CEGEP VANIER COLLEGE
CENTRE FOR CONTINUING EDUCATION
Web Services
420-941-VA

Teacher: Samir Chebbine **Lab 6** **Nov 11, 2024**

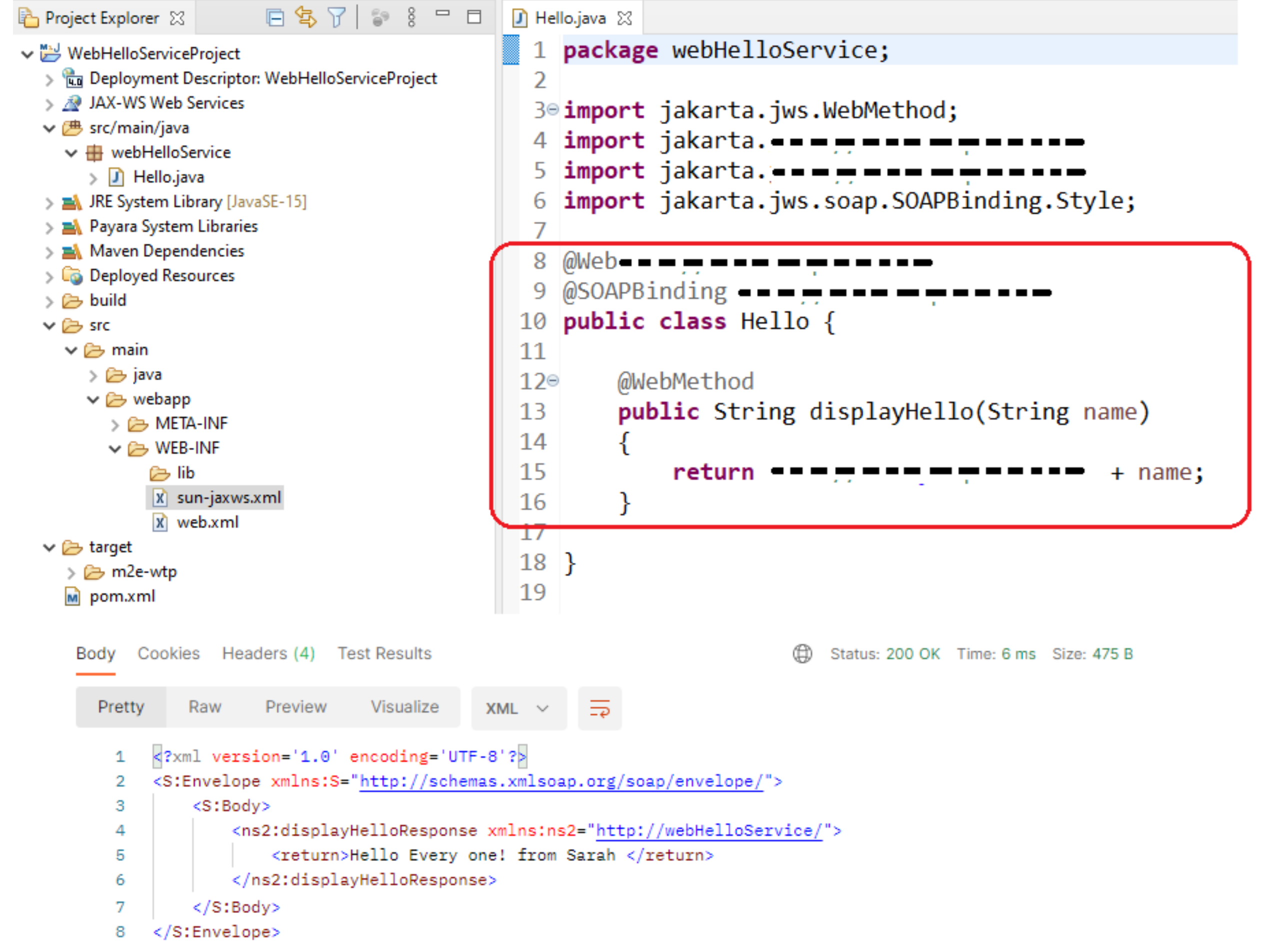
Lab 6: Glassfish SOAP Web Services & Spring Boot REST Resources

Complete all these following programs in class**.** All *missing coding statements* are presented during class time and in Presentation 6.

Create and Submit a Word file ***Lab6WebServicesYourName.doc*** which contains Answers of theory questions if any and output screenshots for every Java EE Project. Submit the Java projects too and submit the whole Lab 6 as compressed zip file

1. SOAP-Based Application

a) Create a new Dynamic Web project called **WebHelloServiceProject** and convert it into Maven Project.
b) Deploy **WebHelloServiceProject** within GalssFish Server.
c) Using Postman, display screenshots testing each SOAP request included in my YouTube Video Lab 6 as shown in Figure 1.



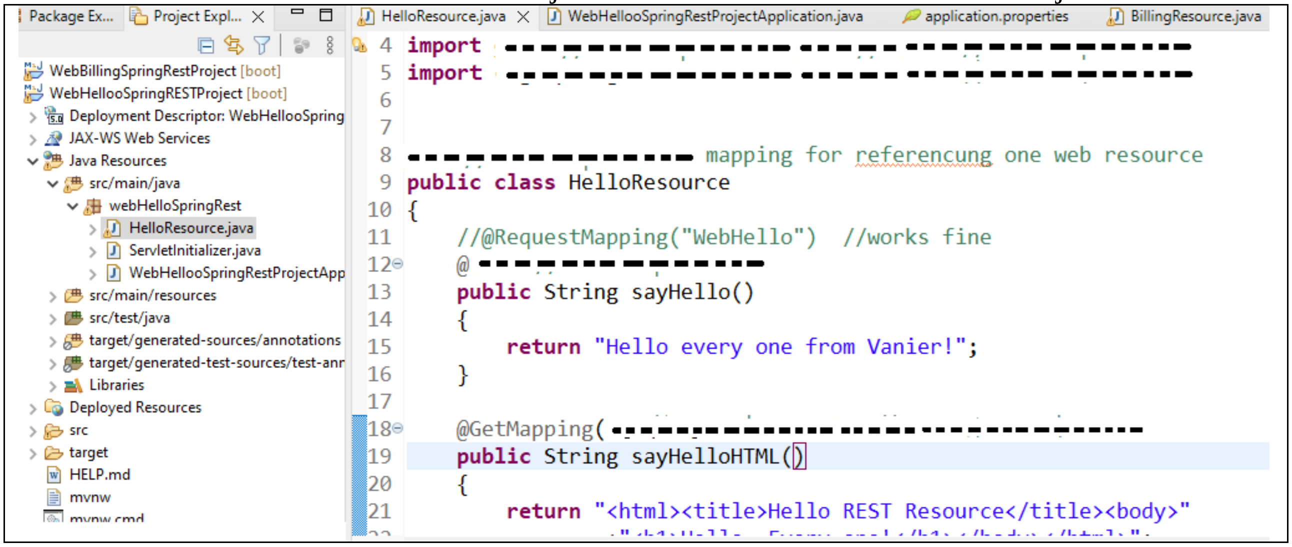
```java
package webHelloService;

import jakarta.jws.WebMethod;
import jakarta.[redacted]
import jakarta.[redacted]
import jakarta.jws.soap.SOAPBinding.Style;

@Web[redacted]
@SOAPBinding [redacted]
public class Hello {

    @WebMethod
    public String displayHello(String name)
    {
        return [redacted] + name;
    }

}
```

```xml
<?xml version='1.0' encoding='UTF-8'?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
    <S:Body>
        <ns2:displayHelloResponse xmlns:ns2="http://webHelloService/">
            <return>Hello Every one! from Sarah </return>
        </ns2:displayHelloResponse>
    </S:Body>
</S:Envelope>
```

Figure 1

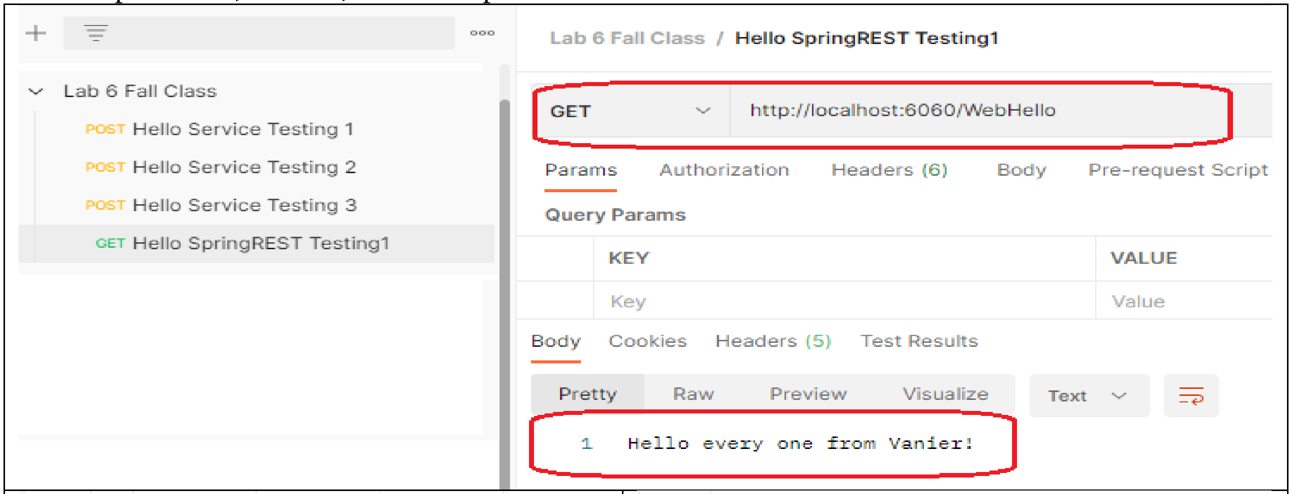
2. Creating REST Projects Using Spring Boot Framework

a) Complete all these following programs in class. All *missing coding statements* were presented during class time. Your WebHelloSpringRESTProject should reflect all options used in Lab 3 WebHelloRESTProject and Lab5 WebHelloFormRESTProject.



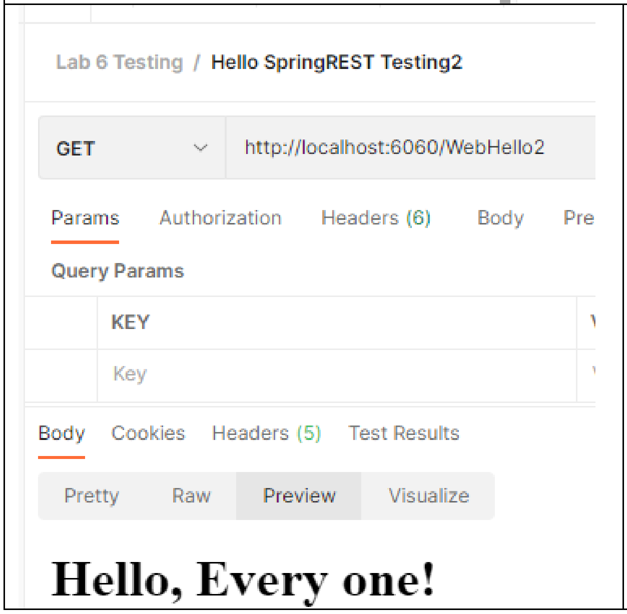
```
import ------------------------------------------
import ------------------------------------------


--------------- mapping for referencung one web resource
public class HelloResource
{
    //@RequestMapping("WebHello")  //works fine
    @----------------
    public String sayHello()
    {
        return "Hello every one from Vanier!";
    }

    @GetMapping(------------------------------------
    public String sayHelloHTML()
    {
        return "<html><title>Hello REST Resource</title><body>"
```

b) Using Postman, display screenshots testing to each media type included in Lab6 related to plain text, HTML, JSON output as shown hereafter.



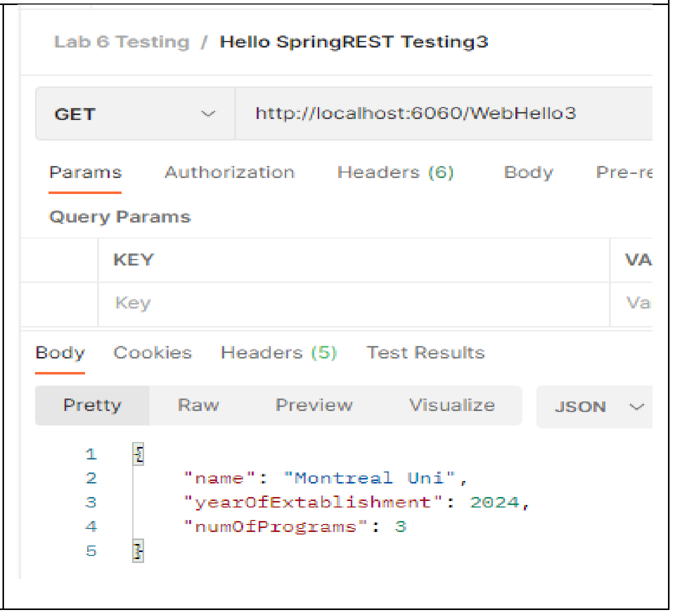
GET http://localhost:6060/WebHello

```
1   Hello every one from Vanier!
```



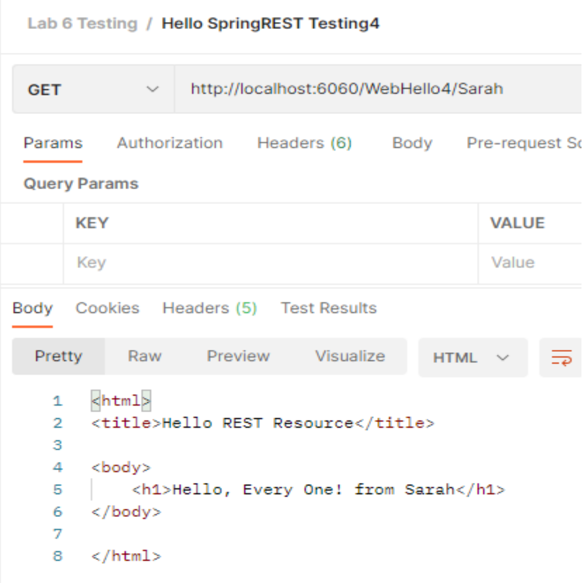
GET http://localhost:6060/WebHello2

Hello, Every one!



GET http://localhost:6060/WebHello3

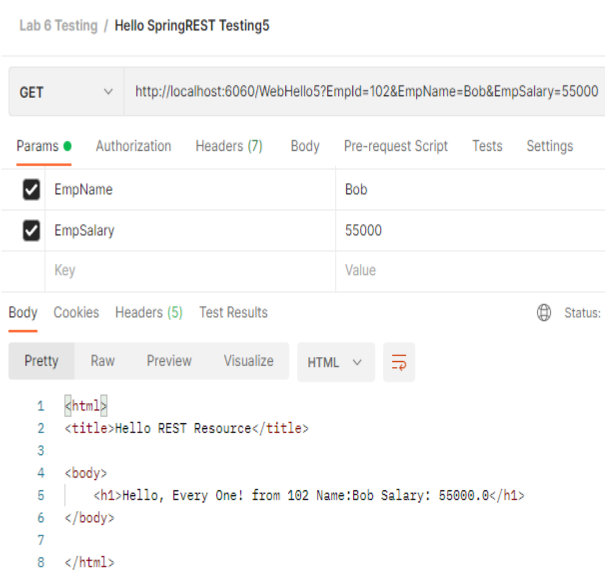
```
1  {
2      "name": "Montreal Uni",
3      "yearOfExtablishment": 2024,
4      "numOfPrograms": 3
5  }
```


Lab 6 Testing / Hello SpringREST Testing4

GET http://localhost:6060/WebHello4/Sarah

```
<html>
<title>Hello REST Resource</title>

<body>
    <h1>Hello, Every One! from Sarah</h1>
</body>

</html>
```

Path Parameter

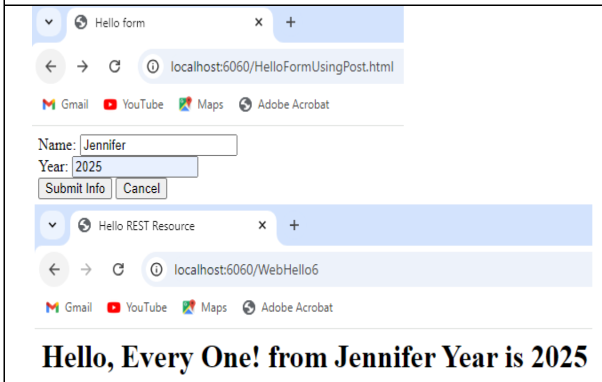

Lab 6 Testing / Hello SpringREST Testing5

GET http://localhost:6060/WebHello5?EmpId=102&EmpName=Bob&EmpSalary=55000

☑	EmpName	Bob
☑	EmpSalary	55000
	Key	Value

```
<html>
<title>Hello REST Resource</title>

<body>
    <h1>Hello, Every One! from 102 Name:Bob Salary: 55000.0</h1>
</body>

</html>
```

Query Parameters



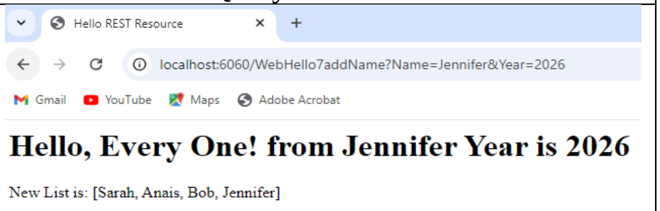
localhost:6060/HelloFormUsingPost.html

Name: Jennifer
Year: 2025
Submit Info Cancel

localhost:6060/WebHello6

Hello, Every One! from Jennifer Year is 2025

Form processing using POST method in HTTP request

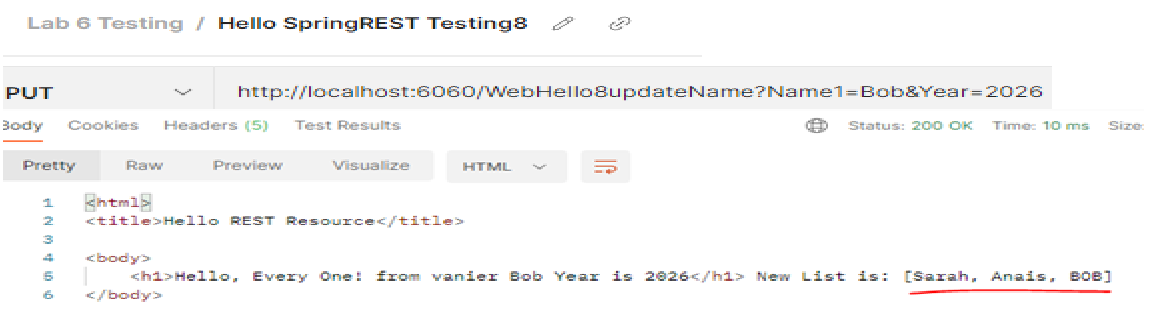


localhost:6060/WebHello7addName?Name=Jennifer&Year=2026

Hello, Every One! from Jennifer Year is 2026

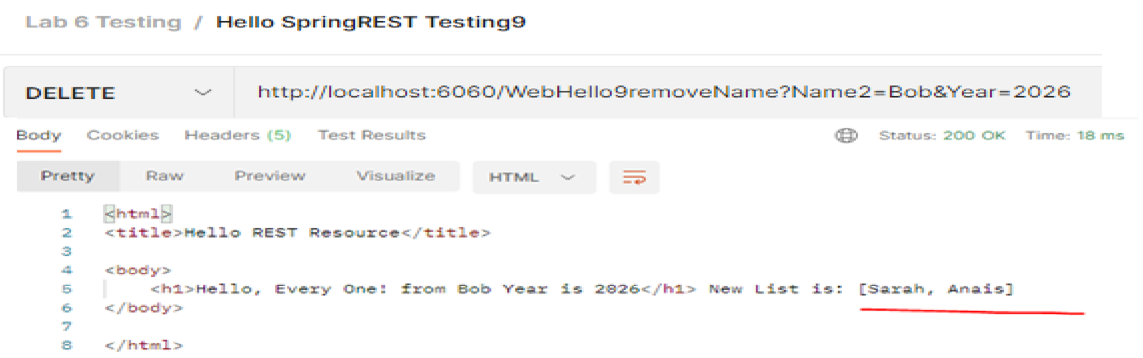
New List is: [Sarah, Anais, Bob, Jennifer]

Form processing using GET method in HTTP request to add new Name into ArrayList


Lab 6 Testing / Hello SpringREST Testing8

PUT http://localhost:6060/WebHello8updateName?Name1=Bob&Year=2026

```
<html>
<title>Hello REST Resource</title>

<body>
    <h1>Hello, Every One! from vanier Bob Year is 2026</h1> New List is: [Sarah, Anais, BOB]
</body>
```


Lab 6 Testing / Hello SpringREST Testing9

DELETE http://localhost:6060/WebHello9removeName?Name2=Bob&Year=2026

```
<html>
<title>Hello REST Resource</title>

<body>
    <h1>Hello, Every One! from Bob Year is 2026</h1> New List is: [Sarah, Anais]
</body>

</html>
```

Form processing using PUT & DELETE methods in HTTP request