

## Task 3: Create a REST API to Manage a List of Books Using Node.js and Express.

- **Objective:** Build simple REST API endpoints for CRUD operations on books (no database needed, store in memory).
- **Tools:** Node.js (free), VS Code, Postman (free).
- **Deliverables:** A Node.js Express server with endpoints: GET, POST, PUT, DELETE

### Hints/Mini Guide:

1. Initialize project with npm init.
2. Install Express (npm install express).
3. Setup basic Express server on port 3000.
4. Create an array to store book objects {id, title, author}.
5. Implement GET /books to return all books.
6. POST /books to add a new book from request body.
7. PUT /books/:id to update a book by ID.
8. DELETE /books/:id to remove a book.
9. Test endpoints with Postman.

**Outcome:** : Understand REST API basics, Express routing, HTTP methods, and JSON handling.

### Interview Questions:

1. What is REST?
2. What are HTTP methods and their use?
3. How do you handle routes in Express?
4. What is middleware in Express?
5. How do you parse JSON in Express?
6. What status codes do you use for CRUD?
7. How would you handle errors in Express?
8. What is CORS?
9. Explain request and response objects in Express.
10. How do you test API endpoints?

**Key Concepts:** REST API, Express.js, HTTP methods, JSON, Middleware, CRUD.

### Submit Here:

After completing the task, paste your GitHub repo link and submit it using the link below:

-  [\[Submission Link\]](#).

## 📌 Task Submission Guidelines

- 🕒 **Time Window:**

You can complete the task anytime between 10:00 AM to 10:00 PM on the given day. Submission link closes at 10 :00 PM

- 🔍 **Self-Research Allowed:**

You are free to explore, Google, or refer to tutorials to understand concepts and complete the task effectively.

- 🛠️ **Debug Yourself:**

Try to resolve all errors by yourself. This helps you learn problem-solving and ensures you don't face the same issues in future tasks.

- 💰 **No Paid Tools:**

If the task involves any paid software/tools, do not purchase anything. Just learn the process or find free alternatives.

- 📁 **GitHub Submission:**

Create a new GitHub repository for each task.

Add everything you used for the task — code, datasets, screenshots (if any), and a **short README.md** explaining what you did.

- 📁 **Submit Here:**

After completing the task, paste your GitHub repo link and submit it using the link below:

- 🖱️ [[Submission Link](#)].

Best  
of  
Luck

