

## **Advanced Java Lab**

### **Week-9**

**Roll Number: 248W5A1213**

#### **1. Build a Simple REST API with GET and POST Endpoints**

##### **Code:**

##### **Student.java:**

```
package com.example.restapi;

public class Student {

    private int id;
    private String name;

    public Student() {}

    public Student(int id, String name) {
        this.id = id;
        this.name = name;
    }

    public int getId() { return id; }
    public void setId(int id) { this.id = id; }

    public String getName() { return name; }
    public void setName(String name) { this.name = name; }
}
```

##### **StudentController.java**

```
package com.example.restapi;

import org.springframework.web.bind.annotation.*;
import java.util.ArrayList;
import java.util.List;

@RestController
@RequestMapping("/students")
public class StudentController {

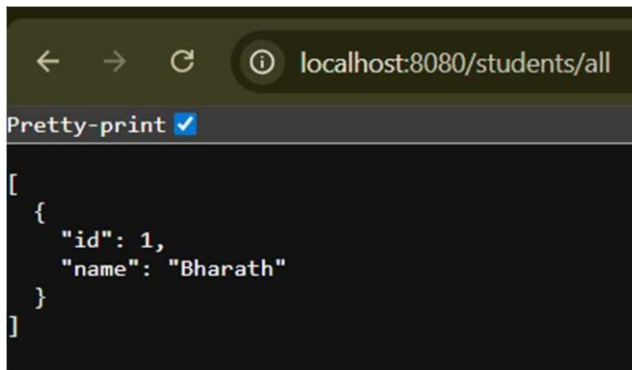
    List<Student> list = new ArrayList<>();

    @GetMapping("/all")
    public List<Student> getAllStudents() {
        return list;
    }

    @PostMapping("/add")
    public String addStudent(@RequestBody Student student) {
        list.add(student);
        return "Student Added Successfully!";
    }

    @GetMapping("/search")
    public Student searchById(@RequestParam int id) {
        return list.stream()
            .filter(s -> s.getId() == id)
            .findFirst()
            .orElse(null);
    }
}
```

**Output:**



## 2. Test the API using Postman and handle basic request parameters.

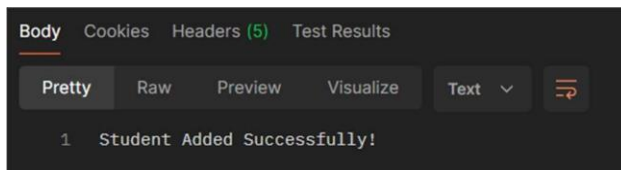
**Code:**

URL: <http://localhost:8080/students/add>

Body → raw → JSON:

```
{
  "id": 1,
  "name": "Bharath"
}
```

Response:



URL: <http://localhost:8080/students/search?id=1>

Response:

