**Exercise:**

**Spring Boot REST CRUD – Student Management**

**Objective:**

Create a **Spring Boot REST API** that performs CRUD (Create, Read, Update, Delete) operations on a list of Student objects using **in-memory ArrayList storage**, without using a database.

**Task Overview**

**1. Project Setup**

* Create a new Spring Boot project using Spring Initializr or your preferred method.
* Add the following dependency:
  + Spring Web
* Use the following package structure:

├── controller

├── model

├── service

└── StudentCrudApplication.java

**2. Create the Student Model**

Inside the model package:

* Create a Student class with the following fields:
  + int id
  + String name
  + int age
  + String email
  + String course
* Add:
  + Constructors
  + Getters and Setters
  + toString() method

**3. Create a StudentService Class**

Inside the service package:

* Use @Service annotation
* Maintain an in-memory list: List<Student> students = new ArrayList<>();
* Add methods for:
  + addStudent(Student student)
  + getAllStudents()
  + getStudentById(int id)
  + updateStudent(int id, Student student)
  + deleteStudent(int id)

**4. Create the StudentController Class**

Inside the controller package:

* Use @RestController
* Map endpoints with @RequestMapping("/students")

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| POST | /students | Add a new student |
| GET | /students | List all students |
| GET | /students/{id} | Get student by ID |
| PUT | /students/{id} | Update student by ID |
| DELETE | /students/{id} | Delete student by ID |

**5. Implement Response Handling**

* Return appropriate response objects.
* Handle error cases (e.g., student not found).
* Avoid exposing internal errors in API responses.

**6. Test the API**

* Use Postman or curl to test all endpoints:
  + Add a few students
  + Fetch all students
  + Fetch by ID
  + Update a student
  + Delete a student