**MongoDB Activity(22/07/2025)**

**IN GitHub**

**Using GitHub Codespaces**

Steps to Execute in Codespaces Terminal

1. Open GitHub Codespace
2. In the Codespace terminal, run:

**mkdir crud-demo**

**cd crud-demo**

**npm init -y**

**npm install express mongoose dotenv**

**mkdir models**

**mkdir routes**

**touch models/user.js**

**touch routes/userRoutes.js**

**touch server.js**

**touch .env**

1. **Add Code to Each File**

**models/user.js**

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

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

**name: { type: String, required: true },**

**email: { type: String, required: true, unique: true },**

**age: { type: Number }**

**});**

**module.exports = mongoose.model('User', userSchema);**

**routes/userRoutes.js**

const express = require('express');

const router = express.Router();

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

// CREATE

router.post('/', async (req, res) => {

    try {

        const newUser = await User.create(req.body);

        res.status(201).json(newUser);

    } catch (err) {

        res.status(400).json({ error: err.message });

    }

});

// READ ALL

router.get('/', async (req, res) => {

    const users = await User.find();

    res.json(users);

});

// READ ONE

router.get('/:id', async (req, res) => {

    try {

        const user = await User.findById(req.params.id);

        if (!user) return res.status(404).json({ error: 'User not found' });

        res.json(user);

    } catch (err) {

        res.status(400).json({ error: err.message });

    }

});

// UPDATE

router.put('/:id', async (req, res) => {

    try {

        const updatedUser = await User.findByIdAndUpdate(req.params.id, req.body, { new: true });

        if (!updatedUser) return res.status(404).json({ error: 'User not found' });

        res.json(updatedUser);

    } catch (err) {

        res.status(400).json({ error: err.message });

    }

});

// DELETE

router.delete('/:id', async (req, res) => {

    try {

        const deletedUser = await User.findByIdAndDelete(req.params.id);

        if (!deletedUser) return res.status(404).json({ error: 'User not found' });

        res.json({ message: 'User deleted' });

    } catch (err) {

        res.status(400).json({ error: err.message });

    }

});

module.exports = router;

**server.js**

require('dotenv').config();

const express = require('express');

const mongoose = require('mongoose');

const userRoutes = require('./routes/userRoutes');

const app = express();

app.use(express.json());

// Request Logger

app.use((req, res, next) => {

    console.log(`[${new Date().toISOString()}] ${req.method} ${req.url}`);

    next();

});

// MongoDB Connection

mongoose.connect(process.env.MONGO\_URI, {

    useNewUrlParser: true,

    useUnifiedTopology: true

})

.then(() => console.log('✅ MongoDB Connected'))

.catch(err => console.error('❌ MongoDB Error:', err));

// Routes

app.use('/users', userRoutes);

// Start Server

const PORT = process.env.PORT || 3000;

app.listen(PORT, () => {

    console.log(`🚀 Server running on http://localhost:${PORT}`);

});

**.env(code)**

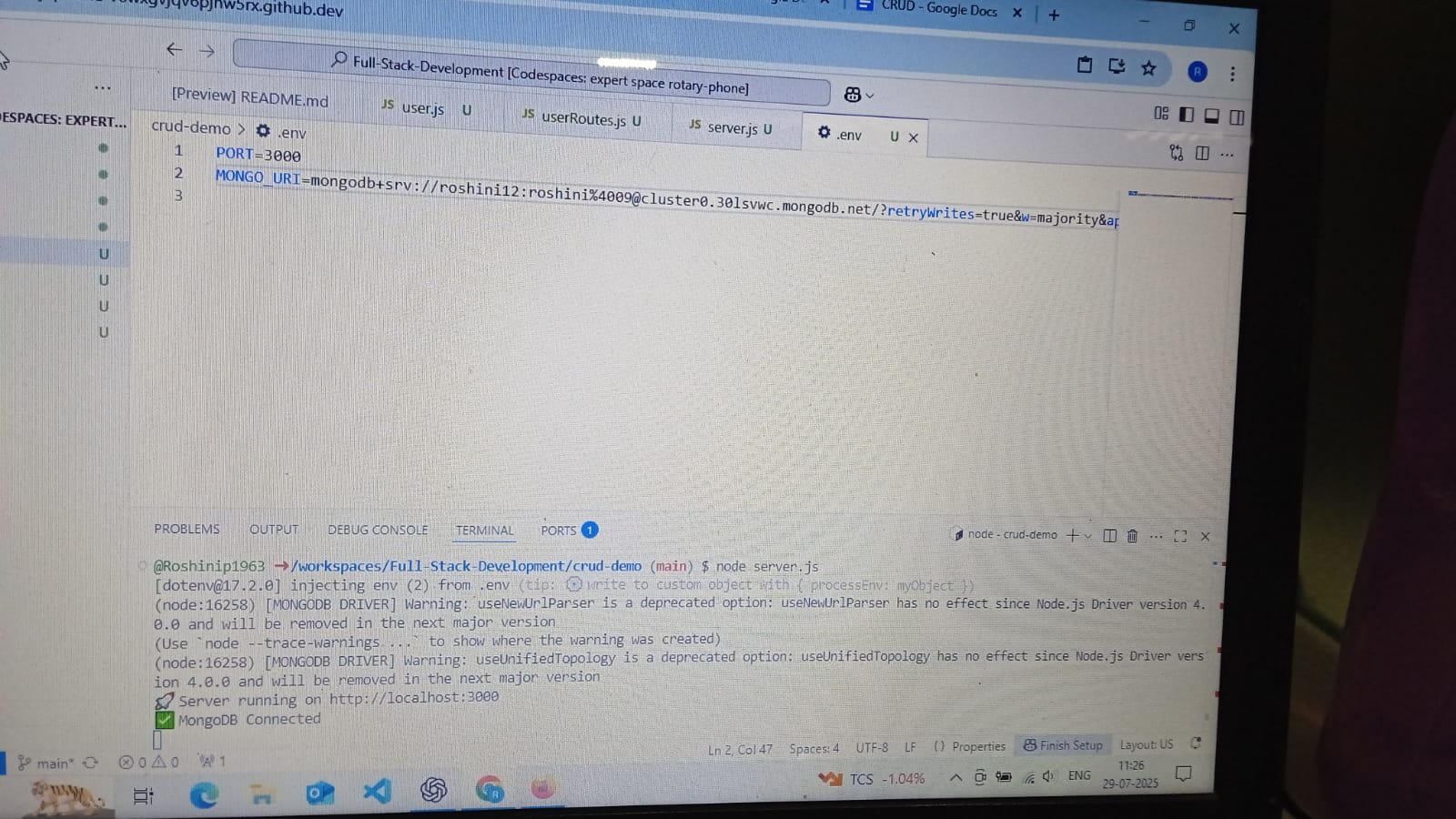
PORT=3000

MONGO\_URI=mongodb+srv://roshini12:roshini%4009@cluster0.30lsvwc.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0

1. **Start Your Server**

node server.js

**Output:**

****

**Step 1: Log in to MongoDB Atlas**

**Go to:** [**https://cloud.mongodb.com**](https://cloud.mongodb.com)

**Log in**

**Click on your Project (e.g., "Project 0")**

**In the left sidebar, under SECURITY, click:**

**Click on Database Access**

**Username: roshini12**

**Password: roshini$40009**

**Set Database Access Permissions:**

**Built-in Role → Read and write to any database**

**Click Add User**

**Network Access (Allow All IPs):**

**In the left sidebar, go to: Network Access**

1. **Click the green "Add IP Address" button**
2. **In the popup:**

**Click "Allow Access from Anywhere"**

**This auto-fills: 0.0.0.0/0**

**Click Confirm**

**Go to Github**

**Start Your Server**

node server.js