

Prerequisites:

- Basic understanding of JavaScript and Node.js.
- MongoDB installation and setup.

Concepts:

- **Connecting to MongoDB:**
 - Use the `mongodb` package to connect to a MongoDB database.
 - MongoDB URL usually looks like: `mongodb://localhost:27017/mydatabase`.

Example:

```
// dbConnection.js
const { MongoClient } = require('mongodb');
const url = 'mongodb://localhost:27017';
const dbName = 'mydatabase';

const client = new MongoClient(url, { useNewUrlParser: true, useUnifiedTopology: true });

async function connectDB() {
  await client.connect();
  console.log('Connected successfully to MongoDB');
  const db = client.db(dbName);
  return db;
}

module.exports = connectDB;
```

- **CRUD Operations:**
 - **Create:** Insert a document into a collection.
 - **Read:** Retrieve documents from a collection.
 - **Update:** Modify existing documents in a collection.
 - **Delete:** Remove documents from a collection.

Example:

```
// crudOperations.js
const connectDB = require('./dbConnection');

async function performCRUDOperations() {
  const db = await connectDB();
  const collection = db.collection('users');

  // Create
  const insertResult = await collection.insertOne({ name: 'John Doe', age: 30 });
  console.log('Inserted document:', insertResult.ops[0]);
}

performCRUDOperations().catch(console.error);
```

Setup:

- **Install MongoDB:**

- Ensure MongoDB is installed and running on your machine. You can access the instructions here:



Instructions to Set Up MongoDB Atlas and MongoDB Compass

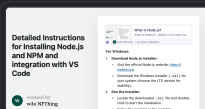
Step 1: Set Up MongoDB Atlas — Sign Up for MongoDB Atlas: — Go to the MongoDB Atla...



w3o NFThing Last Edited 7/2/2024

- **Install Node.js and the mongodb package:**

- Ensure Node.js is installed on your machine. You can access the instructions here:



Detailed Instructions for Installing Node.js and NPM and integration with VS...

For Windows: — Download Node.js Installer: — Visit the official Node.js website: <https://nodejs.org/en/>



w3o NFThing Last Edited 7/3/2024

- Install the mongodb package using npm:

```
npm install mongodb
```

Tasks:

1. Connecting to MongoDB:

- **Task:**

- Create a file named `dbConnection.js` and write code to connect to a MongoDB database using the `mongodb` package.
- Export the database connection function.

- **Outcome:**

- Ensure the connection function connects to the MongoDB database successfully.

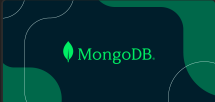



2. Performing CRUD Operations:

- **Task:**
 - Create a file named `crudOperations.js` and write code to:
 - Insert a document into a collection.
 - Retrieve a document from a collection.
 - Update a document in a collection.
 - Delete a document from a collection.
 - Use the database connection function from `dbConnection.js`.
- **Outcome:**
 - Ensure the CRUD operations are performed correctly and log the results to the console.

Instructions:

- **Perform the following tasks:**
 - Write the required code in separate files (`dbConnection.js` , `crudOperations.js`).
 - Run each file using Node.js to ensure the code executes without errors and demonstrates the use of MongoDB with Node.js.

Resources:



-  **MongoDB Node Driver**
Welcome to the documentation site for the official MongoDB Node.js driver. — You can a...
 <https://docs.mongodb.com/drivers/node>
-  **MongoDB Documentation**
Welcome to the official MongoDB Documentation. — Whether you're a developer, databa...
 <https://docs.mongodb.com/>

Videos:

Connect



Connect Node with MongoDB

  Code With Yousaf 5:51min 131,446 Views 1,168 Likes

GitHub Instructions:

1. Open in Visual Studio Code:

- After clicking on the "Open in Visual Studio Code" button from the GitHub Classroom confirmation page, Visual Studio Code (VSCode) will open the repository directly.
- If prompted, select "Open" or "Allow" to open the repository in VSCode.

2. Complete the Task:

- Write your solution in the respective files (`dbConnection.js` , `crudOperations.js`).

3. Run and Test Your Code:

- Run your code to ensure it works correctly. Use the following command for each file:

```
node <filename>.js
```

4. Commit Your Changes:

- Commit your changes with a meaningful message:

```
git commit -m "Completed MongoDB with Node.js task"
```

5. Push Your Changes to Your Forked Repository:

- Push your changes to your forked repository:

```
git push origin main
```

6. Create a Pull Request:

- Go to your forked repository on GitHub.
- Click on the "Pull Requests" tab.
- Click the "New Pull Request" button.
- Ensure the base repository is the original template repository and the base branch is `main`.
- Ensure the head repository is your forked repository and the compare branch is `main`.
- Click "Create Pull Request".
- Add a title and description for your pull request and submit it.

Summary of Commands:

```
# Fork the repository on GitHub

# Clone the forked repository
git clone https://github.com/your-github-username/repository-name.git
cd repository-name

# Complete the task by writing and running the code in the specified files

# Add, commit, and push your changes
git commit -m "Completed MongoDB with Node.js task"
git push origin main

# Create a pull request on GitHub
```