To work with [Node.js](https://Node.js) and MongoDB, you’ll need to follow several steps to set up your environment, connect to your database, and perform CRUD (Create, Read, Update, Delete) operations. Here’s a step-by-step guide to get you started:

**Step 1: Install** [**Node.js**](https://Node.js) **and MongoDB**

1. [**Node.js**](https://Node.js):
	* Download and install [Node.js](https://Node.js) from the [official website](https://nodejs.org/).
	* Verify the installation by running:

node -v

npm -v

**2- MongoDB**:

* Download and install MongoDB from the [official website](https://www.mongodb.com/try/download/community).
* Start the MongoDB server by running:

Mongod

**Step 2: Set Up Your Project**

1. **Create** a new project directory:

mkdir my-node-mongo-project

cd my-node-mongo-project

1. Initialize the project and create a package.json file:

npm init -y

**Step 3: Install Dependencies**

Install the necessary packages, including express for the server and mongoose for MongoDB interaction:

npm install express mongoose

### ****Step 4: Create a Server****

Create a new file named server.js and set up a basic Express server:

### const express = require('express');

### const mongoose = require('mongoose');

### const path = require('path');

### const app = express();

### const port = 3000;

### // Middleware

### app.use(express.json());

### app.use(express.urlencoded({ extended: true }));

### app.use(express.static(path.join(\_\_dirname, 'public')));

### // Connect to MongoDB

### mongoose.connect('mongodb://localhost:27017/mydatabase', {

###  useNewUrlParser: true,

###  useUnifiedTopology: true,

### });

### const db = mongoose.connection;

### db.on('error', console.error.bind(console, 'connection error:'));

### db.once('open', function() {

###  console.log('Connected to MongoDB');

### });

### // Define the root route to serve the HTML form

### app.get('/', (req, res) => {

###  res.sendFile(path.join(\_\_dirname, 'public', 'index.html'));

### });

### // User model

### const User = require('./models/User');

### // Handle form submissions to add new users

### app.post('/users', async (req, res) => {

###  try {

###  const user = new User(req.body);

###  await user.save();

###  res.redirect('/users');

###  } catch (err) {

###  res.status(400).send(err);

###  }

### });

### // Get all users and display them

### app.get('/users', async (req, res) => {

###  try {

###  const users = await User.find();

###  res.json(users); // Change this line if you want to render a HTML view

###  } catch (err) {

###  res.status(500).send(err);

###  }

### });

### // Start the server

### app.listen(port, () => {

###  console.log(`Server is running on http://localhost:${port}`);

### });});

### ****Step**** 5: Define a Mongoose Schema and Model

### const mongoose = require('mongoose');

### const userSchema = new mongoose.Schema({

###  name: String,

###  email: String,

###  age: Number

### });

### const User = mongoose.model('User', userSchema);

### module.exports = User;

### Step 6: Run Your Server

### node server.js

### Step 7: Create the HTML Form

### Create an index.html file in your public directory with the following content:

### <!DOCTYPE html>

### <html lang="en">

### <head>

###  <meta charset="UTF-8">

###  <meta name="viewport" content="width=device-width, initial-scale=1.0">

###  <title>Add User</title>

### </head>

### <body>

###  <h1>Add User</h1>

###  <form action="/users" method="POST">

###  <label for="name">Name:</label>

###  <input type="text" id="name" name="name" required><br><br>

###

###  <label for="email">Email:</label>

###  <input type="email" id="email" name="email" required><br><br>

###

###  <label for="age">Age:</label>

###  <input type="number" id="age" name="age" required><br><br>

###

###  <input type="submit" value="Add User">

###  </form>

### </body>

### </html>