**CODE :**

**Index.js**

require('dotenv').config();

const app = require('./src/app');

const db = require('./src/models');

const PORT = process.env.PORT || 5000;

db.sequelize.authenticate()

.then(() => {

console.log("✅ Database connected!");

app.listen(PORT, () => console.log(`🚀 Server running on http://localhost:${PORT}`));

})

.catch(err => console.error("❌ Database connection error:", err));

**Db.config.js**

require('dotenv').config();

module.exports = {

HOST: process.env.DB\_HOST,

USER: process.env.DB\_USER,

PASSWORD: process.env.DB\_PASSWORD,

DB: process.env.DB\_NAME,

DIALECT: process.env.DB\_DIALECT

};

**Db.helper.js**

const { Sequelize } = require('sequelize');

const dbConfig = require('./db.config');

const sequelize = new Sequelize(dbConfig.DB, dbConfig.USER, dbConfig.PASSWORD, {

host: dbConfig.HOST,

dialect: dbConfig.DIALECT

});

sequelize.authenticate()

.then(() => console.log("✅ Database connected successfully!"))

.catch(err => console.error("❌ Database connection failed:", err));

module.exports = sequelize;

**Student.controller.js**

const studentService = require('../services/student.service');

exports.createStudent = async (req, res) => {

try {

const student = await studentService.createStudent(req.body);

res.status(201).json(student);

} catch (error) {

res.status(400).json({ error: error.message });

}

};

exports.getAllStudents = async (req, res) => {

try {

const students = await studentService.getAllStudents();

res.status(200).json(students);

} catch (error) {

res.status(500).json({ error: error.message });

}

};

exports.getStudentById = async (req, res) => {

try {

const student = await studentService.getStudentById(req.params.id);

student ? res.status(200).json(student) : res.status(404).json({ message: "Student not found" });

} catch (error) {

res.status(500).json({ error: error.message });

}

};

exports.updateStudent = async (req, res) => {

try {

const result = await studentService.updateStudent(req.params.id, req.body);

result[0] ? res.status(200).json({ message: "Student updated successfully" }) : res.status(404).json({ message: "Student not found" });

} catch (error) {

res.status(500).json({ error: error.message });

}

};

exports.deleteStudent = async (req, res) => {

try {

const result = await studentService.deleteStudent(req.params.id);

result ? res.status(200).json({ message: "Student deleted successfully" }) : res.status(404).json({ message: "Student not found" });

} catch (error) {

res.status(500).json({ error: error.message });

}

};

**Model.Index.js**

const sequelize = require('../config/db.helper');

const Student = require('./student.model');

const db = { sequelize, Student };

sequelize.sync()

.then(() => console.log("✅ Tables synchronized!"))

.catch(err => console.error("❌ Sync error:", err));

module.exports = db;

**Student.model.js**

const { DataTypes } = require('sequelize');

const sequelize = require('../config/db.helper');

const Student = sequelize.define('students', {

id: { type: DataTypes.INTEGER, autoIncrement: true, primaryKey: true },

name: { type: DataTypes.STRING, allowNull: false },

email: { type: DataTypes.STRING, allowNull: false, unique: true },

age: { type: DataTypes.INTEGER }

}, { timestamps: false });

module.exports = Student;

**Student.routes.js**

const express = require('express');

const studentController = require('../controllers/student.controller');

const router = express.Router();

router.post('/', studentController.createStudent);

router.get('/', studentController.getAllStudents);

router.get('/:id', studentController.getStudentById);

router.put('/:id', studentController.updateStudent);

router.delete('/:id', studentController.deleteStudent);

module.exports = router;

**Student.services.js**

const { Student } = require('../models');

const createStudent = async (data) => await Student.create(data);

const getAllStudents = async () => await Student.findAll();

const getStudentById = async (id) => await Student.findByPk(id);

const updateStudent = async (id, data) => await Student.update(data, { where: { id } });

const deleteStudent = async (id) => await Student.destroy({ where: { id } });

module.exports = { createStudent, getAllStudents, getStudentById, updateStudent, deleteStudent };

**App.js**

const express = require('express');

const bodyParser = require('body-parser');

const cors = require('cors');

const studentRoutes = require('./routes/student.routes');

const app = express();

app.use(cors());

app.use(bodyParser.json());

app.use('/students', studentRoutes);

app.get('/', (req, res) => res.send('🎉 Student Management API is running!'));

module.exports = app;

**.env**

DB\_HOST=localhost

DB\_USER=root

DB\_PASSWORD=

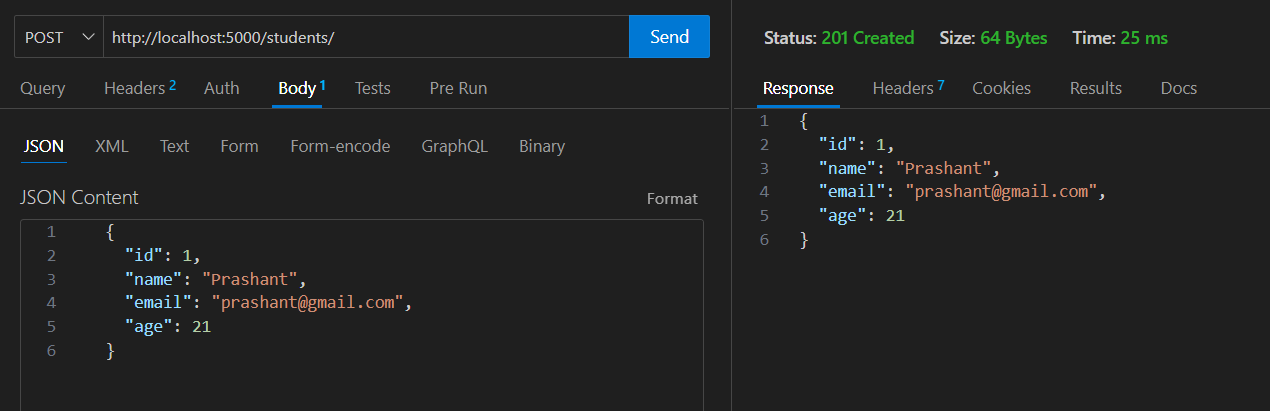
DB\_NAME=mu\_2425\_test

DB\_DIALECT=mysql

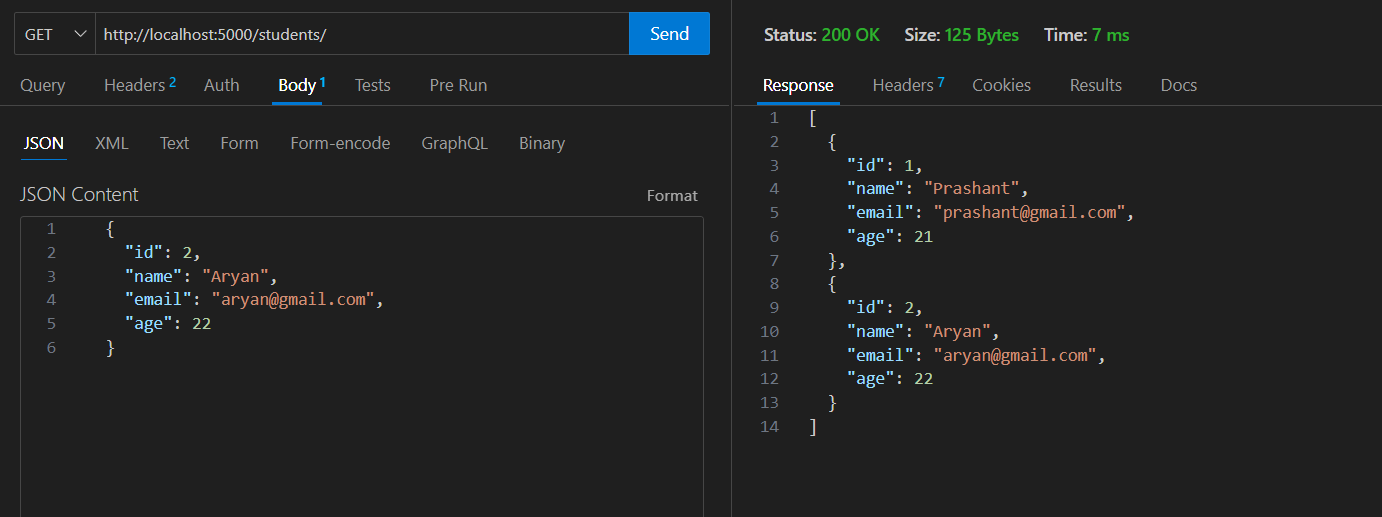
PORT=5000

**OUTPUT :**

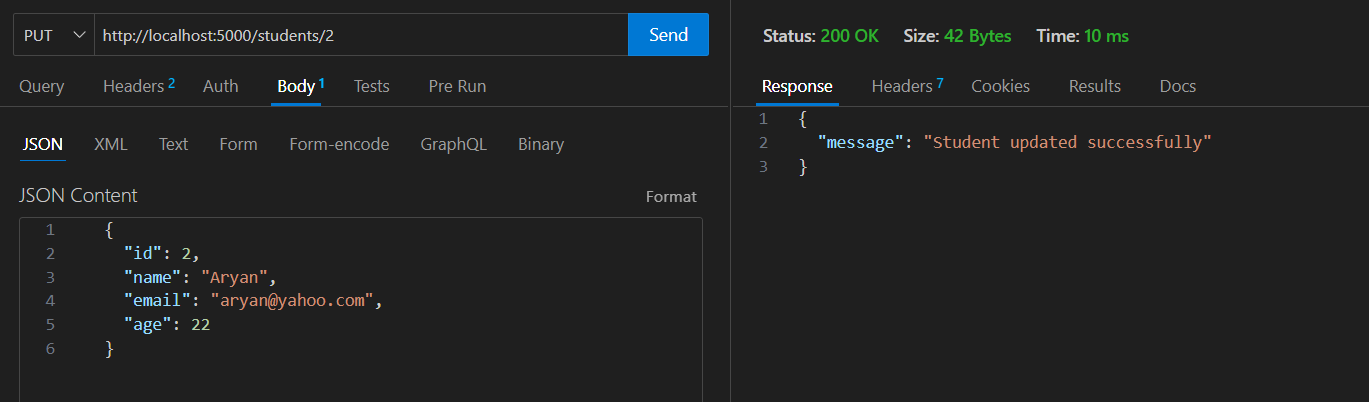
**Insert the student**



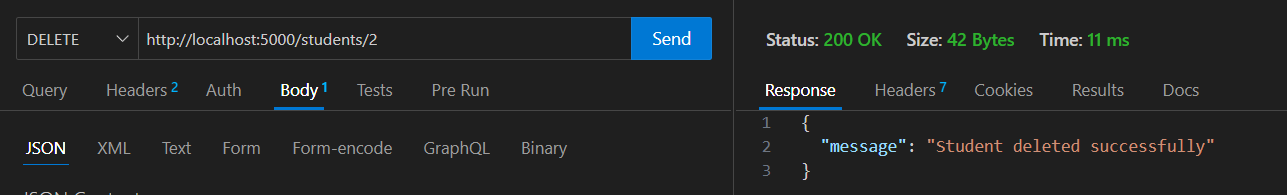
**Get all the student**



**Edit the student**



**Delete the student**



**DATABASE**

