

Day 18 – [14th July 2025]

TOPICS COVERED

MongoDB Atlas and MongoDB Compass:

MongoDB Atlas is a cloud-hosted version of MongoDB. It allows us to:

- ❖ Host and manage databases online
- ❖ Access them from any app or location
- ❖ Connect via connection strings in our backend

Steps I followed:

- ❖ Signed up on MongoDB Atlas
- ❖ Created a Cluster and Database
- ❖ Created a collection (like a table)
- ❖ Whitelisted my IP and generated a connection string
- ❖ Connected using Compass to visually inspect data

MongoDB Compass is the GUI for MongoDB:

- ❖ Allows viewing and editing of database records easily
- ❖ Helpful to test if data is inserted or read properly

Nodemon:

- ❖ Nodemon is a development tool for Node.js applications:
- ❖ It auto-restarts the server whenever file changes are detected.
- ❖ Saves time while testing backend changes.

Installation:

```
npm install -g nodemon
```

Usage:

```
npx nodemon index.js
```

I also added it in package.json as a script:

```
"scripts": {  
  "start": "node index.js",
```

```
"dev": "nodemon index.js"
}
```

Now I just run:

```
npm run dev
```

HTTP Methods:

These methods define how the client (frontend) interacts with the server (backend):

Method	Use
GET	Read or fetch data
POST	Send or create new data
PUT / PATCH	Update existing data
DELETE	Remove data

I created a basic Express app and used these methods to create dummy routes:

```
app.get('/api/users', (req, res) => {
  res.send('All users');
});
app.post('/api/users', (req, res) => {
  res.send('User created');
});
app.put('/api/users/:id', (req, res) => {
  res.send('User updated');
});
app.delete('/api/users/:id', (req, res) => {
  res.send('User deleted');
});
```

TOOLS USED:

Visual Studio Code (Writing and managing backend code)

MongoDB Atlas (Hosting cloud-based NoSQL database)

MongoDB Compass (GUI to visualize and test database data)

Postman / Hoppscotch (For testing API requests like GET/POST)

Node.js + Express (Server-side scripting and route handling)

Nodemon (Auto-refresh backend server on changes)

TASK:

Read about jwt, secret key and clouinary.