University of Sri Jayewardenepura

Faculty of Technology

Department of Information Communication Technology

ITS 4243 - Microservices and Cloud Computing

Assignment 01

ICT/21/929

Part 1

1. What is Spring Boot and why is it used?

Spring Boot is a Java framework. It helps to build Spring applications easily. It gives many built-in tools, so we don't need to do much setup. It is used because it makes development faster and easier.

2. Explain the difference between Spring Framework and Spring Boot.

Spring Framework is big and needs a lot of setup. Spring Boot is a simpler way to use Spring. Spring Boot gives automatic setup and an embedded server. So we can run the project easily.

3. What is Inversion of Control (IoC) and Dependency Injection (DI)?

IoC means the control of objects is given to the Spring container. We don't create objects ourselves.

DI means Spring gives the needed objects to a class automatically. It helps to connect parts of the program easily.

4. What is the purpose of application.properties / application.yml?

These files are used to keep the project's settings. For example, we can add database details, server port, and other options. Spring Boot reads these files when the app starts.

5. Explain what a REST API is and list HTTP methods used.

REST API helps two systems to talk using the internet. It uses HTTP methods like:

GET – to get data

POST – to add data

PUT – to update data

DELETE – to remove data

6. What is Spring Data JPA? What is an Entity and a Repository?

Spring Data JPA helps to work with databases easily.

An Entity is a class that maps to a table in the database.

A Repository is an interface that helps to save, find, update, and delete data.

7. What is the difference between @Component, @Service, @Repository, @Controller, @RestController?

@Component – for general classes

@Service – for business logic

@Repository – for database layer

@Controller – for web pages (HTML)

@RestController – for REST APIs (JSON data)

8. What is @Autowired? When should we avoid it?

@Autowired helps to automatically connect one class to another.

We should avoid it when we use too many of them, or when it makes the code hard to test or understand.

9. Explain how Exception Handling works in Spring Boot (@ControllerAdvice).

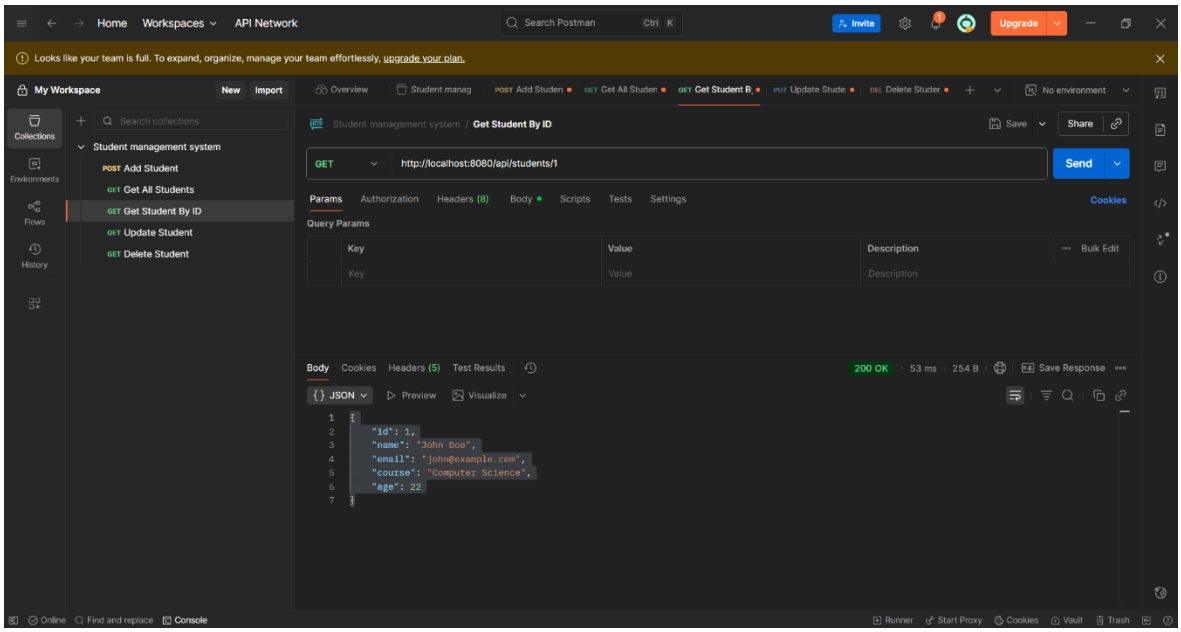
@ControllerAdvice helps to handle errors in one place. When an error happens in any controller, it goes to the advice class, and it gives a custom message or response.

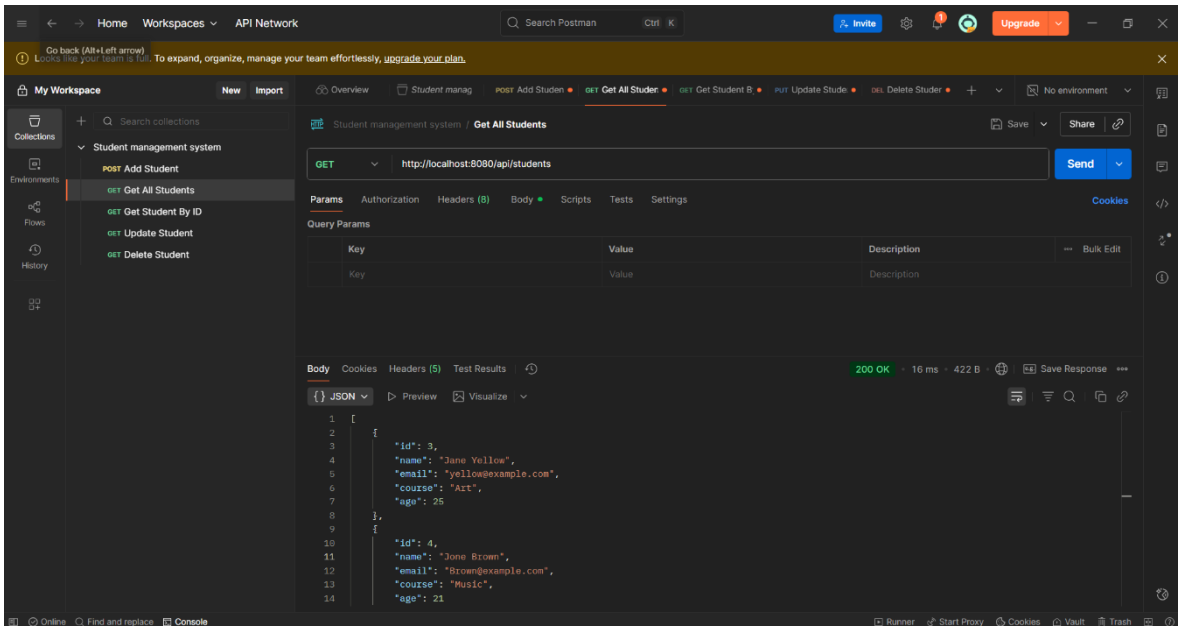
10. What is the role of Maven/Gradle in a Spring Boot project?

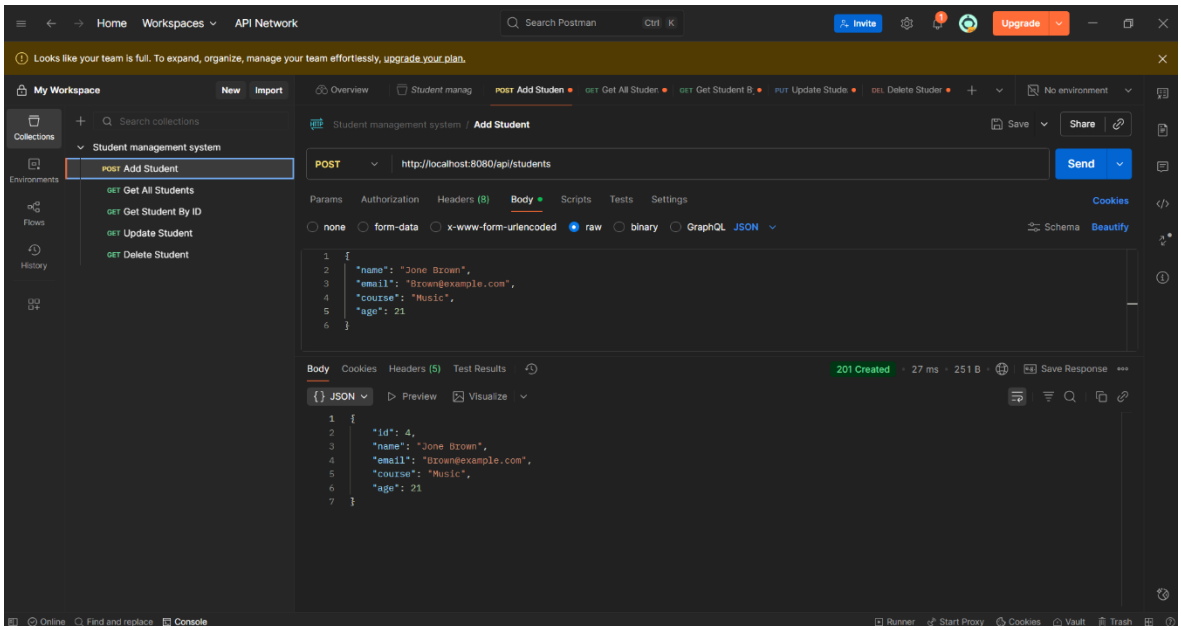
Maven and Gradle are build tools. They help to add libraries, build the project, and manage versions. They make it easy to run and package the Spring Boot app.

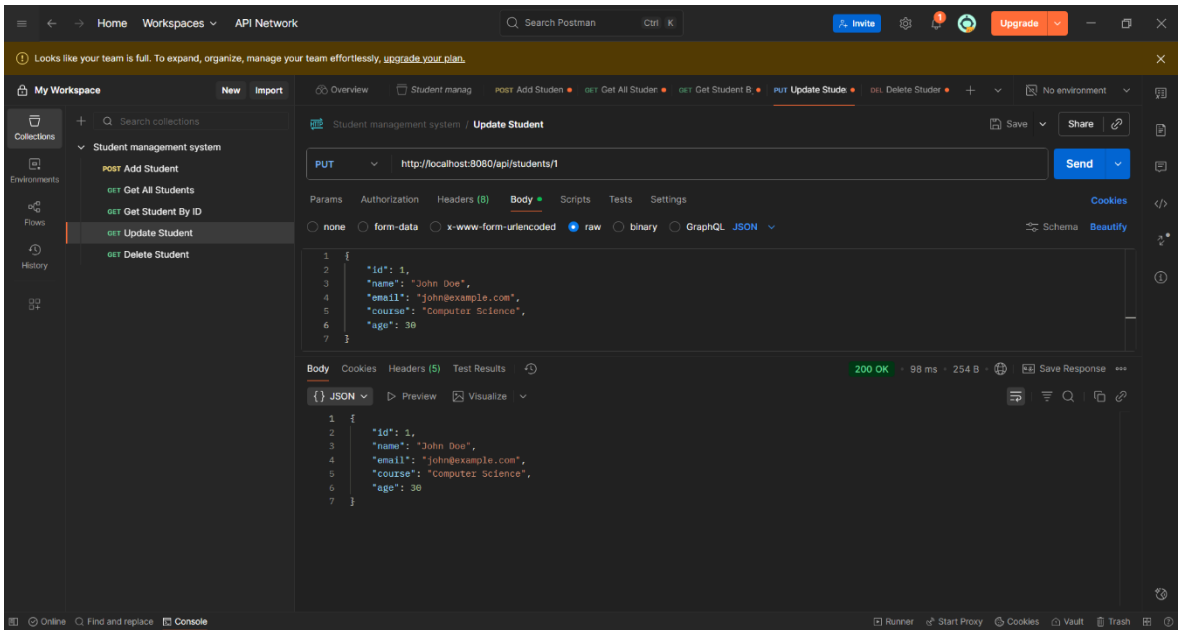
Part 2

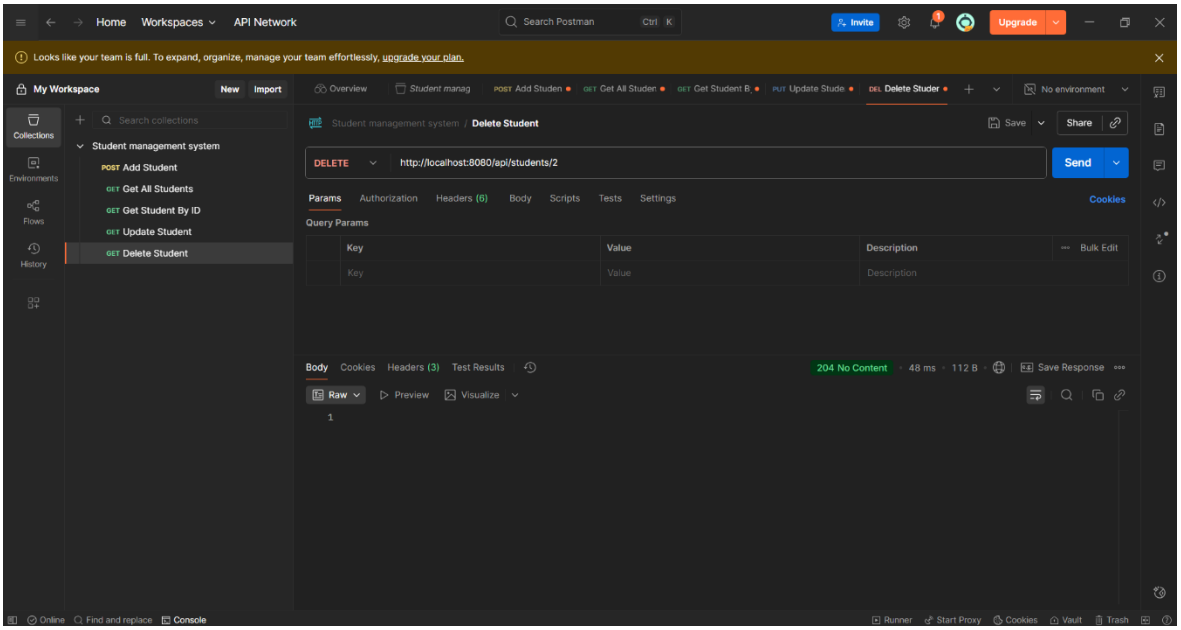
Screenshots






Home
Workspaces
API Network
Search Postman
Invite
Upgrade
Looks like your team is full. To expand, organize, manage your team effortlessly, upgrade your plan.
My Workspace
New
Import
Collections
Environments
Flows
History
Student management system
POST Add Student
GET Get All Students
GET Get Student By ID
GET Update Student
GET Delete Student
Student management system / Add Student
Save
Share
POST
http://localhost:8080/api/students
Send
Params
Authorization
Headers (8)
Body
Scripts
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Schema
Beautify
{
"name": "Jone Brown",
"email": "Brown@example.com",
"course": "Music",
"age": 21
}
Body
Cookies
Headers (5)
Test Results
201 Created
27 ms
251 B
Save Response
{ } JSON
Preview
Visualize
{
"id": 4,
"name": "Jone Brown",
"email": "Brown@example.com",
"course": "Music",
"age": 21
}
Online
Find and replace
Console
Runner
Start Proxy
Cookies
Vault
Trash


Home
Workspaces
API Network
Search Postman
Invite
Upgrade
Looks like your team is full. To expand, organize, manage your team effortlessly, upgrade your plan.
My Workspace
New
Import
Collections
Environments
Flows
History
Student management system
POST Add Student
GET Get All Students
GET Get Student By ID
GET Update Student
GET Delete Student
Student management system / Update Student
Save
Share
PUT
http://localhost:8080/api/students/1
Send
Params
Authorization
Headers (8)
Body
Scripts
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Schema
Beautify
{
"id": 1,
"name": "John Doe",
"email": "john@example.com",
"course": "Computer Science",
"age": 30
}
Body
Cookies
Headers (5)
Test Results
200 OK
98 ms
254 B
Save Response
{ } JSON
Preview
Visualize
{
"id": 1,
"name": "John Doe",
"email": "john@example.com",
"course": "Computer Science",
"age": 30
}
Online
Find and replace
Console
Runner
Start Proxy
Cookies
Vault
Trash



GitHub repository Link - https://github.com/NuwanSubasingha99/Student-management-system-Springboot