Lab 5.1

Request Parameters

- In your existing project in the class StudentController, add two new methods as below:

```
public Student getStudent(long id) {
    return studentService.get(id);
}

public Student getSingleStudent(Optional<Long> optional) {
    return studentService.get(optional.orElse(1l));
}
```

- Annotate the method getStudent(long) with @GetMapping("/{id}") and the argument with @PathVariable("id")

- Annotate the method getSingleStudent(Optional) with @GetMapping("/single") and the argument with @RequestParam("id")

Lab 5.1

Request Parameters

Still in StudentController, alter the method getMessage() to accept a string argument for the header(“user-agent”) i.e. @RequestHeader("user-agent") String userAgent, and concatenate it onto the end of the methods returned String

Relaunch your Spring Boot Application and try out the urls:

- http://localhost:8081/college/student/msg
- http://localhost:8081/college/student/1
- http://localhost:8081/college/student/single?id=1

Lab 5.2

Request Parameters - Query

- We will add a custom query to our application using both @PathVariable and @RequestParam

- In the Interface com.student.service.StudentService add the abstract method;

```
Collection<Student> getAllStudentsInDepartment(String department, String lastNamelike);
```

- In the class that implements this Interface (com.student.service.StudentServiceImpl) implement the method as shown below, to filter all students by department name and that their lastName contains the lastNameLike argument

```
public Collection<Student> getAllStudentsInDepartment(String department, String lastNameLike) {
return studentDao.getAll()
    .stream()
    .filter(p-> p.getDept().equals(department))
    .filter(p-> p.getSurname().contains(lastNameLike))
    .collect(Collectors.toList());
}
```

Lab 5.2

Request Parameters - Query

In the class StudentController, add a new method as below;

```
public Collection<Student> getStudentsPerDepartment(String department, Optional<String>
optional) {
    return studentService.getAllStudentsInDepartment(department, optional.orElse(""));
}
```

- Annotate the new method with @GetMapping("/search/{department}")
- The first argument “department” is to be annotated with (@PathVariable("department")
- The second argument is to be annotated with @RequestParam("name")

Re launch your Spring Boot Application

Try out the new endpoint url at;
http://localhost:8081/college/student/search/English%20Literature?name=t