💡 Activity: Set Up MongoDB Atlas and Connect via Mongoose

🧭 Objective:

Students will create a free MongoDB cluster using MongoDB Atlas, configure database access and network access, and test a connection from a Node.js project using Mongoose.

🔧 Part 1: Create a MongoDB Atlas Cluster

1. Go to https://www.mongodb.com/cloud/atlas.
2. Sign up with a GitHub or Google account, or create a new MongoDB account.
3. Click **"Build a Database"**.
4. Select the **Shared Cluster (Free)** option.
5. Choose:
   - Cloud Provider: **AWS** or **GCP**
   - Region: Select a nearby region (e.g., US West or US East)
6. Click **Create Cluster** (this takes 2–5 minutes to initialize).

🔐 Part 2: Configure Database Access

1. Go to **Database Access** from the left sidebar.
2. Click **"Add New Database User"**.
3. Choose:
   - Authentication Method: **Password**
   - Username: studentuser
   - Password: yourSecurePassword123
   - Privileges: **Read and write to any database**
4. Click **Add User**.

📌 **Tip**: Students should copy their username and password somewhere secure. They'll use it in their .env file later.

🌐 Part 3: Configure Network Access

1. Click **Network Access** from the left sidebar.
2. Click **"Add IP Address"**.
3. Click **"Allow Access from Anywhere"** (adds 0.0.0.0/0) — for classroom testing only.
4. Click **Confirm**.

🔒 Later, students can limit access to their specific IP address.

🔧 Part 4: Connect Your Application

1. Go back to **Database > Connect**.
2. Choose **"Connect your application"**.
3. Select:
   - o Driver: **Node.js**
   - o Version: **5.0 or later**
4. Copy the connection string (example):

mongodb+srv://studentuser:<password>@cluster0.mongodb.net/myFirstDatabase?retryWrites=true&w=majority

📄 Part 5: Use in Node.js Project

1. Create a .env file in your Node.js folder:

MONGO_URL=mongodb+srv://studentuser:yourSecurePassword123@cluster0.mongodb.net/myFirstDatabase?retryWrites=true&w=majority

Add dotenv and mongoose to your project:

npm install mongoose dotenv

Use the Mongoose setup code from the slide (LISTING 14.23) to connect.

✅ Wrap-up

Once students are connected, have them:

- Insert a simple document using Model.create()
- Query the database with Model.find()
- Log results to console

📽 Project Structure:

📁 mongo-wrap-up/

├── .env

├── app.js

└── package.json

📄 .env file:

Replace the URI with your actual cluster URI:

MONGO_URL=mongodb+srv://studentuser:yourSecurePassword123@cluster0.mongodb.net/myFirstDatabase?retryWrites=true&w=majority

📜 app.js file:

```javascript
require('dotenv').config();

const mongoose = require('mongoose');


// 1. Connect to MongoDB

mongoose.connect(process.env.MONGO_URL, {

  useNewUrlParser: true,

  useUnifiedTopology: true

});


const db = mongoose.connection;

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

db.once('open', async () => {

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


  // 2. Define a Schema

  const studentSchema = new mongoose.Schema({

    name: String,

    age: Number,

    enrolled: Boolean,

    favoriteLanguages: [String]
```

```javascript
});

// 3. Create a Model
const Student = mongoose.model('Student', studentSchema);

// 4. Insert a Document
await Student.create({
  name: 'Alex',
  age: 22,
  enrolled: true,
  favoriteLanguages: ['JavaScript', 'Python']
});

console.log('📥 Inserted one student record.');

// 5. Query and Display Results
const students = await Student.find({ enrolled: true });
console.log('🎓 Enrolled Students:', students);

// 6. Close the connection
mongoose.connection.close();
});
```

✅ Expected Console Output:

✅ Connected to MongoDB

📥 Inserted one student record.

🎓 Enrolled Students: [{ _id: ..., name: 'Alex', age: 22, ... }]

Here's great **YouTube tutorials** that closely matches the activity we just built (Node.js + MongoDB Atlas using Mongoose):

👉 https://www.youtube.com/watch?v=fgTGADljAeg

👉 https://www.youtube.com/watch?utm_source=chatgpt.com