✅ **Complete Node.js + MongoDB Backend Training**
🕐 **Duration:** 6 to 8 Weeks

📚 **Detailed Syllabus**

📌 **Week 1: JavaScript & Node.js Basics**

- JavaScript ES6+ features (let/const, arrow functions, promises, async/await)

- What is Node.js?

- Setting up Node.js environment

- Node.js core modules (fs, http, path, etc.)

- npm & package.json

- Introduction to CommonJS (require) vs ES Modules (import)
  📦 **Output:** Build a basic CLI app using Node.js (e.g., To-do list or File Manager)

📌 **Week 2: Asynchronous Programming & HTTP**

- Callbacks vs Promises vs Async/Await

- Event Loop & Event Emitters

- Handling HTTP with core Node (http.createServer)

- Introduction to REST APIs

📌 **Week 3: Express.js Framework**

- Setting up Express

- Routes, Controllers, Middleware

- CRUD APIs (Create, Read, Update, Delete)

- RESTful principles

- Error handling middleware
  🛠️ **Mini Project:** Build a small blog API

📌 **Week 4: MongoDB & Mongoose**

- Introduction to NoSQL & MongoDB

- Prisma ORM (PostgresSQL as an alternative to Mongoose)

- Installing MongoDB locally or using MongoDB Atlas

- Mongoose ODM basics: Models, Schemas, Validations

- CRUD operations with Mongoose

- Relationships & Population

- **MongoDB $ operations** ($match, $group, $project, $lookup, $sort, etc.)

- **User-product aggregation using $lookup**
  🛠️ **Mini Project:** Connect blog API to MongoDB and persist data

---

📌 **Week 5: Authentication & Authorization**
- Hashing passwords with bcrypt

- JWT token-based authentication

- Middleware for route protection

- Role-based access control

- Login/Signup APIs

📌 **Week 6: Advanced Features**
- Email sending (nodemailer or AWS SES)

- File upload (multer or S3 presigned URLs)

- Environment variables with dotenv

- **Pagination list with search and date filters**

- Error handling and response standardization

📌 **Week 7: Final Project (Capstone)**

A full-featured backend system that includes:

- Auth system (register/login/roles)

- CRUD APIs for resources (e.g., products, users)

- MongoDB integration

- JWT-based protection

- File uploads

- Email verification or password reset

- **Join user and products using MongoDB $lookup aggregation**

- **Implement pagination with search, filters (e.g., by date or status)**

- **Advanced MongoDB queries using $ operators**

- **Bonus:** Swagger API docs

📌 **Week 8: Deployment & CI/CD**

- Hosting with Railway, Render, or Vercel

- MongoDB Atlas setup

- CI/CD basics using GitHub Actions

- Using Postman for testing

- Best practices for production

🧠 **Bonus Topics (Optional for Advanced Learners)**

- Socket.io (Real-time apps)

- Caching with Redis

- Rate limiting (express-rate-limit)

- Docker basics

🧾 **Deliverables per Trainee**

- 2 mini projects
- 1 capstone project
- GitHub repo with commits
- Postman collection for testing APIs
- Weekly code review & feedback