Mini-Project (All Groups): Single-Table CRUD App Using Node.Js

For your assigned group projecst, build a single-table database, a Node.js/Express REST API that exposes GET, POST, PUT, DELETE, and a simple HTML/JavaScript page that uses the API to perform CRUD from the browser. You must also verify all endpoints in Postman.

Part A — Database (1 table only)
Create one table for your theme (see Group list below).
Include:
Id (integer primary key, auto-increment)
At least 5 additional fields relevant to your theme (choose appropriate data types).
Timestamps (created_at, updated_at) are recommended.
Populate at least 10 sample records (can be inserted via SQL using INSERT)

Part B — API (Node.js + Express)
Implement a REST API with these endpoints (replace <resource> with your group's resource name below):
GET /api/<resource> → List all records
GET /api/<resource>/:id → Get one record
POST /api/<resource> → Create a record
PUT /api/<resource>/:id → Update a record
DELETE /api/<resource>/:id → Delete a record

Requirements
Return JSON for all responses.
Use environment variables for DB config (e.g., .env).
Enable CORS so your HTML page can call the API.

Postman Evidence
Show successful requests for all five endpoints (a Video) including URL, method, request body (for POST/PUT/GET/DELETE), and JSON responses.

Part C — Frontend (HTML + JavaScript)
Build a minimal single-page interface that:

- Lets the user create a new record (a form with relevant inputs).
- Lists existing records in a table.
- Allows edit (loads values into the form and saves via PUT) and delete operations.
- Updates the UI after each operation

1. **Project Structure**

node-mysql-crud/
├── backend/
│ ├── *.env*
│ ├── *package.json*
│ ├── *schema.sql*
│ ├── *db.js*
│ └── *server.js*
└── frontend/
├── *index.html*
└── *app.js*

2. Backend — Quick Start

1.1 Install deps

Open powershell then navigate to your project

```
cd backend
npm init -y
npm i express mysql2 cors dotenv
```

1.2 `.env`

```
PORT=3000
DB_HOST=127.0.0.1
DB_USER=root
DB_PASSWORD=
DB_NAME=campus_crud
```

For Database Tables and Resource Names

Group 1 — Campus Club Profile (resource: clubs, table: clubs)

Fields & Types
id INT (Primary Key, Auto_Increment)
name VARCHAR(120) NOT NULL
created_at TIMESTAMP (default CURRENT_TIMESTAMP)
updated_at TIMESTAMP (default CURRENT_TIMESTAMP on update)

Group 2 — Personal Recipe Card (resource: recipes, table: recipes)

Fields & Types
id INT (Primary Key, Auto_Increment)
title VARCHAR(150) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 3 — Student Timetable & Goals (resource: timetable, table: timetable)

Fields & Types
id INT (Primary Key, Auto_Increment)
course_code VARCHAR(20) NOT NULL
course_name VARCHAR(150)
created_at TIMESTAMP

updated_at TIMESTAMP

Group 4 — Museum Exhibit Poster (resource: exhibits, table: exhibits)
Fields & Types
id INT (Primary Key, Auto_Increment)
exhibit_title VARCHAR(150) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 5 — Tech Product Showcase (resource: products, table: products)
Fields & Types
id INT (Primary Key, Auto_Increment)
product_name VARCHAR(150) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 6 — Travel Destination Guide (resource: destinations, table: destinations)
Fields & Types
id INT (Primary Key, Auto_Increment)
destination VARCHAR(150) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 7 — Music Festival Line-Up (resource: lineup, table: lineup)
Fields & Types
id INT (Primary Key, Auto_Increment)
festival_name VARCHAR(150) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 8 — Book Club Meeting Page (resource: meetings, table: meetings)
Fields & Types
id INT (Primary Key, Auto_Increment)
book_title VARCHAR(200) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 9 — Sports Team Roster & Fixtures (resource: sports, table: sports)
Fields & Types
id INT (Primary Key, Auto_Increment)
player_name VARCHAR(120) *(NULL for fixtures)*
position VARCHAR(60) *(NULL for fixtures)*
jersey_number INT *(NULL for fixtures)*
created_at TIMESTAMP
updated_at TIMESTAMP

Group 10 — NGO Cause Awareness (resource: campaigns, table: campaigns)
Fields & Types
id INT (Primary Key, Auto_Increment)

campaign_name VARCHAR(150) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 11 — Portfolio Landing (resource: portfolio, table: portfolio)
Fields & Types
id INT (Primary Key, Auto_Increment)
owner_name VARCHAR(120) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 12 — School Event Noticeboard (resource: notices, table: notices)
Fields & Types
id INT (Primary Key, Auto_Increment)
event_title VARCHAR(150) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP

Group 13 — Science Fair Projects (resource: projects, table: projects)
Fields & Types
id INT (Primary Key, Auto_Increment)
project_title VARCHAR(200) NOT NULL
student_name VARCHAR(150)
created_at TIMESTAMP
updated_at TIMESTAMP

Group 14 — Health Clinic Info Page (resource: clinic, table: clinic)
Fields & Types
id INT (Primary Key, Auto_Increment)
clinic_name VARCHAR(150) NOT NULL
doctor_name VARCHAR(150)
created_at TIMESTAMP
updated_at TIMESTAMP

Group 15 — Cinema Showtimes (resource: showtimes, table: showtimes)
Fields & Types
id INT (Primary Key, Auto_Increment)
movie_title VARCHAR(200) NOT NULL
created_at TIMESTAMP
updated_at TIMESTAMP