On terminal:

node -v

npm -v

mongosh

1. Create a folder:
2. mkdir curd-api
3. cd curd-api
4. Initialize the project:
5. npm init -y
6. Install required packages:
7. npm install express mongoose dotenv
8. npm install --save-dev nodemon
9. Modify package.json:
10. "scripts": {
11. "start": "nodemon index.js"
12. }
13. 📁 curd-api  
    ├── index.js  
    ├── .env  
    ├── models  
    │ └── Item.js  
    ├── routes  
    │ └── items.js  
    ├── package.json
14. ✅ Step 4: Create .env File
15. Create a file named .env:
16. PORT=5000
17. MONGO\_URI=mongodb://127.0.0.1:27017/curd\_api\_db
18. ✅ Step 5: index.js (Main Server File)
19. const express = require('express');
20. const mongoose = require('mongoose');
21. const dotenv = require('dotenv');
22. dotenv.config();
23. const app = express();
24. app.use(express.json());
25. const itemRoutes = require('./routes/items');
26. app.use('/api/items', itemRoutes);
27. mongoose.connect(process.env.MONGO\_URI, {
28. useNewUrlParser: true,
29. useUnifiedTopology: true
30. })
31. .then(() => console.log('MongoDB connected'))
32. .catch(err => console.log(err));
33. const PORT = process.env.PORT || 5000;
34. app.listen(PORT, () => console.log(`Server started on port ${PORT}`));
35. ✅ Step 6: models/Item.js
36. const mongoose = require('mongoose');
37. const ItemSchema = new mongoose.Schema({
38. name: {
39. type: String,
40. required: true
41. },
42. quantity: {
43. type: Number,
44. default: 0
45. }
46. });
47. module.exports = mongoose.model('Item', ItemSchema);
48. ✅ Step 7: routes/items.js
49. const express = require('express');
50. const router = express.Router();
51. const Item = require('../models/Item');
52. // CREATE
53. router.post('/', async (req, res) => {
54. try {
55. const newItem = new Item(req.body);
56. const saved = await newItem.save();
57. res.status(201).json(saved);
58. } catch (err) {
59. res.status(400).json({ error: err.message });
60. }
61. });
62. // READ ALL
63. router.get('/', async (req, res) => {
64. try {
65. const items = await Item.find();
66. res.json(items);
67. } catch (err) {
68. res.status(500).json({ error: err.message });
69. }
70. });
71. // UPDATE
72. router.put('/:id', async (req, res) => {
73. try {
74. const updatedItem = await Item.findByIdAndUpdate(
75. req.params.id,
76. req.body,
77. { new: true }
78. );
79. res.json(updatedItem);
80. } catch (err) {
81. res.status(400).json({ error: err.message });
82. }
83. });
84. // DELETE
85. router.delete('/:id', async (req, res) => {
86. try {
87. const deletedItem = await Item.findByIdAndDelete(req.params.id);
88. res.json({ message: 'Deleted', item: deletedItem });
89. } catch (err) {
90. res.status(500).json({ error: err.message });
91. }
92. });
93. module.exports = router;
94. ✅ Step 8: Run the API
95. npm start
96. You should see:
97. MongoDB connected
98. Server started on port 5000

| **Operation** | **Method** | **URL** |
| --- | --- | --- |
| Create | POST | http://localhost:5000/api/items |
| Read | GET | http://localhost:5000/api/items |
| Update | PUT | http://localhost:5000/api/items/<id> |
| Delete | DELETE | http://localhost:5000/api/items/<id> |

**🟢 Create Item (POST)**

* **URL**: http://localhost:5000/api/items
* **Method**: POST
* **Body** → raw → JSON:

json

CopyEdit

{

"name": "Apples",

"quantity": 5

}

* Click **Send**

AJAX:

Register.html

<!DOCTYPE html>

<html>

<head><title>Register</title></head>

<body>

<h2>User Registration</h2>

<form id="form">

<input name="name" placeholder="Name" required><br><br>

<input name="email" type="email" placeholder="Email" required><br><br>

<input name="password" type="password" placeholder="Password" required><br><br>

<button type="submit">Register</button>

</form>

<script>

document.getElementById("form").addEventListener("submit", function (e) {

e.preventDefault();

const formData = new FormData(this);

const user = {

name: formData.get("name"),

email: formData.get("email"),

password: formData.get("password")

};

// Simulating AJAX POST with setTimeout

setTimeout(() => {

let users = JSON.parse(localStorage.getItem("users") || "[]");

users.push(user);

localStorage.setItem("users", JSON.stringify(users));

alert("User registered!");

window.location.href = "list.html";

}, 300);

});

</script>

</body>

</html>

//list.html///////////////////////

<!DOCTYPE html>

<html>

<head><title>Registered Users</title></head>

<body>

<h2>Registered Users</h2>

<table border="1" cellpadding="10">

<tr><th>Name</th><th>Email</th><th>Password</th></tr>

<tbody id="list"></tbody>

</table>

<script>

const users = JSON.parse(localStorage.getItem("users") || "[]");

users.forEach(u => {

document.getElementById("list").innerHTML += `<tr><td>${u.name}</td><td>${u.email}</td><td>${u.password}</td></tr>`;

});

</script>

</body>

</html>

http://localhost:8080/register.html