WEB SERVICES

PRACTICAL JOURNAL

TCS2223013 CHIRAG BHATIA

SIES College of Arts, Science & Commerce
RISE WITH EDUCATION
NAAC REACCREDITED - 'A' GRADE
ISO 9001 : 2008

S.I.E.S College of Arts, Science and Commerce
Sion(W), Mumbai – 400 022.

CERTIFICATE

This is to certify that Mr. / ~~Miss.~~ CHIRAG KUMAR BHATIA

Roll No. TCS2223013 Has successfully completed the necessary course of experiments in the subject of **WEB SERVICES** during the academic year **2022 – 2023** complying with the requirements of **University of Mumbai**, for the course of **T.Y.BSc. Computer Science [Semester-5]**

Prof. In-Charge
Mrs. Maya Nair
(WEB SERVICES)

Examination Date:
Examiner's Signature & Date:

Head of the Department
Prof. Manoj Singh

College Seal
And
Date

	returns special news of that day, 2nd function called getPrediction accepts sunsign name as string parameter and returns predictions as string. Design a client to test the above web service.		
6	Design a Restful webservice from a database table Employee with columns empid,empname and Designation. Test the webservice for the various http requests	56	
7	Design a Restful webservice from a database table Student with columns rollno, name and totalmarks. Create a restful client that displays the data by accessing restful service.	64	
8	Create a WCF service to perform calculations like Addition, Subtraction, Multiplication and Division. Create a client for WCF which invokes the various operations.	71	
9	Create a WCF service with different endpoint for Soap based and Rest based implementation	80	
10	Design a Restful webservice and create a restful client that displays the data by fetching using the HTTP GET method	85	

WEB SERVICES

PRACTICAL NO. 1

TCS2223013

CHIRAG BHATIA

Aim:

Create a TimeServer webservice in Java and Consume it in other technologies like php and .NET

Java Server + Client

Code:

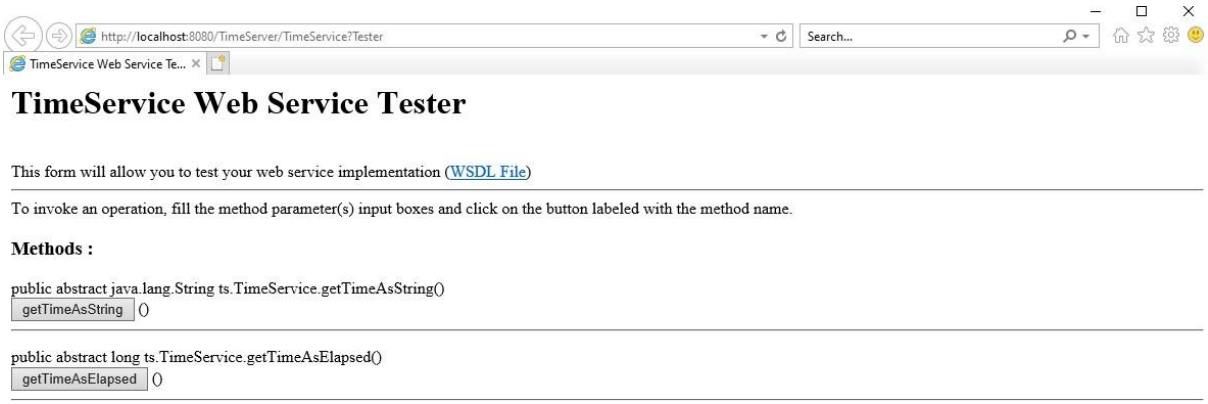
```
/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ts;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

/**
 *
 * @author chira
 */
@WebService(serviceName = "TimeService")
```

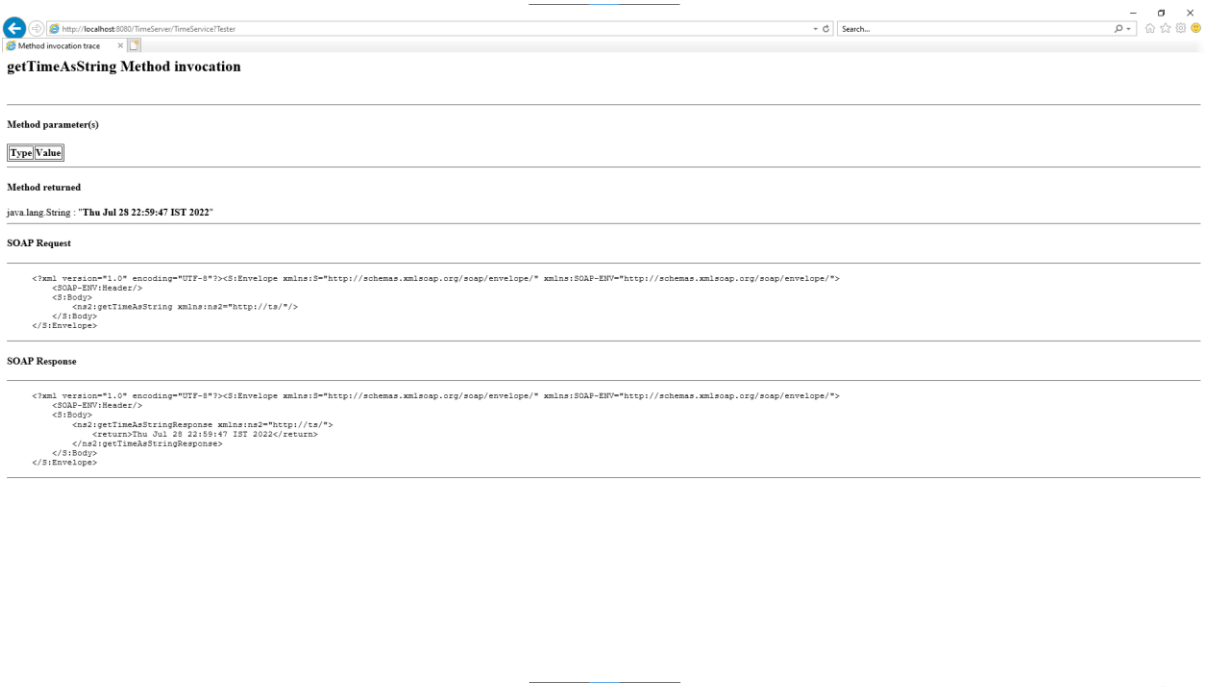
```
public class TimeService {

    /**
     * This is a sample web service operation
     */
    @WebMethod(operationName = "getTimeAsString")
    public String getTimeAsString() {
        //TODO write your implementation code here:
        return new java.util.Date().toString();
    }

    @WebMethod(operationName = "getTimeAsElapsed")
    public long getTimeAsElapsed() {
        //TODO write your implementation code here:
        return new java.util.Date().getTime();
    }
}
```

```
<%--
    Document   : index
    Created on : 28 Jul, 2022, 8:22:43 PM
    Author     : chira
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <h1>Hello World!</h1>
            <%-- start web service invocation --%><hr/>
    <%
    try {
        ts.TimeService_Service service = new ts.TimeService_Service();
        ts.TimeService port = service.getTimeServicePort();
        // TODO process result here
        java.lang.String result = port.getTimeAsString();
        out.println("Result = "+result);
    } catch (Exception ex) {
        // TODO handle custom exceptions here
    }
    %>
    <%-- end web service invocation --%><hr/>
    <%-- start web service invocation --%><hr/>
    <%
    try {
        ts.TimeService_Service service = new ts.TimeService_Service();
        ts.TimeService port = service.getTimeServicePort();
        // TODO process result here
        long result = port.getTimeAsElapsed();
        out.println("Result = "+result);
    } catch (Exception ex) {
```

```
            // TODO handle custom exceptions here
        }
        %>
        <%-- end web service invocation --%><hr/>

        </body>
</html>
```

Output:





http://localhost:8080/TimeServer/TimeServiceTester

Search...

Method invocation trace

getTimeAsString Method invocation

Method parameter(s)

Type	Value

Method returned

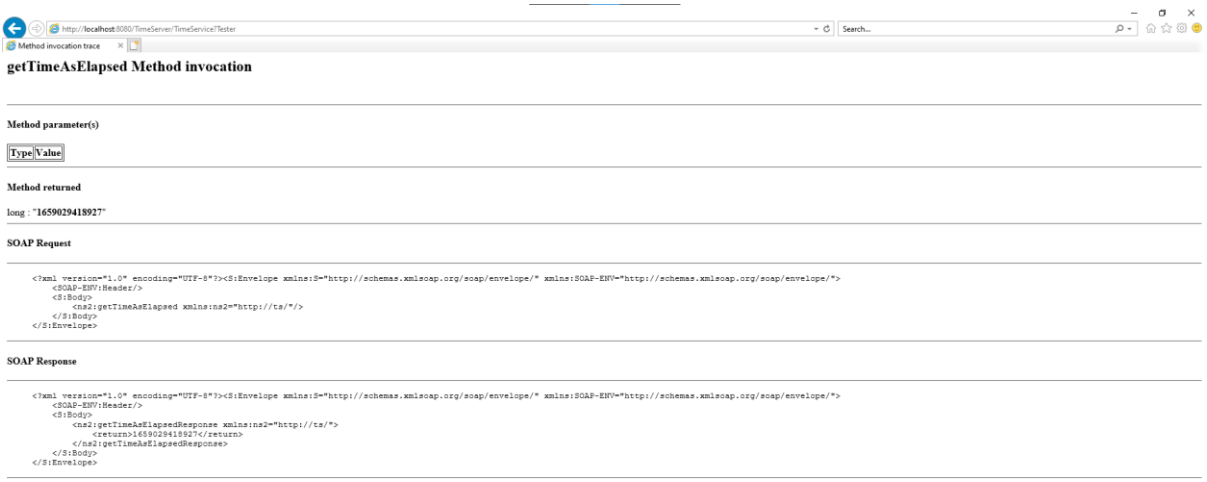
java.lang.String : "**Thu Jul 28 22:59:47 IST 2022**"

SOAP Request

```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header/>
    <S:Body>
        <ns2:getTimeAsString xmlns:ns2="http://ts/"/>
    </S:Body>
</S:Envelope>
```

SOAP Response

```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header/>
    <S:Body>
        <ns2:getTimeAsStringResponse xmlns:ns2="http://ts/">
            <return>Thu Jul 28 22:59:47 IST 2022</return>
        </ns2:getTimeAsStringResponse>
    </S:Body>
</S:Envelope>
```



http://localhost:8080/TimeServer/TimeServiceTester

Search...

Method invocation trace

getTimeAsElapsed Method invocation

Method parameter(s)

Type	Value

Method returned

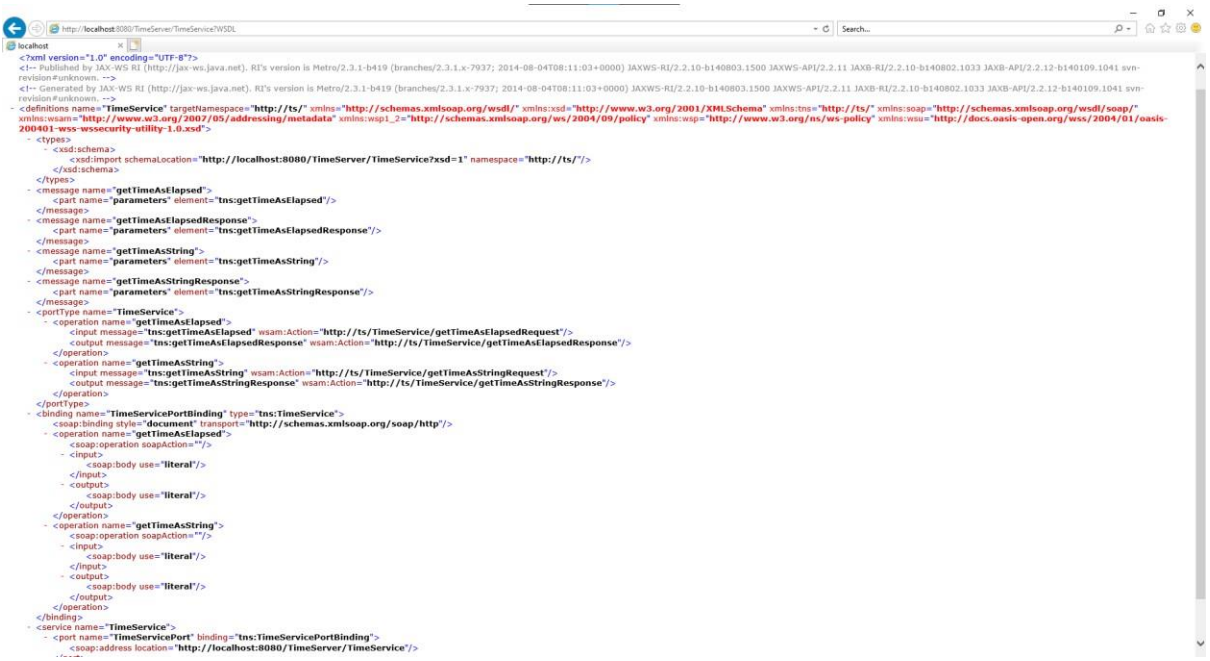
long : "**1659029418927**"

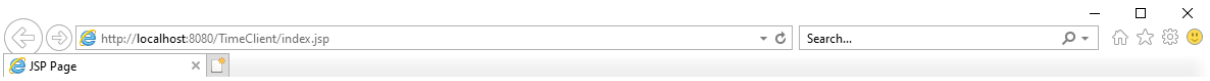
SOAP Request

```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header/>
    <S:Body>
        <ns2:getTimeAsElapsed xmlns:ns2="http://ts/"/>
    </S:Body>
</S:Envelope>
```

SOAP Response

```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header/>
    <S:Body>
        <ns2:getTimeAsElapsedResponse xmlns:ns2="http://ts/">
            <return>1659029418927</return>
        </ns2:getTimeAsElapsedResponse>
    </S:Body>
</S:Envelope>
```





Hello World!

Result = Thu Jul 28 23:01:15 IST 2022

Result = 1659029475813

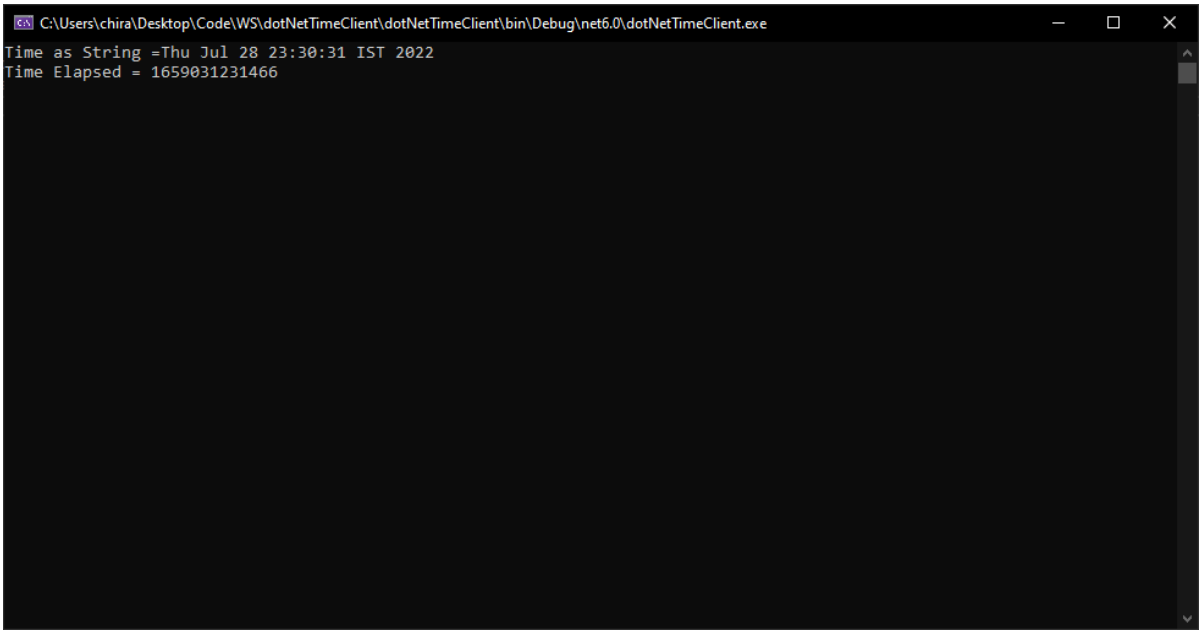
.NET Client

Code:

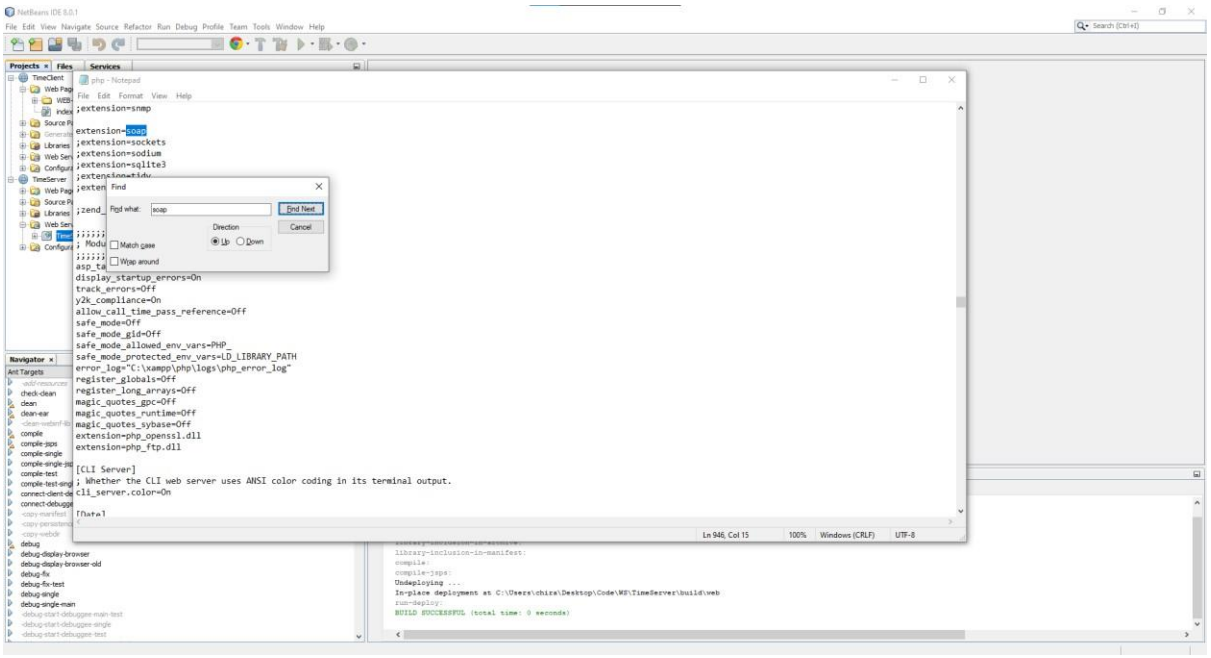
```
using System;
```

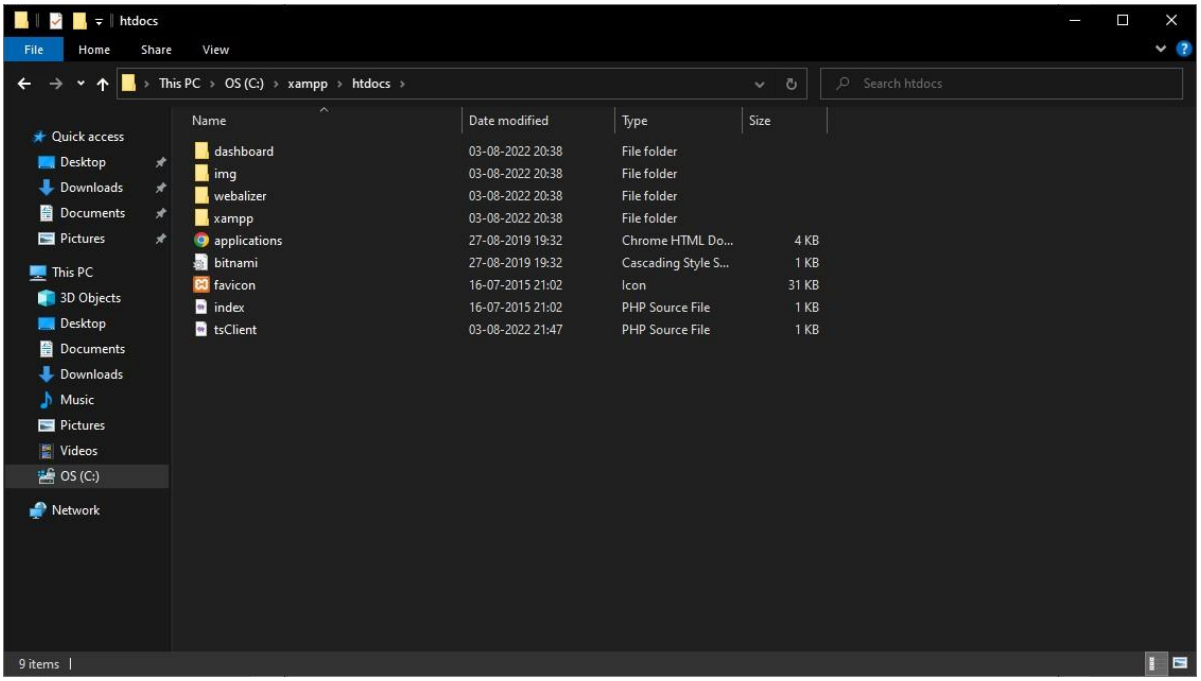
```
namespace TimeServiceClient
{
    class Program
    {
        static void Main(string[] args)
        {
            ServiceReference1.TimeServiceClient client = new
ServiceReference1.TimeServiceClient();
            Console.WriteLine(value: "Time as String =" +
client.getTimeAsString());
            Console.WriteLine(value: "Time Elapsed = " +
client.getTimeAsElapsed());
            Console.Read();
        }
    }
}
```

Output:



PHP Client





Code:

```php
<?php

$client = new
SoapClient("http://localhost:8080/TimeServer/TimeService?WSDL");

$t1 = $client -> getTimeAsElapsed();

echo "Elapsed time is ",$t1 -> return;
```

```
$t2 = $client -> getTimeAsString();

echo "<br>Time as String is ",$t2 -> return;

?>
```

Output:

Elapsed time is 1659543468052
Time as String is Wed Aug 03 21:47:48 IST 2022

WEB SERVICES

PRACTICAL NO. 2

TCS2223013

CHIRAG BHATIA

Aim:

Create a Java WS for performing basic calculations like addition, subtraction, multiplication and division and create a java client that consumes the same.

Code:

Java Server:

```
package cs;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

@WebService(serviceName = "CalcService")
public class CalcService {

    @WebMethod(operationName = "addition")
    public float addition(@WebParam(name = "x") float x, @WebParam(name =
"y") float y) {
        return x + y;
    }
```

```
    @WebMethod(operationName = "subtraction")
    public float subtraction(@WebParam(name = "x") float x, @WebParam(name
= "y") float y) {
        return x - y;
    }

    @WebMethod(operationName = "multiplication")
    public float multiplication(@WebParam(name = "x") float x,
@WebParam(name = "y") float y) {
        return x * y;
    }

    @WebMethod(operationName = "division")
    public float division(@WebParam(name = "x") float x, @WebParam(name =
"y") float y) {
        return x / y;
    }

}
```

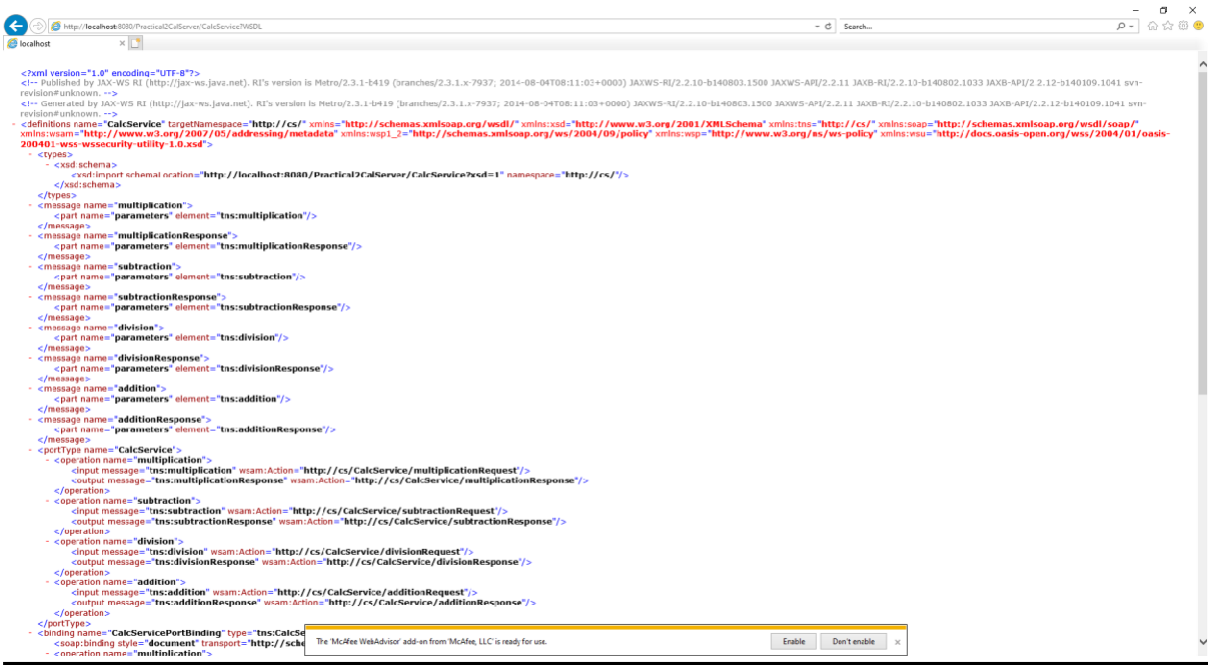
Java Client:

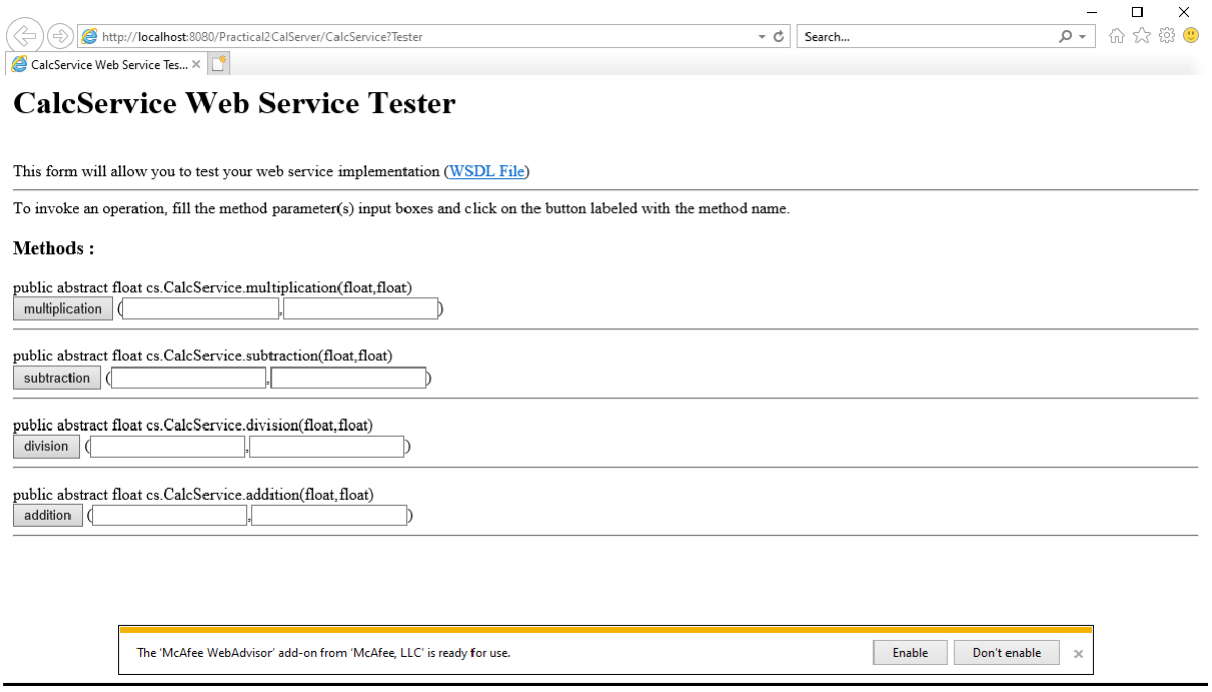
```
<!DOCTYPE html>
<html>
    <head>
        <title>CalClient</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    </head>
    <body>
        <form action="logic.jsp">
            Enter first number<input type="text" id="x" name="x" ><br>
            Enter second number<input type="text" id="y" name="y" ><br>
            <input type="radio" name="RadiobtnGroup" value="Add" />Add<br>
            <input type="radio" name="RadiobtnGroup" value="Subtract"
/>Subtract<br>
            <input type="radio" name="RadiobtnGroup" value="Multiply"
/>Multiply<br>
            <input type="radio" name="RadiobtnGroup" value="Divide"
/>Divide<br>
            <input type="submit" value="Calculate" name="Submit" />
        </form>
    </body>
</html>
```

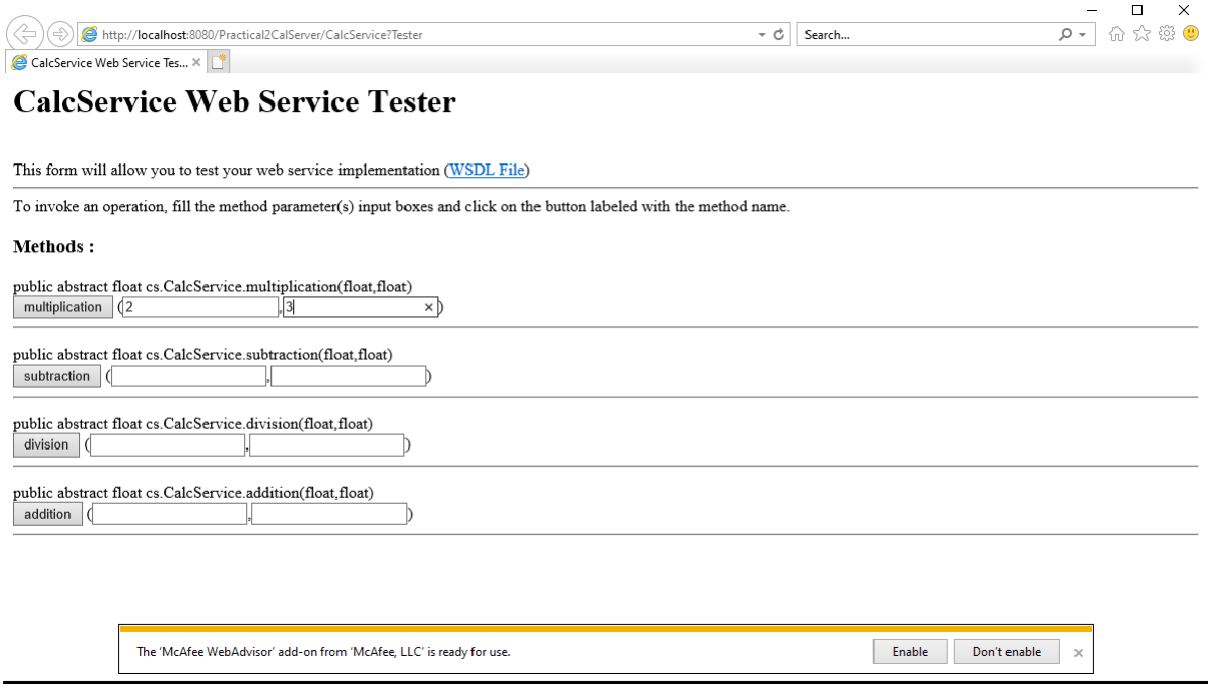
```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
```

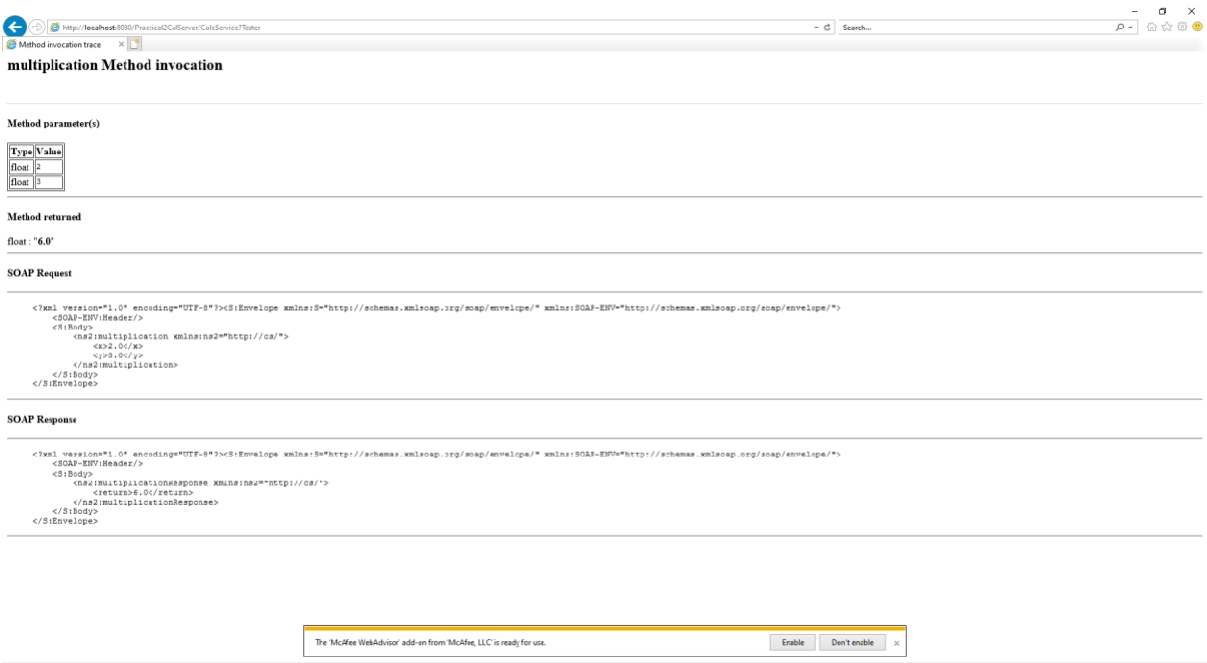
```
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Logic</title>
    </head>
    <body>
        <%
            int a = Integer.parseInt(request.getParameter("x"));
            int b = Integer.parseInt(request.getParameter("y"));
            String choice = request.getParameter("RadiobtnGroup");
            if (choice.equals("Add")) {
                cs.CalcService_Service service = new
cs.CalcService_Service();
                cs.CalcService port = service.getCalcServicePort();
                float x = a;
                float y = b;
                float result = port.addition(x, y);
                out.println("Result = " + result);
            } else if (choice.equals("Subtract")) {
                cs.CalcService_Service service = new
cs.CalcService_Service();
                cs.CalcService port = service.getCalcServicePort();
                float x = a;
                float y = b;
                float result = port.subtraction(x, y);
                out.println("Result = " + result);
            } else if (choice.equals("Multiply")) {
                cs.CalcService_Service service = new
cs.CalcService_Service();
                cs.CalcService port = service.getCalcServicePort();
                float x = a;
                float y = b;
                float result = port.multiplication(x, y);
                out.println("Result = " + result);
            } else if (choice.equals("Divide")) {
                cs.CalcService_Service service = new
cs.CalcService_Service();
                cs.CalcService port = service.getCalcServicePort();
                float x = a;
                float y = b;
                float result = port.division(x, y);
                out.println("Result = " + result);
            }
        %>
    </body>
</html>
```

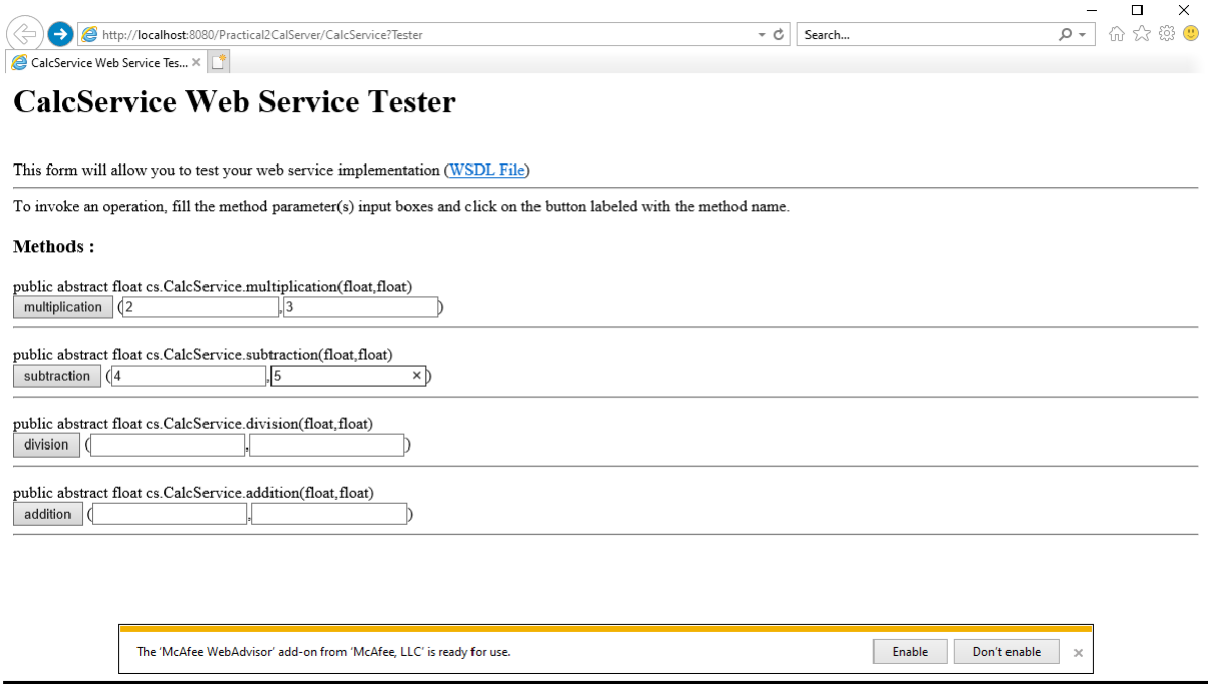
Output:

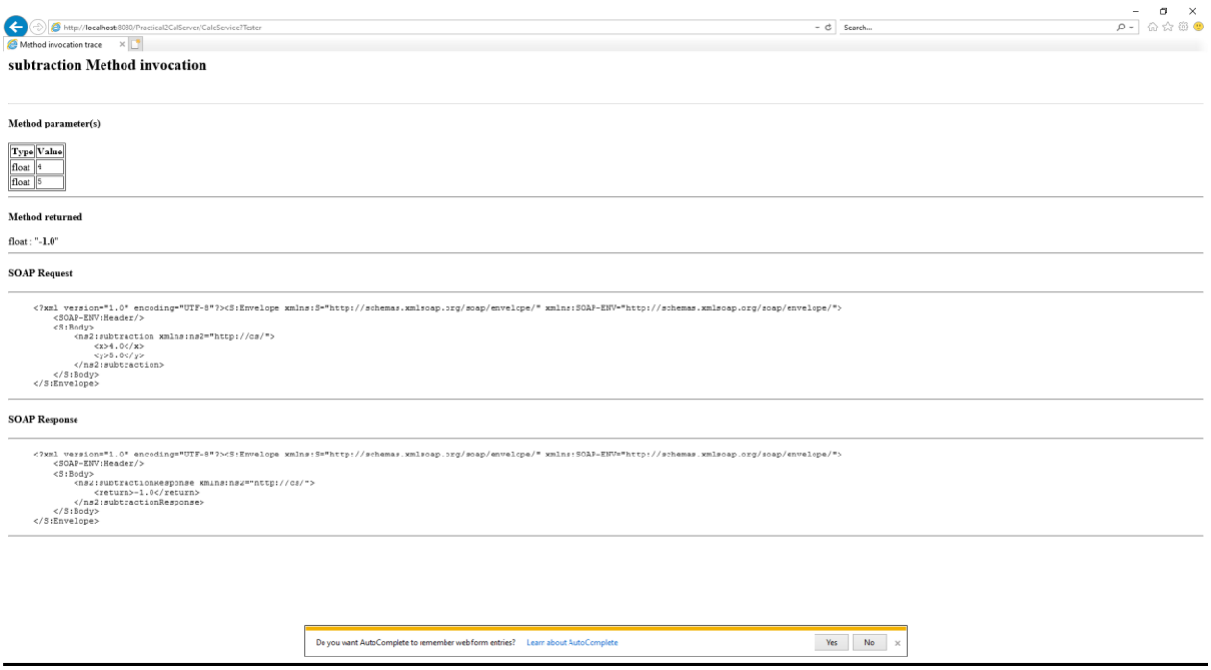


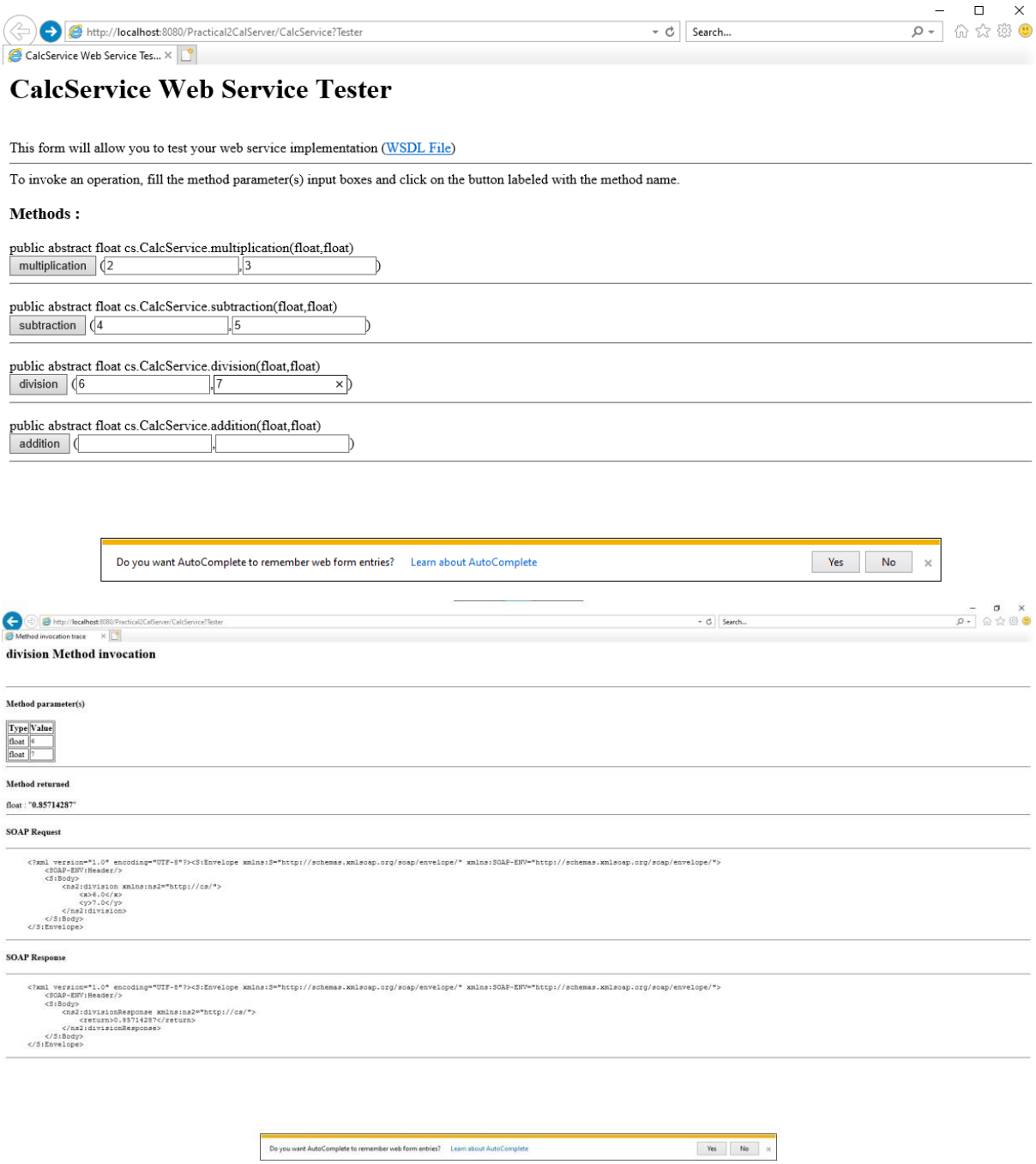

http://localhost:8080/Practical2CalServer/CalcService?Tester
Search...
CalcService Web Service Tes...
CalcService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract float cs.CalcService.multiplication(float,float)
multiplication
public abstract float cs.CalcService.subtraction(float,float)
subtraction
public abstract float cs.CalcService.division(float,float)
division
public abstract float cs.CalcService.addition(float,float)
addition
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable

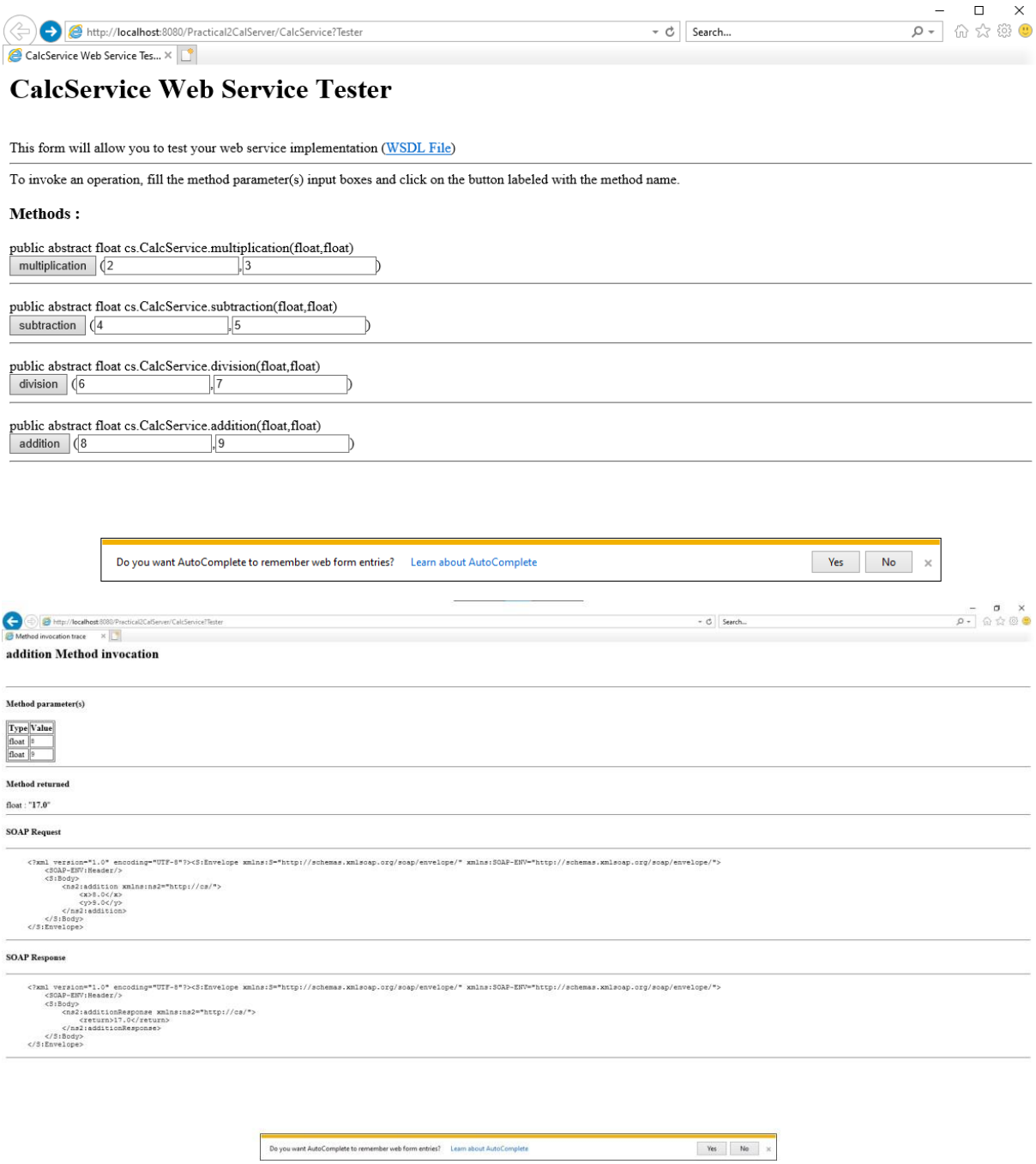

http://localhost:8080/Practical2CalServer/CalcService?Tester
Search...
CalcService Web Service Tes...
CalcService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract float cs.CalcService.multiplication(float,float)
multiplication (2 , 3)
public abstract float cs.CalcService.subtraction(float,float)
subtraction (,)
public abstract float cs.CalcService.division(float,float)
division (,)
public abstract float cs.CalcService.addition(float,float)
addition (,)
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable

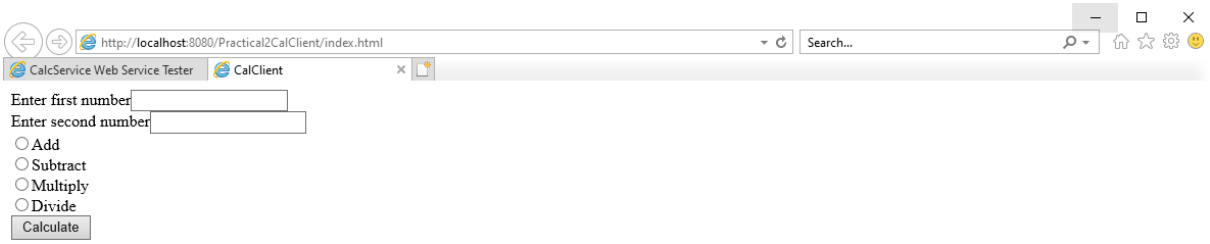

Method invocation trace
multiplication Method invocation
Method parameter(s)
Type
Value
float
2
float
3
Method returned
float : "6.0"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:multiplication xmlns:ns2="http://cs/">
<x>2.0</x>
<y>3.0</y>
</ns2:multiplication>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:multiplicationResponse xmlns:ns2="http://cs/">
<return>6.0</return>
</ns2:multiplicationResponse>
</S:Body>
</S:Envelope>
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable


http://localhost:8080/Practical2CalServer/CalcService?Tester
Search...
CalcService Web Service Tes...
CalcService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract float cs.CalcService.multiplication(float,float)
multiplication (2 , 3)
public abstract float cs.CalcService.subtraction(float,float)
subtraction (4 , 5)
public abstract float cs.CalcService.division(float,float)
division (,)
public abstract float cs.CalcService.addition(float,float)
addition (,)
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable

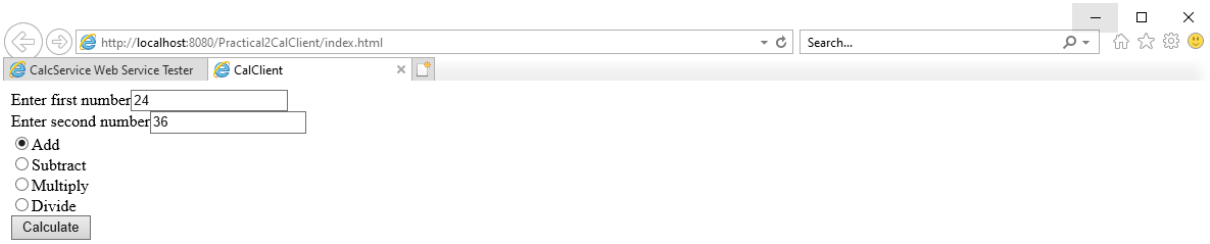

Method invocation trace
subtraction Method invocation
Method parameter(s)
Type
Value
float
4
float
5
Method returned
float : "-1.0"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:subtraction xmlns:ns2="http://cs/">
<x>4.0</x>
<y>5.0</y>
</ns2:subtraction>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:subtractionresponse xmlns:ns2="http://cs/">
<return>-1.0</return>
</ns2:subtractionResponse>
</S:Body>
</S:Envelope>
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/Practical2CalServer/CalcService?Tester
Search...
CalcService Web Service Tes...
CalcService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract float cs.CalcService.multiplication(float,float)
multiplication (2 , 3)
public abstract float cs.CalcService.subtraction(float,float)
subtraction (4 , 5)
public abstract float cs.CalcService.division(float,float)
division (6 , 7)
public abstract float cs.CalcService.addition(float,float)
addition (,)
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No
division Method invocation
Method parameter(s)
Type
Value
float
6
float
7
Method returned
float : "0.85714287"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:division xmlns:ns2="http://cs/">
<x>6.0</x>
<y>7.0</y>
</ns2:division>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:divisionResponse xmlns:ns2="http://cs/">
<return>0.85714287</return>
</ns2:divisionResponse>
</S:Body>
</S:Envelope>

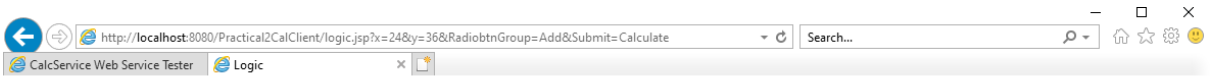

http://localhost:8080/Practical2CalServer/CalcService?Tester
Search...
CalcService Web Service Tes...
CalcService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract float cs.CalcService.multiplication(float,float)
multiplication (2 , 3)
public abstract float cs.CalcService.subtraction(float,float)
subtraction (4 , 5)
public abstract float cs.CalcService.division(float,float)
division (6 , 7)
public abstract float cs.CalcService.addition(float,float)
addition (8 , 9)
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No
addition Method invocation
Method parameter(s)
Type
Value
float
8
float
9
Method returned
float : "17.0"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:addition xmlns:ns2="http://cs/">
<x>8.0</x>
<y>9.0</y>
</ns2:addition>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:additionResponse xmlns:ns2="http://cs/">
<return>17.0</return>
</ns2:additionResponse>
</S:Body>
</S:Envelope>

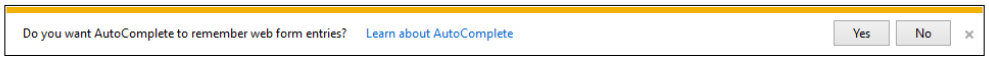

http://localhost:8080/Practical2CalClient/index.html
Search...
CalcService Web Service Tester
CalClient
Enter first number
Enter second number
Add
Subtract
Multiply
Divide
Calculate

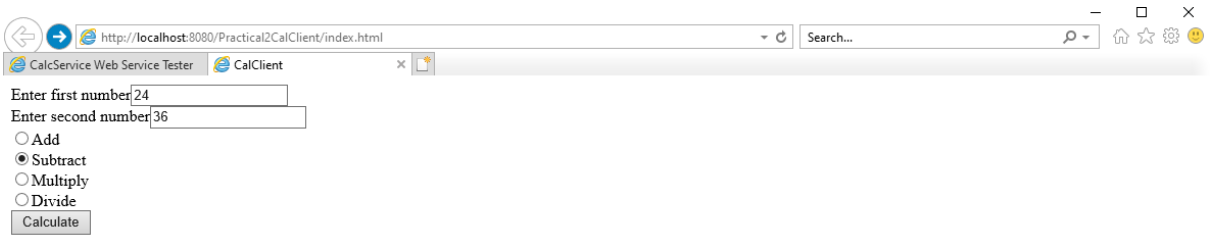

The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable


http://localhost:8080/Practical2CalClient/index.html
Search...
CalcService Web Service Tester
CalClient
Enter first number 24
Enter second number 36
Add
Subtract
Multiply
Divide
Calculate


The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable

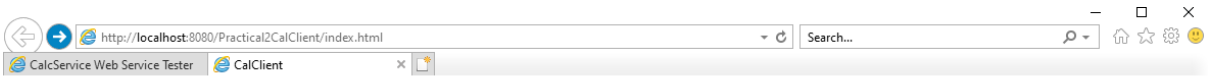

http://localhost:8080/Practical2CalClient/logic.jsp?x=24&y=36&RadiobtnGroup=Add&Submit=Calculate
Search...
CalcService Web Service Tester
Logic
Result = 60.0


Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/Practical2CalClient/index.html
Search...
CalcService Web Service Tester
CalClient
Enter first number 24
Enter second number 36
Add
Subtract
Multiply
Divide
Calculate

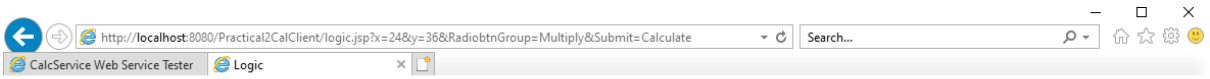

The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable

Result = -12.0

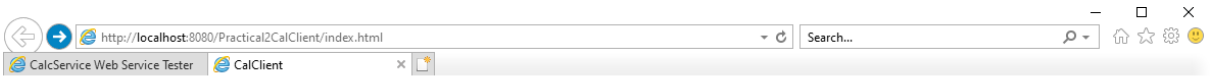


Enter first number 24
Enter second number 36
- ○ Add
- ○ Subtract
- ◉ Multiply
- ○ Divide

Calculate

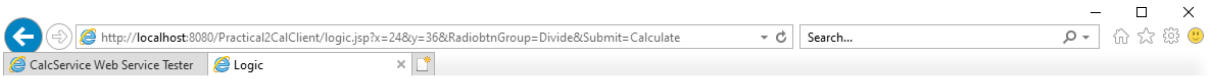


Result = 864.0



Enter first number 24
Enter second number 36
- ○ Add
- ○ Subtract
- ○ Multiply
- ◉ Divide

Calculate



Result = 0.6666667

WEB SERVICES

PRACTICAL NO. 3

TCS2223013

CHIRAG BHATIA

Aim:

Create a web service that gives – (i) NSE Index, (ii) BSE Index, (iii)Gold Rate. The values are stored in database. Also create a web client for a share trading firm that displays these values on its home page

Code:

Server:

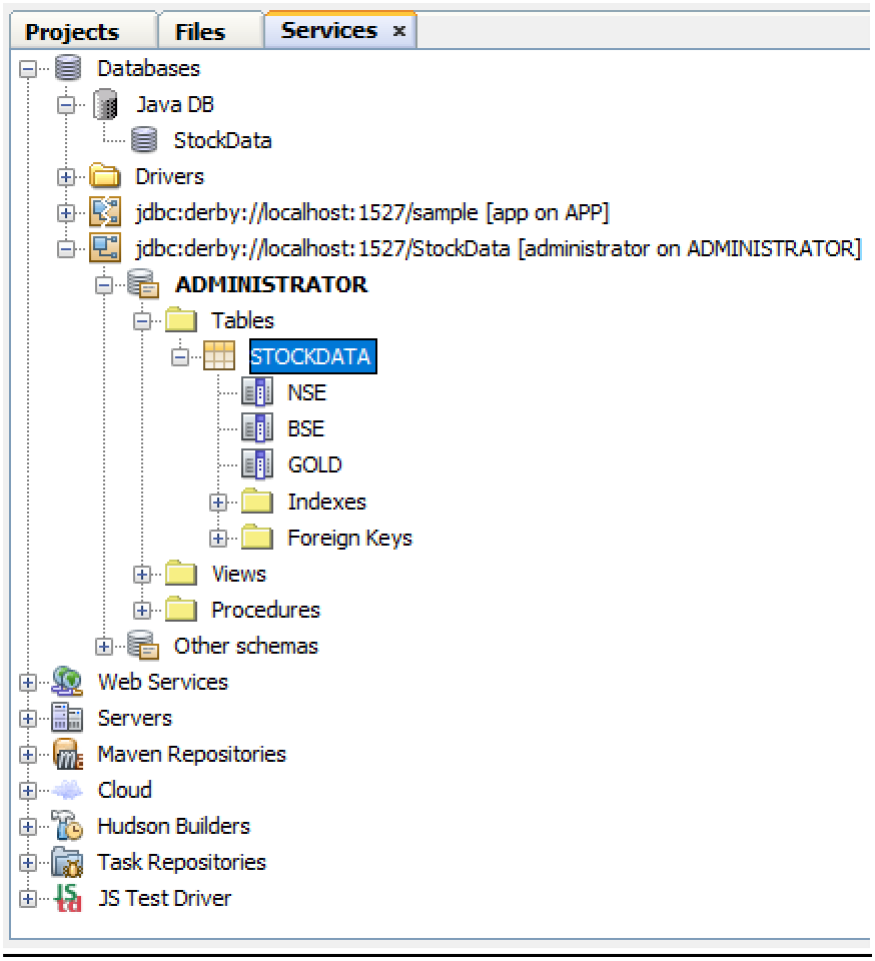
```
package ss;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;
import java.sql.*;

@WebService(serviceName = "StockService")
public class StockService {

    @WebMethod(operationName = "getNSE")
    public long getNSE() {
        long nse = 0;
```

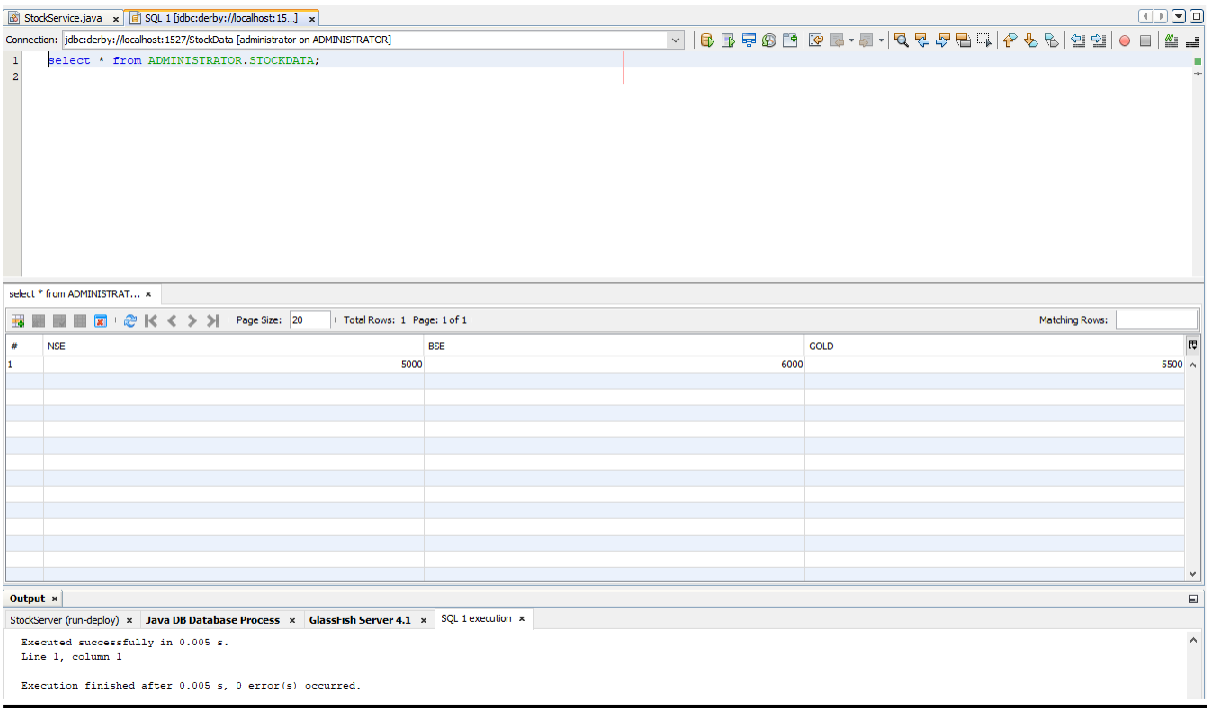
```
        try   {
            Class.forName("org.apache.derby.jdbc.ClientDriver");
            Connection con =
DriverManager.getConnection("jdbc:derby://localhost:1527/StockData",
"administrator", "administrator");
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("select * from stockdata");
            rs.next();
            nse = rs.getInt("NSE");
        } catch (Exception e) {
            e.printStackTrace();
        }
        return nse;
    }

    @WebMethod(operationName = "getBSE")
    public long getBSE() {
        long bse = 0;
        try {
            Class.forName("org.apache.derby.jdbc.ClientDriver");
            Connection con =
DriverManager.getConnection("jdbc:derby://localhost:1527/StockData",
"administrator", "administrator");
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("select * from stockdata");
            rs.next();
            bse = rs.getInt("BSE");
        } catch (Exception e) {
            e.printStackTrace();
        }
        return bse;
    }

    @WebMethod(operationName = "getGold")
    public long getGold() {
        long Goldrate = 0;
        try {
            Class.forName("org.apache.derby.jdbc.ClientDriver");
            Connection con =
DriverManager.getConnection("jdbc:derby://localhost:1527/StockData",
"administrator", "administrator");
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("select * from stockdata");
            rs.next();
            Goldrate = rs.getInt("GOLD");
        } catch (Exception e) {
            e.printStackTrace();
        }
        return Goldrate;
    }
}
```

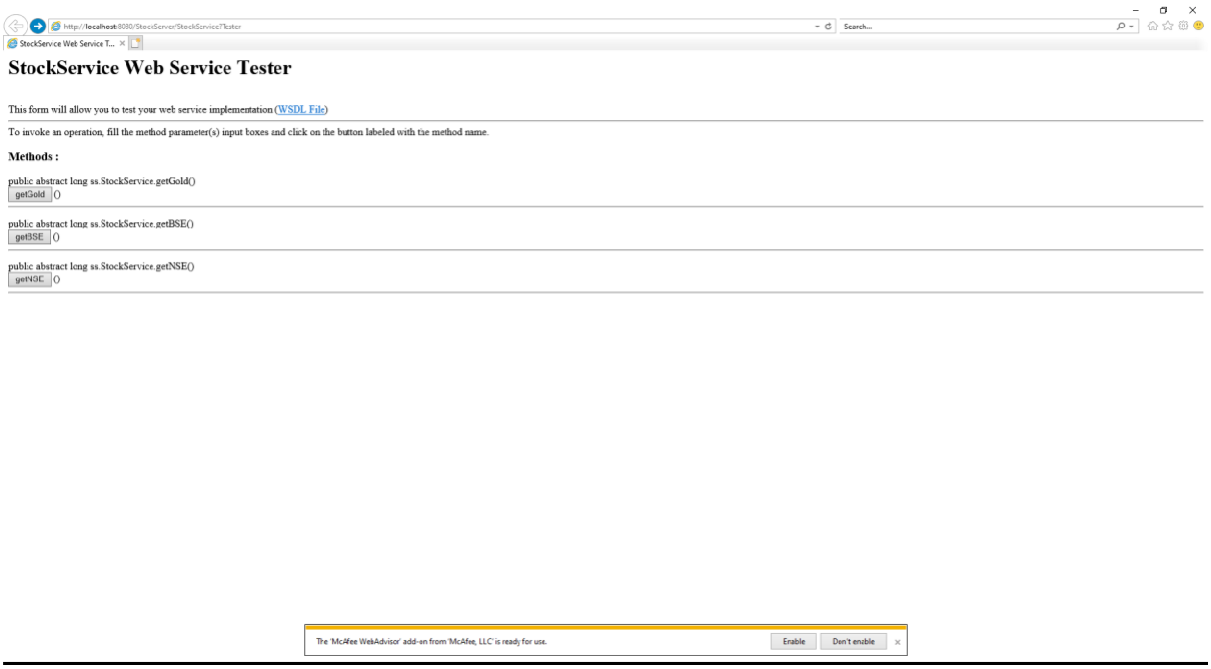
Client:

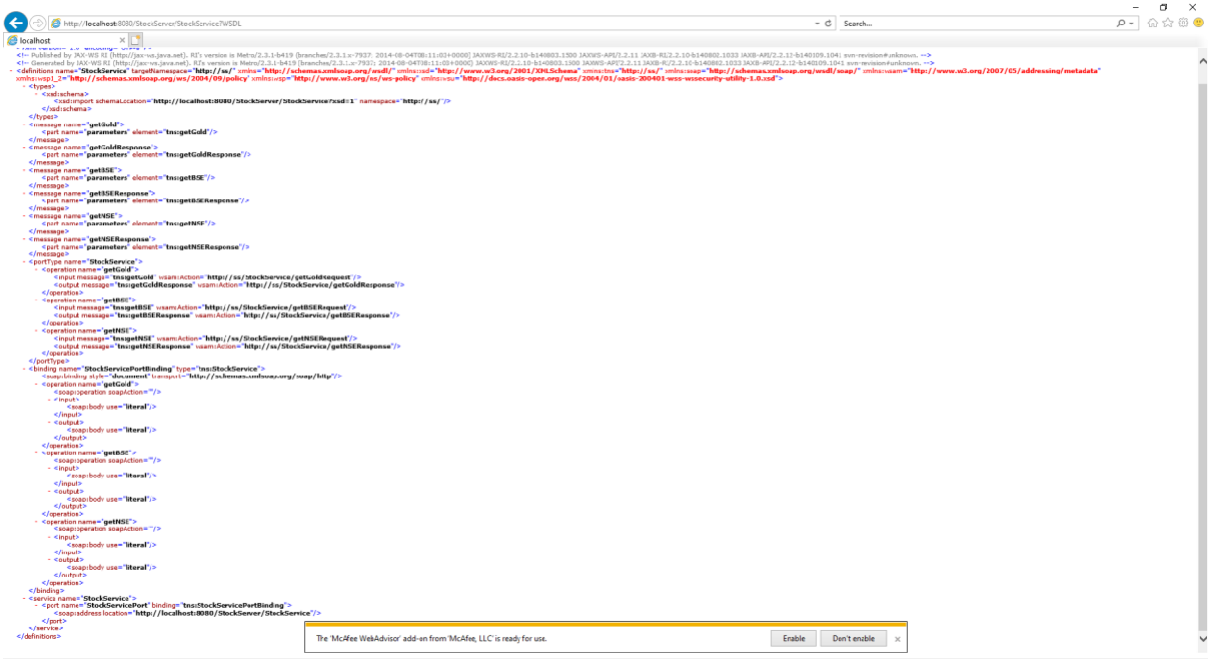
```jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Stocks</title>
    </head>
    <body>
        <h1>BSE</h1>
        <%
            try {
                ss.StockService_Service service = new
ss.StockService_Service();
                ss.StockService port = service.getStockServicePort();
                long result = port.getBSE();
                out.println("Result = " + result);
            } catch (Exception ex) {
            }
        %>
        <h1>NSE</h1>
        <%
            try {
                ss.StockService_Service service = new
ss.StockService_Service();
                ss.StockService port = service.getStockServicePort();
                long result = port.getNSE();
                out.println("Result = " + result);
            } catch (Exception ex) {
            }
        %>
        <h1>GOLD</h1>
        <%
            try {
                ss.StockService_Service service = new
ss.StockService_Service();
                ss.StockService port = service.getStockServicePort();
                long result = port.getGold();
                out.println("Result = " + result);
            } catch (Exception ex) {
            }
        %>
    </body>
</html>
```

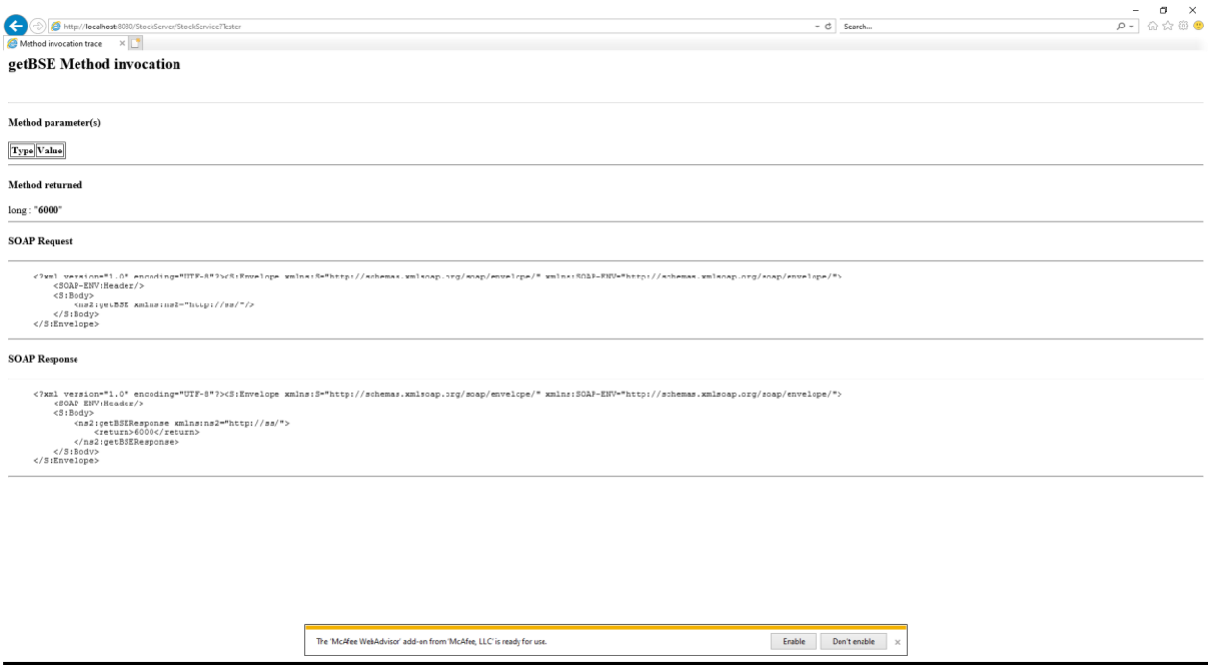
Output:

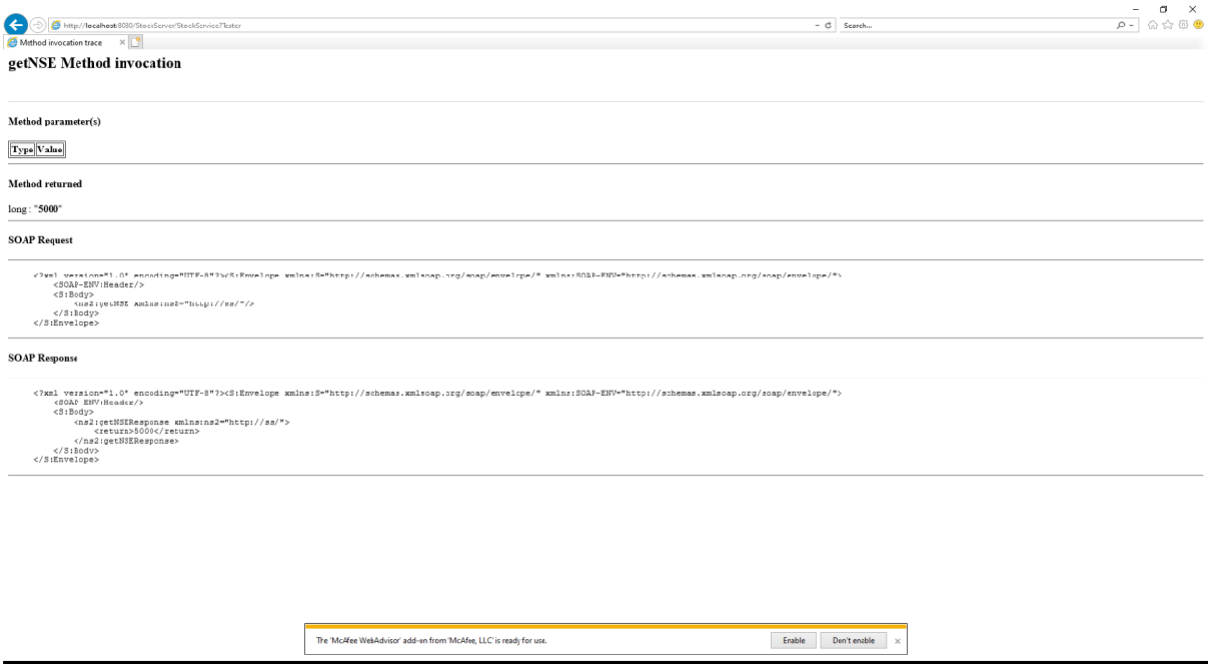

Projects
Files
Services
Databases
Java DB
StockData
Drivers
jdbc:derby://localhost:1527/sample [app on APP]
jdbc:derby://localhost:1527/StockData [administrator on ADMINISTRATOR]
ADMINISTRATOR
Tables
STOCKDATA
NSE
BSE
GOLD
Indexes
Foreign Keys
Views
Procedures
Other schemas
Web Services
Servers
Maven Repositories
Cloud
Hudson Builders
Task Repositories
JS Test Driver

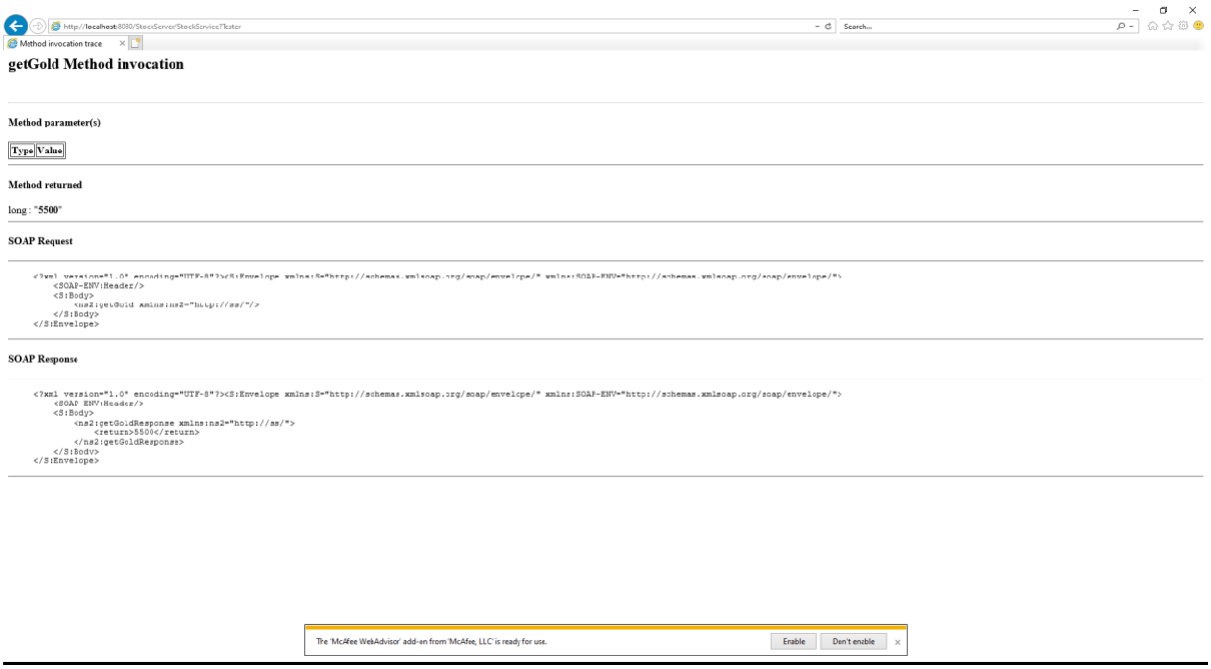

StockService.java
SQL 1 [jdbc:derby://localhost:15...]
Connection: jdbc:derby://localhost:1527/StockData [administrator on ADMINISTRATOR]
select * from ADMINISTRATOR.STOCKDATA;
Page Size: 20
Total Rows: 1 Page: 1 of 1
Matching Rows:
NSE BSE GOLD
1 5000 6000 5500
Output
Executed successfully in 0.005 s.
Line 1, column 1
Execution finished after 0.005 s, 0 error(s) occurred.

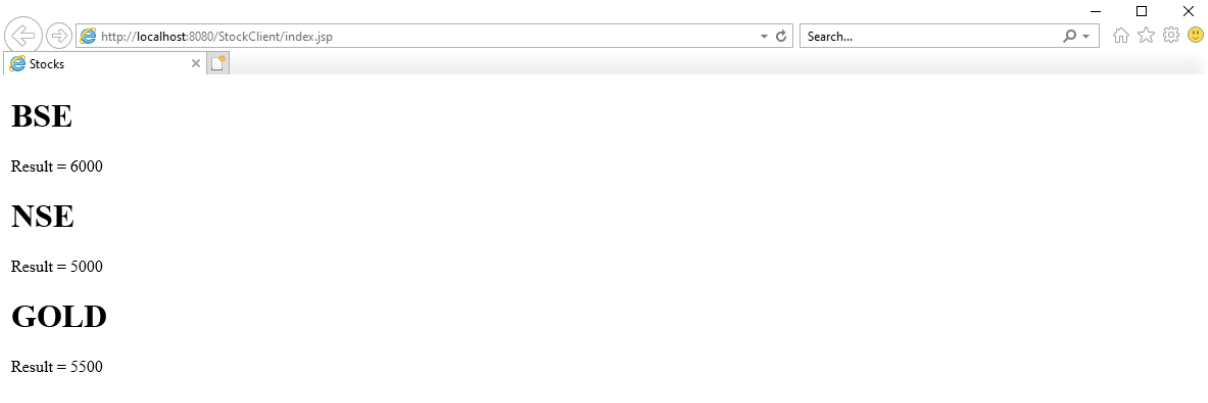

StockService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract long ss.StockService.getGold()
getGold ()
public abstract long ss.StockService.getBSE()
getBSE ()
public abstract long ss.StockService.getNSE()
getNSE ()
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable


localhost
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable


Method invocation trace
getBSE Method invocation
Method parameter(s)
Type
Value
Method returned
long : "6000"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getBSE xmlns:ns2="http://ss/"/>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getBSEResponse xmlns:ns2="http://ss/">
<return>6000</return>
</ns2:getBSEResponse>
</S:Body>
</S:Envelope>
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable


Method invocation trace
getNSE Method invocation
Method parameter(s)
Type
Value
Method returned
long : "5000"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getNSE xmlns:ns2="http://ss/"/>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getNSEResponse xmlns:ns2="http://ss/">
<return>5000</return>
</ns2:getNSEResponse>
</S:Body>
</S:Envelope>
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable


Method invocation trace
getGold Method invocation
Method parameter(s)
Type
Value
Method returned
long : "5500"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getGold xmlns:ns2="http://ss/"/>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getGoldResponse xmlns:ns2="http://ss/">
<return>5500</return>
</ns2:getGoldResponse>
</S:Body>
</S:Envelope>
The 'McAfee WebAdvisor' add-on from 'McAfee, LLC' is ready for use.
Enable
Don't enable


http://localhost:8080/StockClient/index.jsp
Search...
Stocks
BSE
Result = 6000
NSE
Result = 5000
GOLD
Result = 5500

WEB SERVICES

PRACTICAL NO. 4

TCS2223013

CHIRAG BHATIA

Aim:

Create a web service for UGC that contains a method which accepts college name as parameter and returns the NAAC rating. The college names and their ratings are stored in database. Design a web client to test the above web service.

Code:

Server:

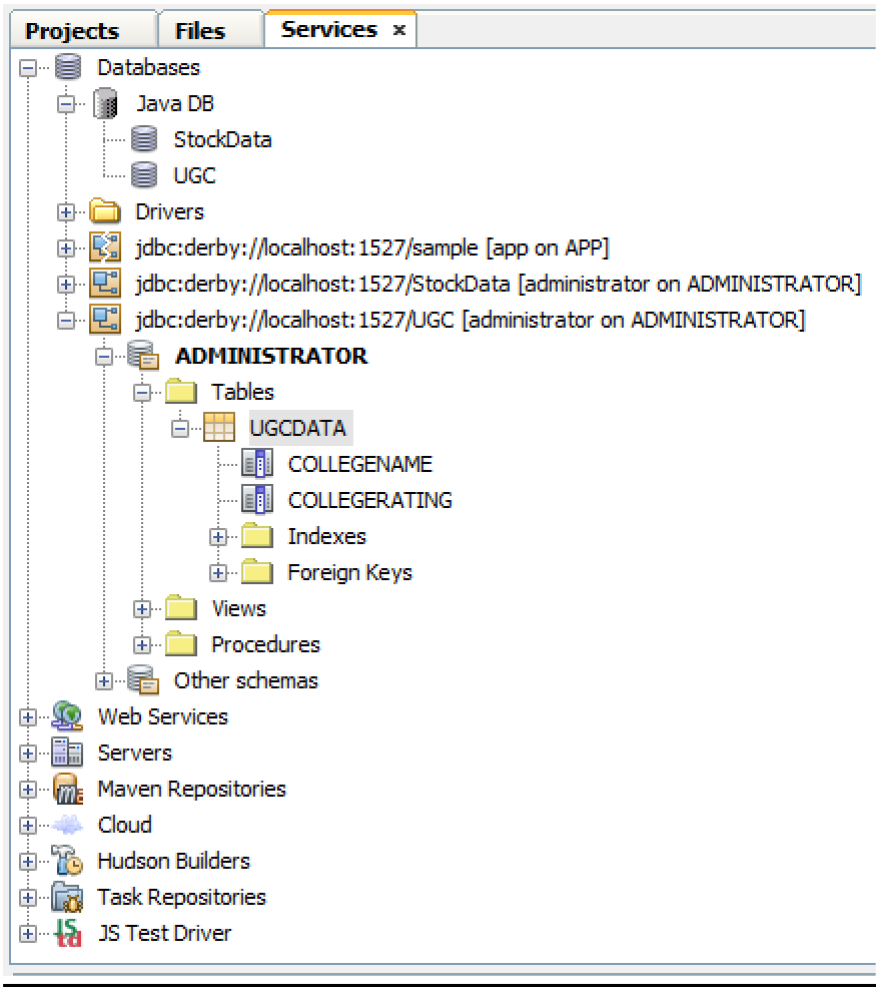
```
package us;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;
import java.sql.*;

@WebService(serviceName = "UGCService")
public class UGCService {
    @WebMethod(operationName = "getRating")
    public String getRating(@WebParam(name = "collegeName") String
collegeName) {
        String rating="";
```

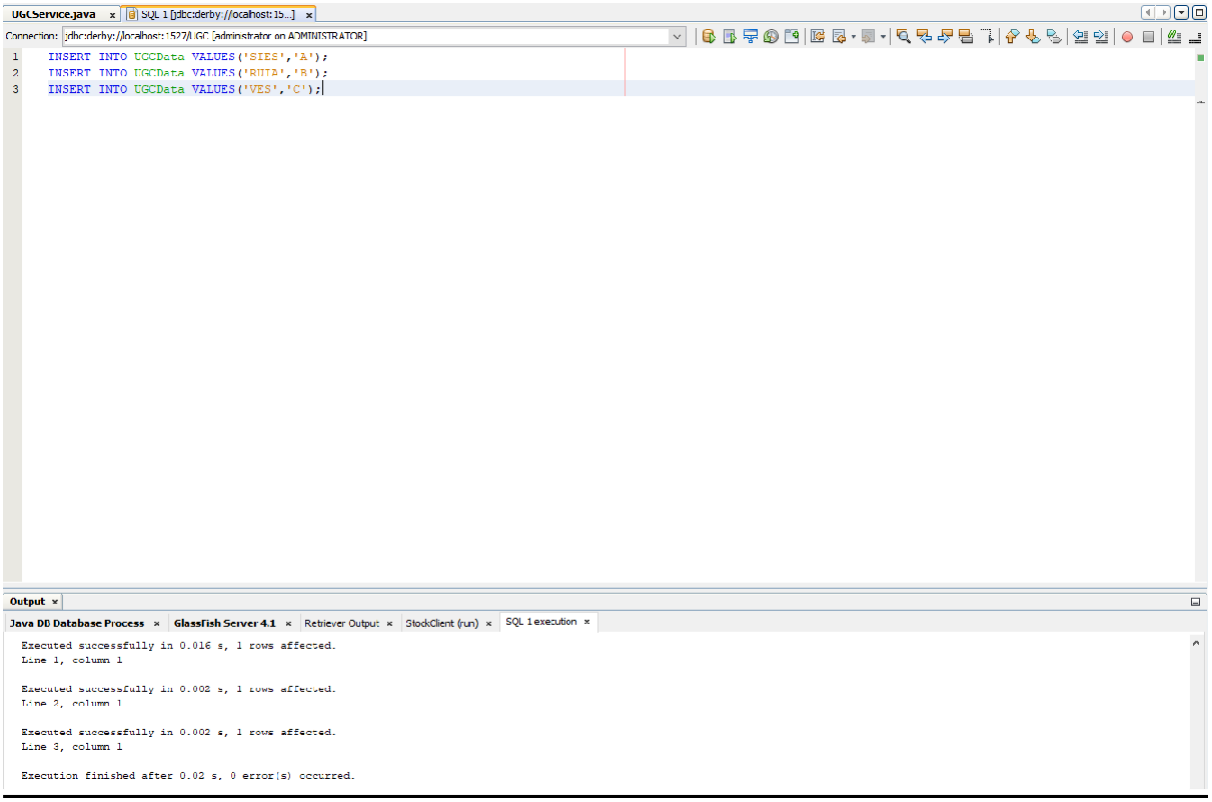
```
        try{
            Class.forName("org.apache.derby.jdbc.ClientDriver");
            Connection con =
DriverManager.getConnection("jdbc:derby://localhost:1527/UGC",
"administrator", "administrator");
            PreparedStatement pstmt=con.prepareStatement("select * from
UGCDATA where COLLEGENAME=?");
            pstmt.setString(1, collegeName);
            ResultSet rs=pstmt.executeQuery();
            rs.next();
            rating=rs.getString("COLLEGERATING");
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
        return rating;
    }
}
```

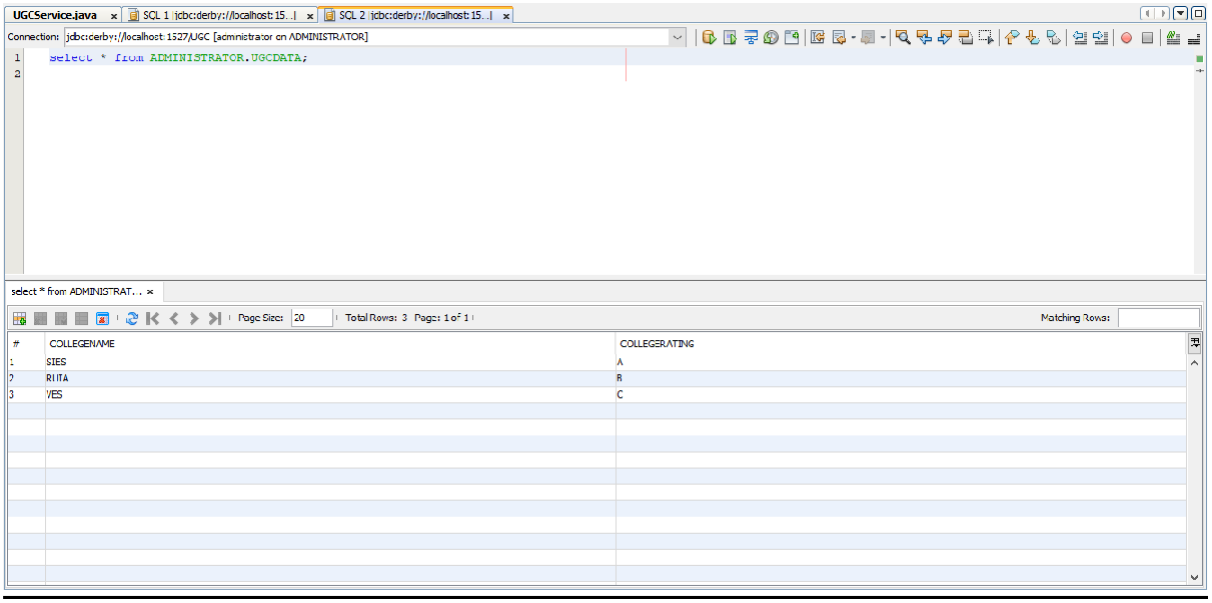
Client:

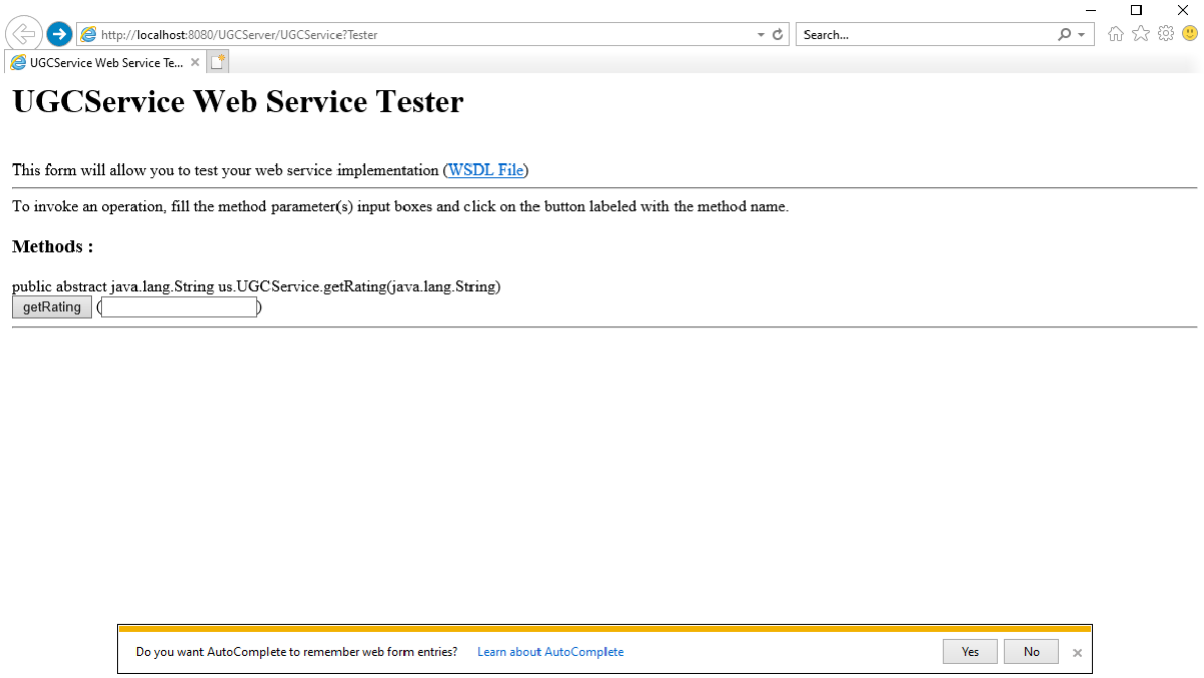
```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>UGC</title>
    </head>
    <body>
        <h1>College Rating</h1>
        <form action="index.jsp" method="POST">
            Enter College Name <input type="text" name="value1">
            <input type="submit" name="btnsubmit" value="Submit"/>
        </form>
        <%
            try {
                us.UGCService_Service service = new
us.UGCService_Service();
                us.UGCService port = service.getUGCServicePort();
                java.lang.String collegeName =
request.getParameter("value1");
                java.lang.String result = port.getRating(collegeName);
                out.println("Result = " + result);
            } catch (Exception ex) {
            }
        %>
    </body>
</html>
```

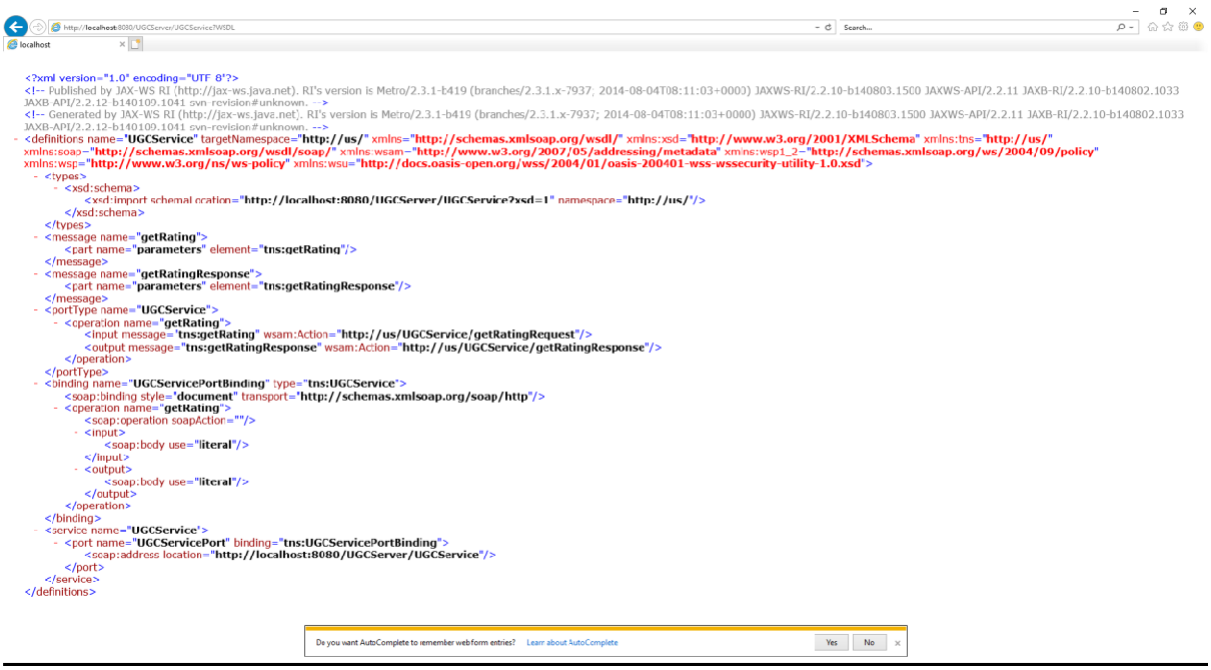
Output:

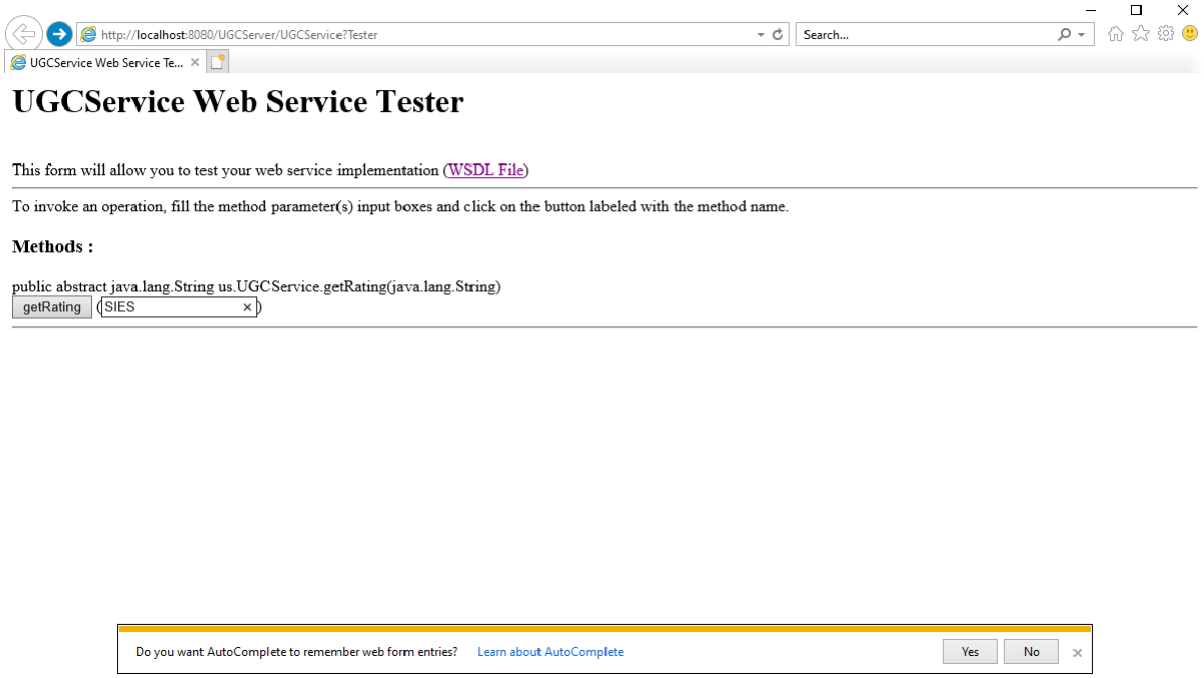

Projects
Files
Services
Databases
Java DB
StockData
UGC
Drivers
jdbc:derby://localhost:1527/sample [app on APP]
jdbc:derby://localhost:1527/StockData [administrator on ADMINISTRATOR]
jdbc:derby://localhost:1527/UGC [administrator on ADMINISTRATOR]
ADMINISTRATOR
Tables
UGCDATA
COLLEGENAME
COLLEGERATING
Indexes
Foreign Keys
Views
Procedures
Other schemas
Web Services
Servers
Maven Repositories
Cloud
Hudson Builders
Task Repositories
JS Test Driver

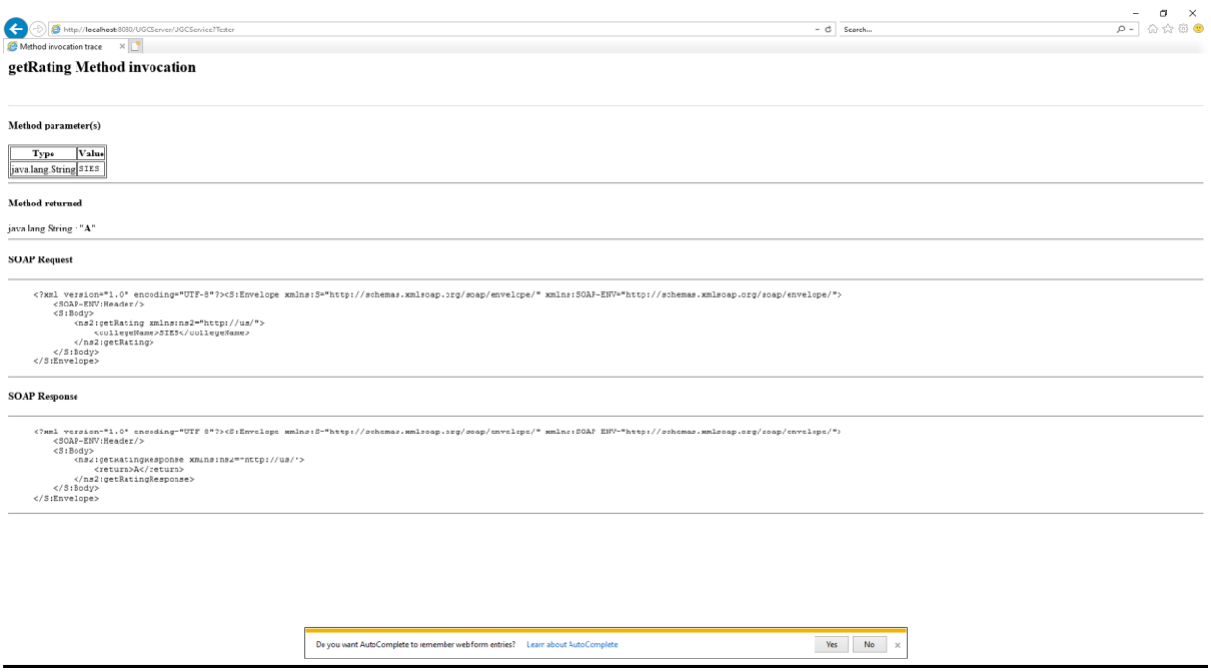

UGCService.java
SQL 1 [jdbc:derby://localhost:15...]
INSERT INTO UGCData VALUES('SIES','A');
INSERT INTO UGCData VALUES('RUIA','B');
INSERT INTO UGCData VALUES('VES','C');
Output
Java DB Database Process
GlassFish Server 4.1
Retriever Output:
StockClient (run)
SQL 1 execution
Executed successfully in 0.016 s, 1 rows affected.
Line 1, column 1
Executed successfully in 0.002 s, 1 rows affected.
Line 2, column 1
Executed successfully in 0.002 s, 1 rows affected.
Line 3, column 1
Execution finished after 0.02 s, 0 error(s) occurred.

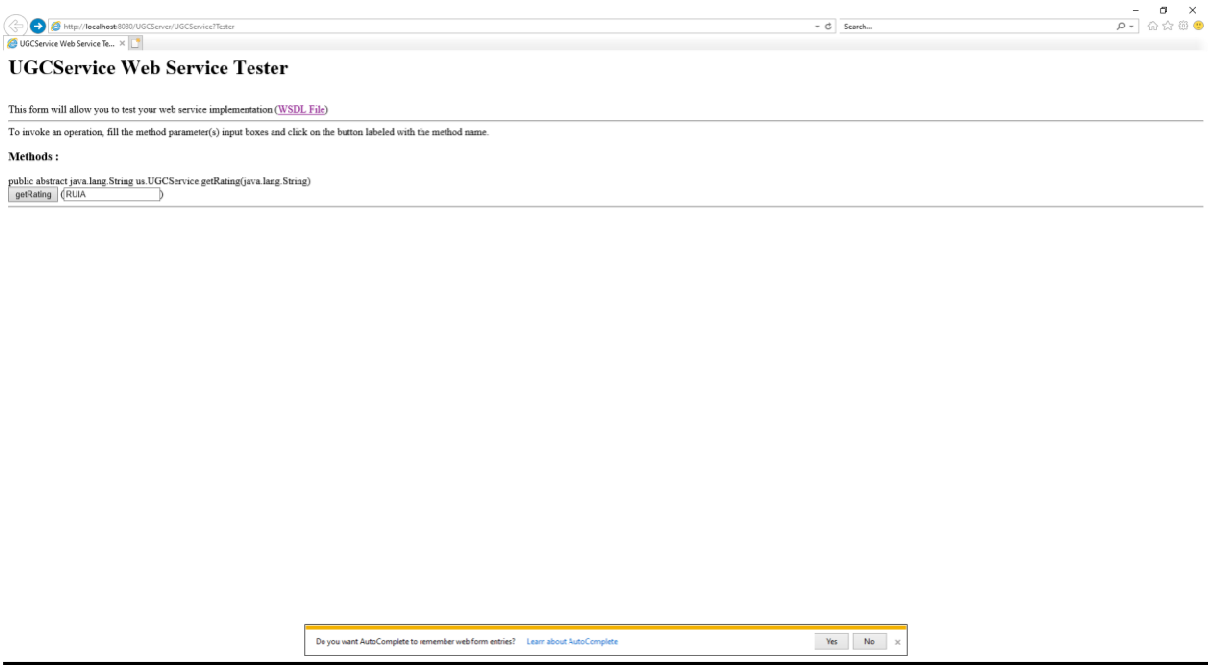

UGCService.java
SQL 1 [jdbc:derby://localhost:15...]
SQL 2 [jdbc:derby://localhost:15...]
Connection: jdbc:derby://localhost:1527/UGC [administrator on ADMINISTRATOR]
select * from ADMINISTRATOR.UGCDATA;
select * from ADMINISTRAT...
Page Size: 20
Total Rows: 3 Page: 1 of 1
Matching Rows:
COLLEGENAME COLLEGERATING
1 SIES A
2 RUIA B
3 VES C

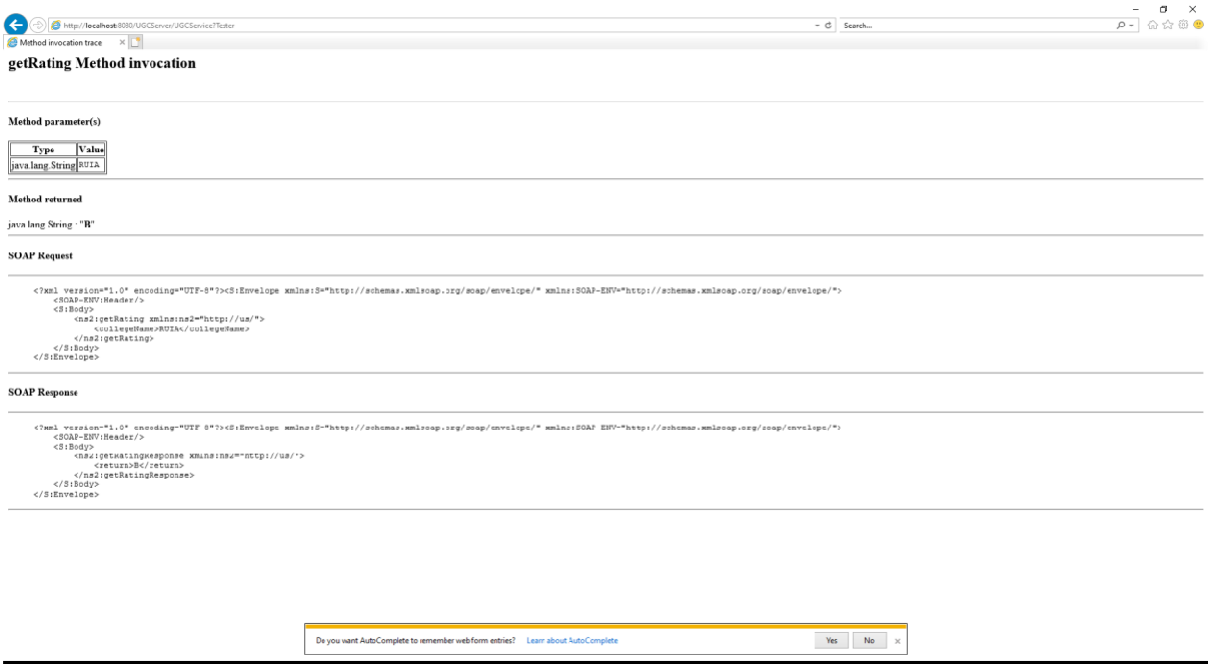

http://localhost:8080/UGCServer/UGCService?Tester
Search...
UGCService Web Service Te...
UGCService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String us.UGCService.getRating(java.lang.String)
getRating
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

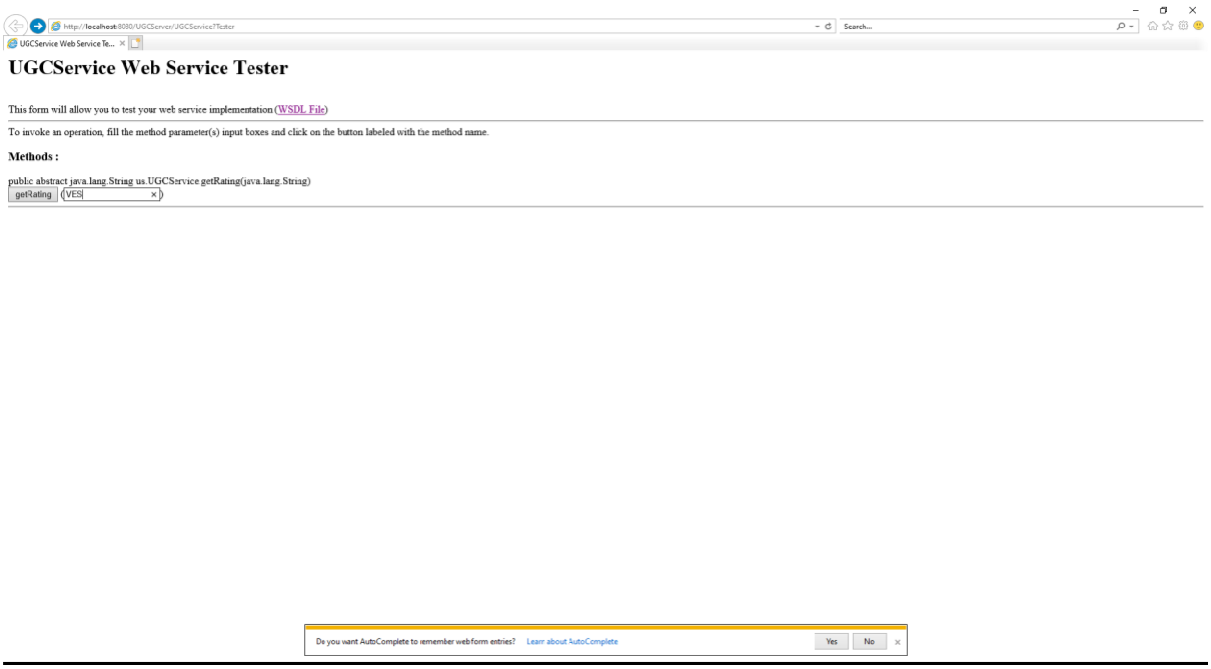

localhost
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="UGCService" targetNamespace="http://us/" xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:tns="http://us/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:wsp="http://www.w3.org/ns/ws-policy" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
<types>
<xsd:schema>
<xsd:import schemaLocation="http://localhost:8080/UGCServer/UGCService?xsd=1" namespace="http://us/"/>
</xsd:schema>
</types>
<message name="getRating">
<part name="parameters" element="tns:getRating"/>
</message>
<message name="getRatingResponse">
<part name="parameters" element="tns:getRatingResponse"/>
</message>
<portType name="UGCService">
<operation name="getRating">
<input message="tns:getRating" wsam:Action="http://us/UGCService/getRatingRequest"/>
<output message="tns:getRatingResponse" wsam:Action="http://us/UGCService/getRatingResponse"/>
</operation>
</portType>
<binding name="UGCServicePortBinding" type="tns:UGCService">
<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
<operation name="getRating">
<soap:operation soapAction=""/>
<input>
<soap:body use="literal"/>
</input>
<output>
<soap:body use="literal"/>
</output>
</operation>
</binding>
<service name="UGCService">
<port name="UGCServicePort" binding="tns:UGCServicePortBinding">
<soap:address location="http://localhost:8080/UGCServer/UGCService"/>
</port>
</service>
</definitions>
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

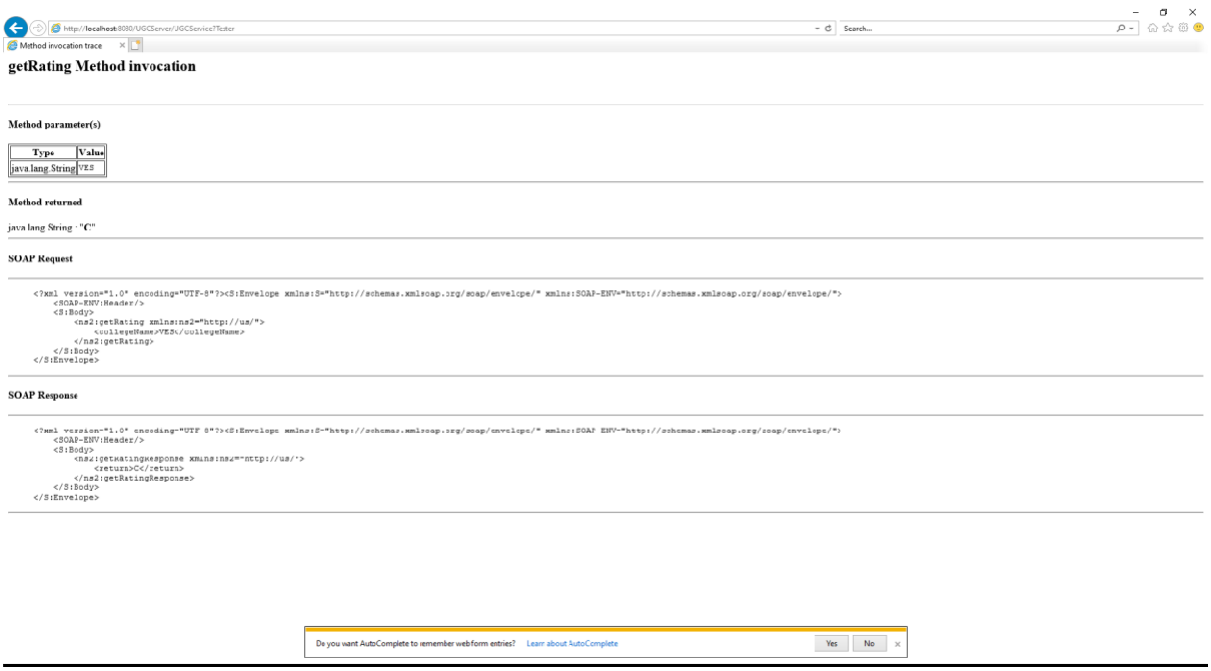

http://localhost:8080/UGCServer/UGCService?Tester
Search...
UGCService Web Service Te...
UGCService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String us.UGCService.getRating(java.lang.String)
getRating
SIES
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

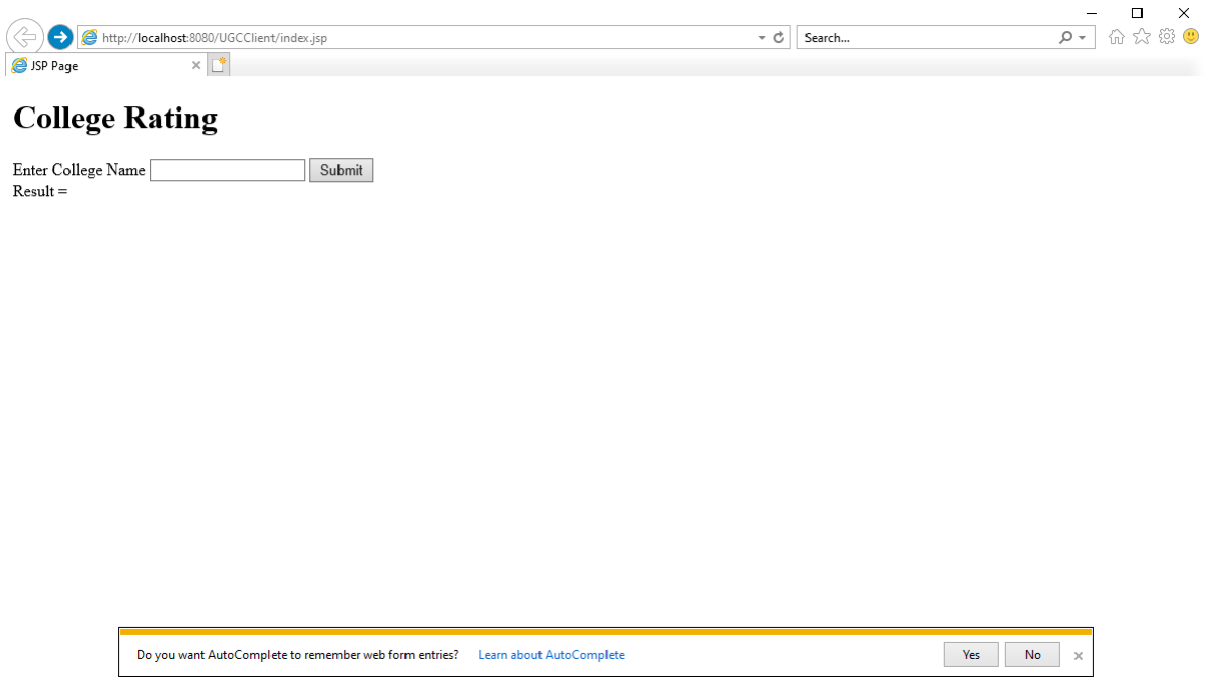

Method invocation trace
getRating Method invocation
Method parameter(s)
Type
Value
java.lang.String
SIES
Method returned
java.lang.String : "A"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getRating xmlns:ns2="http://us/">
<collegeName>SIES</collegeName>
</ns2:getRating>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getRatingResponse xmlns:ns2="http://us/">
<return>A</return>
</ns2:getRatingResponse>
</S:Body>
</S:Envelope>
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

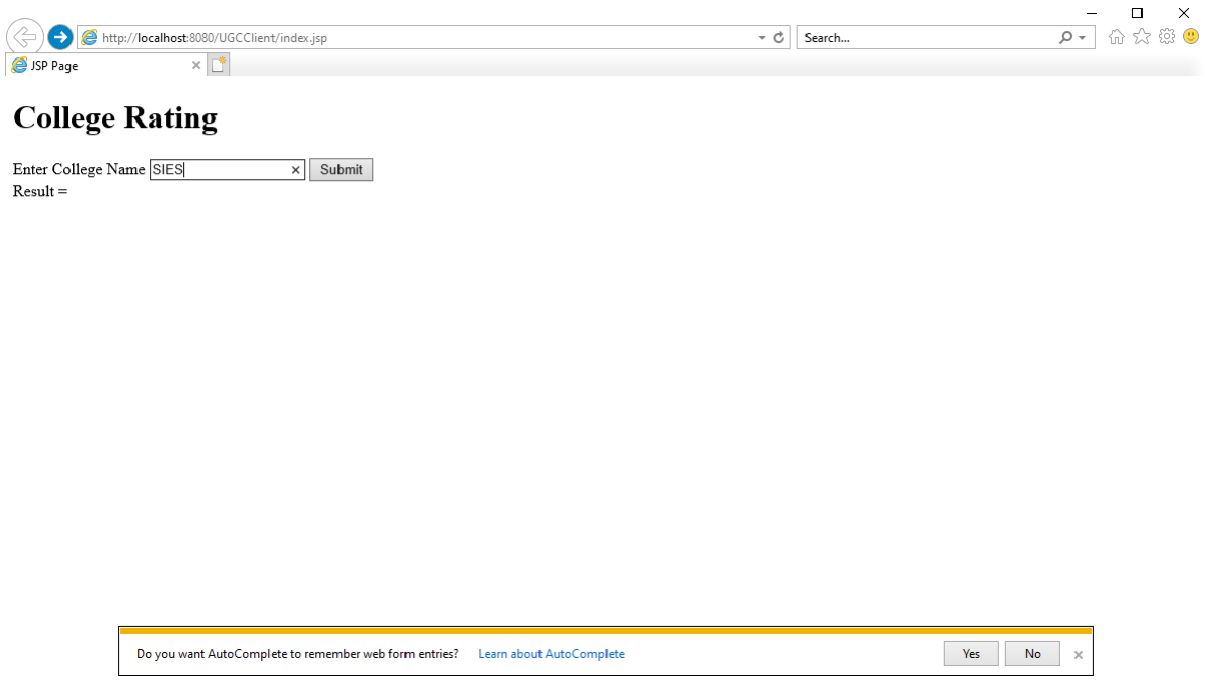

UGCService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String us.UGCService.getRating(java.lang.String)
getRating
RUIA
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

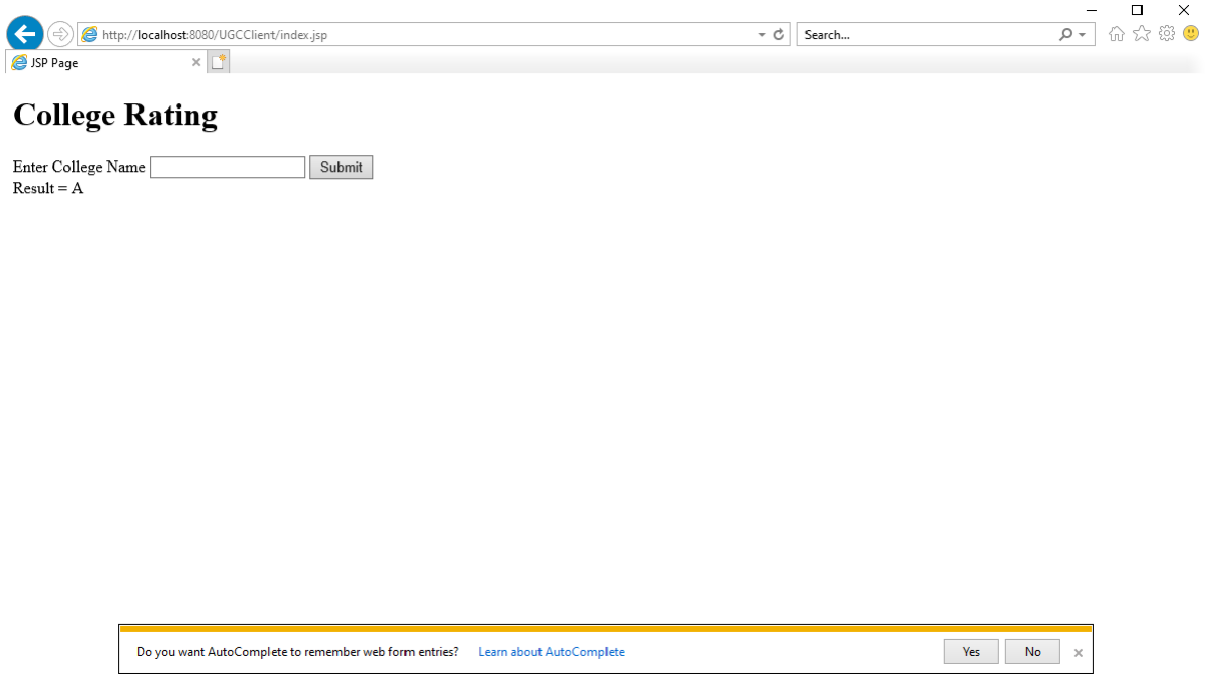

getRating Method invocation
Method parameter(s)
Type
Value
java.lang.String
RUIA
Method returned
java.lang.String : "B"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getRating xmlns:ns2="http://us/">
<collegeName>RUIA</collegeName>
</ns2:getRating>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getRatingResponse xmlns:ns2="http://us/">
<return>B</return>
</ns2:getRatingResponse>
</S:Body>
</S:Envelope>
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

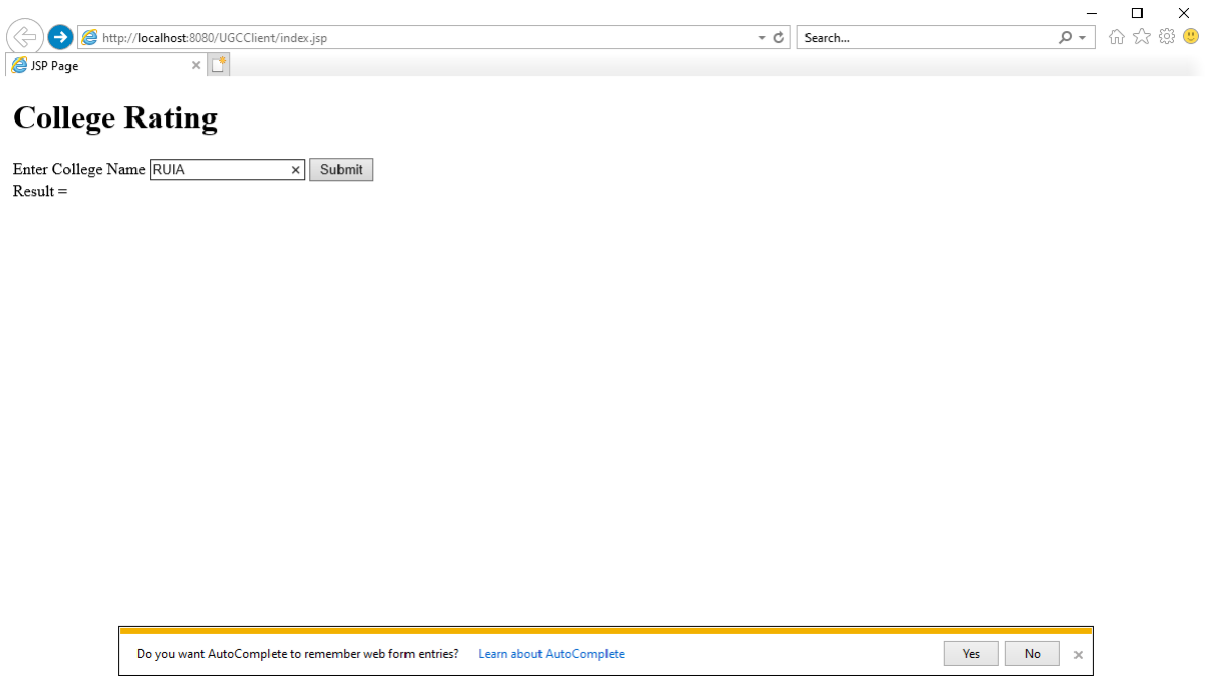

UGCService Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String us.UGCService.getRating(java.lang.String)
getRating
VES
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

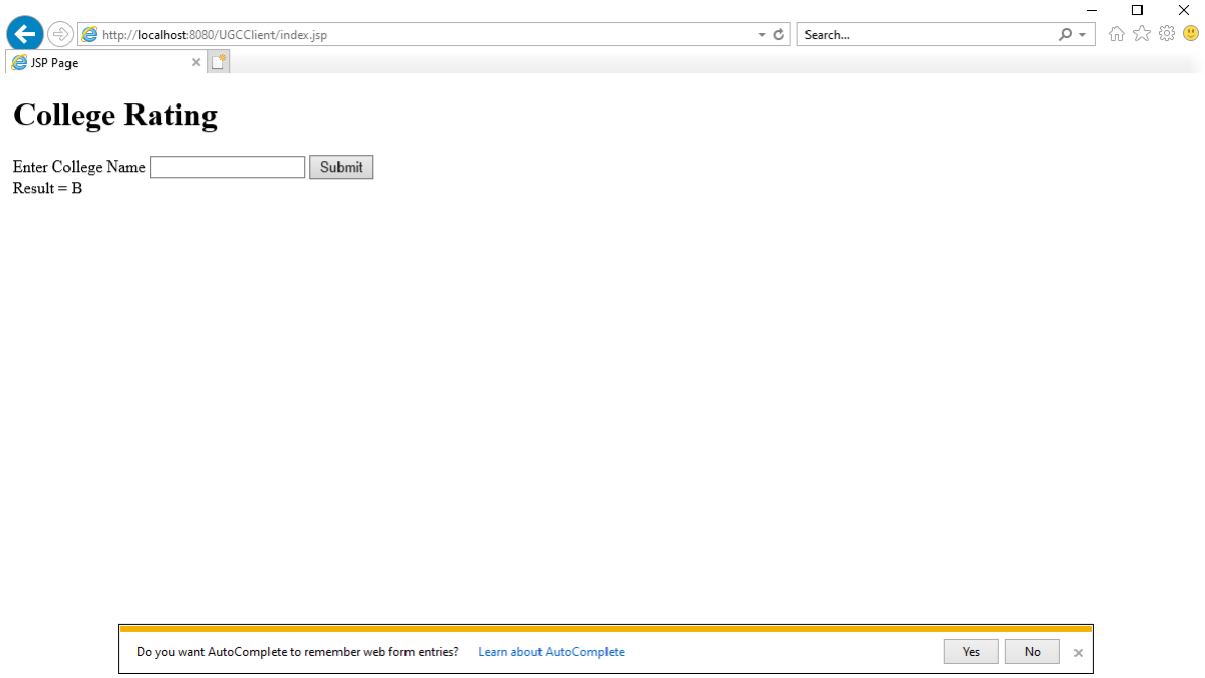

getRating Method invocation
Method parameter(s)
Type
Value
java.lang.String
VES
Method returned
java.lang.String : "C"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getRating xmlns:ns2="http://us/">
<uulleyeName>VES</uulleyeName>
</ns2:getRating>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getRatingResponse xmlns:ns2="http://us/">
<return>C</return>
</ns2:getRatingResponse>
</S:Body>
</S:Envelope>
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

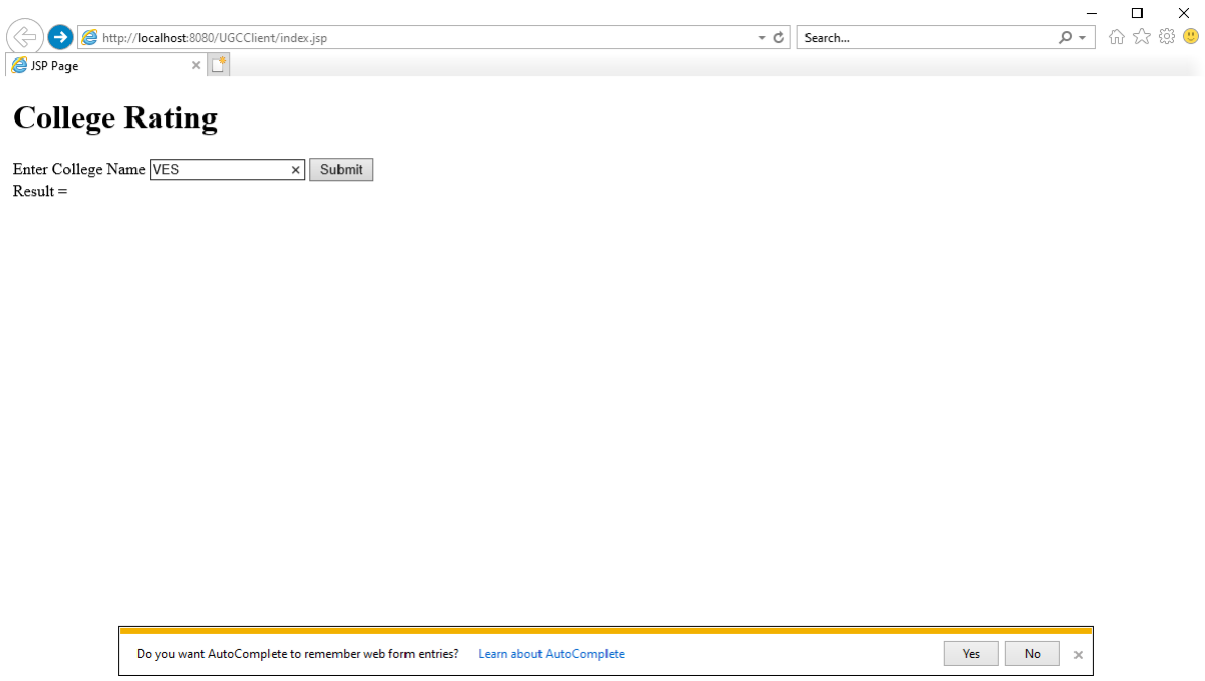

http://localhost:8080/UGCClient/index.jsp
Search...
JSP Page
College Rating
Enter College Name
Submit
Result =
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

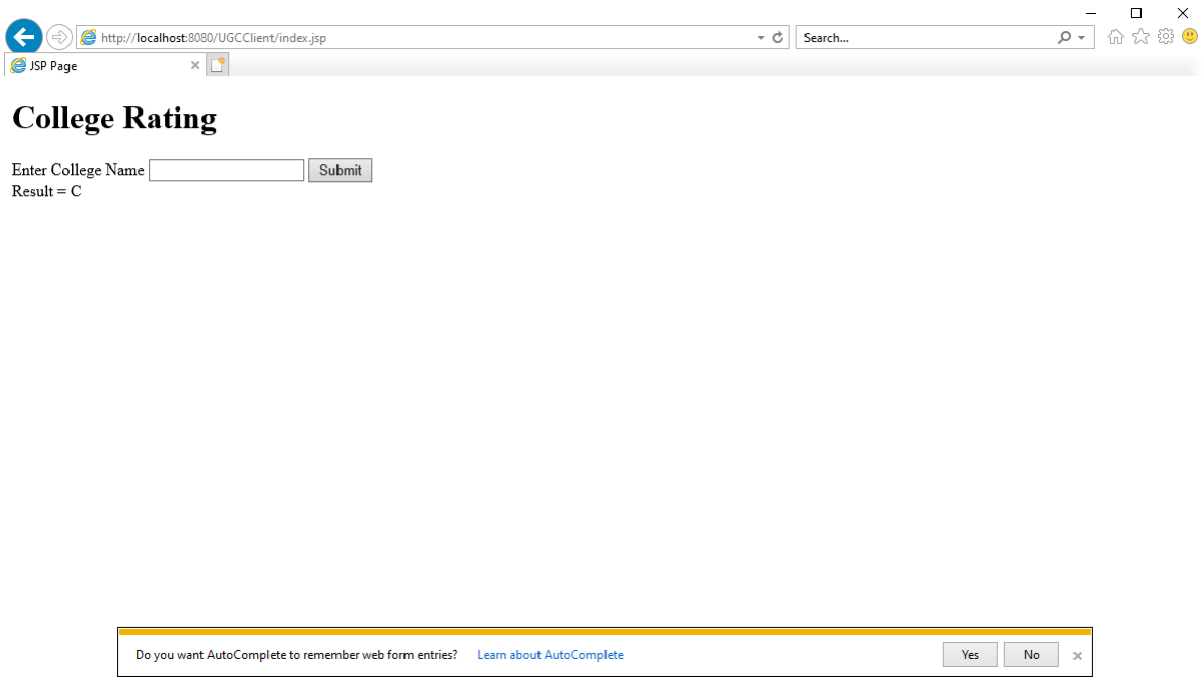

http://localhost:8080/UGCClient/index.jsp
Search...
JSP Page
College Rating
Enter College Name
SIES
Submit
Result =
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/UGCClient/index.jsp
Search...
JSP Page
College Rating
Enter College Name
Submit
Result = A
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/UGCClient/index.jsp
Search...
JSP Page
College Rating
Enter College Name
RUIA
Submit
Result =
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/UGCClient/index.jsp
Search...
JSP Page
College Rating
Enter College Name
Submit
Result = B
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/UGCClient/index.jsp
Search...
JSP Page
College Rating
Enter College Name
VES
Submit
Result =
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/UGCClient/index.jsp
Search...
JSP Page
College Rating
Enter College Name
Submit
Result = C
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

WEB SERVICES

PRACTICAL NO. 5

TCS2223013

CHIRAG BHATIA

Aim:

Design a web service for a channel containing 2 functions – 1st function called getBreakingNews which accepts date as string parameter and returns special news of that day, 2nd function called getPrediction accepts sun sign name as string parameter and returns predictions as string. Design a client to test the above web service.

Code:

Server:

```
package ps;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;
import java.sql.*;

@WebService(serviceName = "Practical5Service")
public class Practical5Service {
```

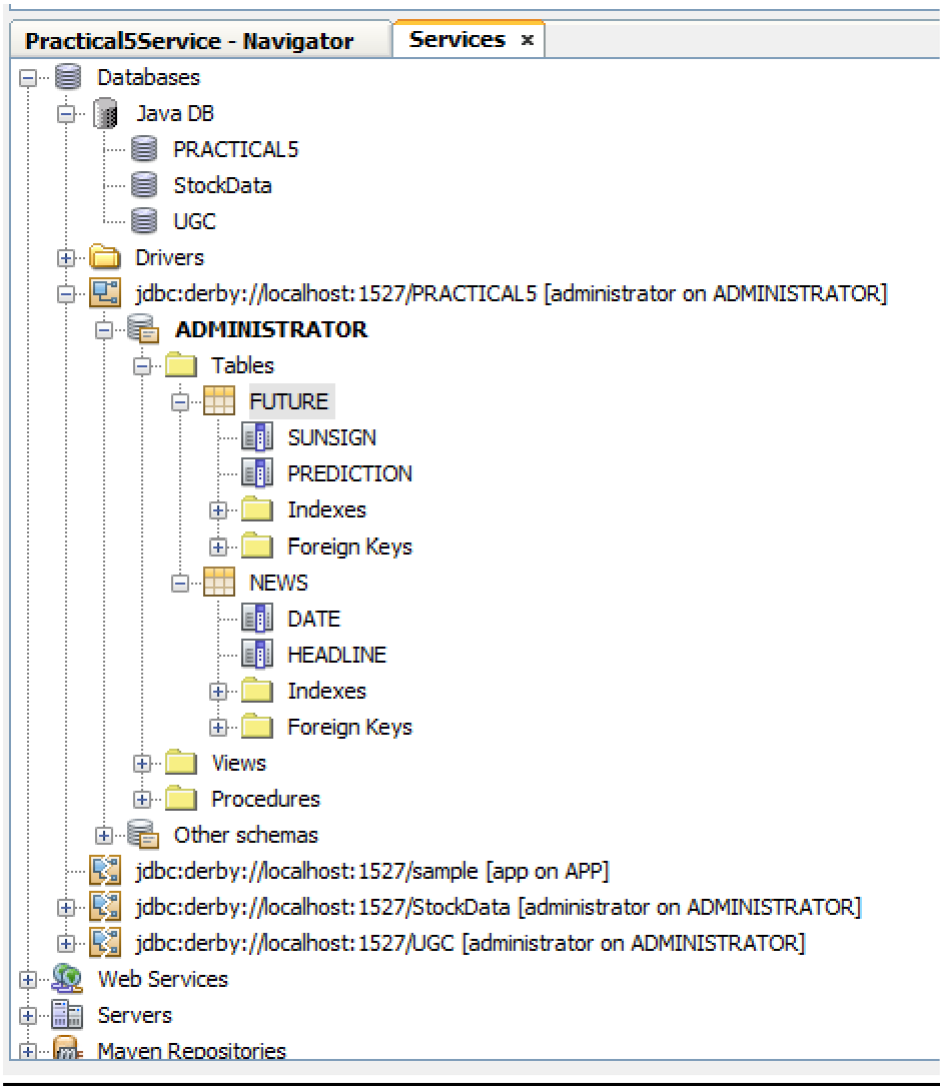
```
    @WebMethod(operationName = "getBreakingNews")
    public String getBreakingNews(@WebParam(name = "date") String date) {
        String news = "";
        try    {
            Class.forName("org.apache.derby.jdbc.ClientDriver");
            Connection con =
DriverManager.getConnection("jdbc:derby://localhost:1527/Practical5",
"administrator", "administrator");
            PreparedStatement pstmt = con.prepareStatement("select * from
News where date=?");
            pstmt.setString(1, date);
            ResultSet rs = pstmt.executeQuery();
            rs.next();
            news = rs.getString("HEADLINE");
        } catch (Exception e) {
            e.printStackTrace();
        }
        return news;
    }

    @WebMethod(operationName = "getPrediction")
    public String getPrediction(@WebParam(name = "sunsign") String sunsign)
{
        String prediction = "";
        try    {
            Class.forName("org.apache.derby.jdbc.ClientDriver");
            Connection con =
DriverManager.getConnection("jdbc:derby://localhost:1527/Practical5",
"administrator", "administrator");
            PreparedStatement pstmt = con.prepareStatement("select * from
Future where SUNSIGN=?");
            pstmt.setString(1, sunsign);
            ResultSet rs = pstmt.executeQuery();
            rs.next();
            prediction = rs.getString("PREDICTION");
        } catch (Exception e) {
            e.printStackTrace();
        }
        return prediction;
    }
}
```

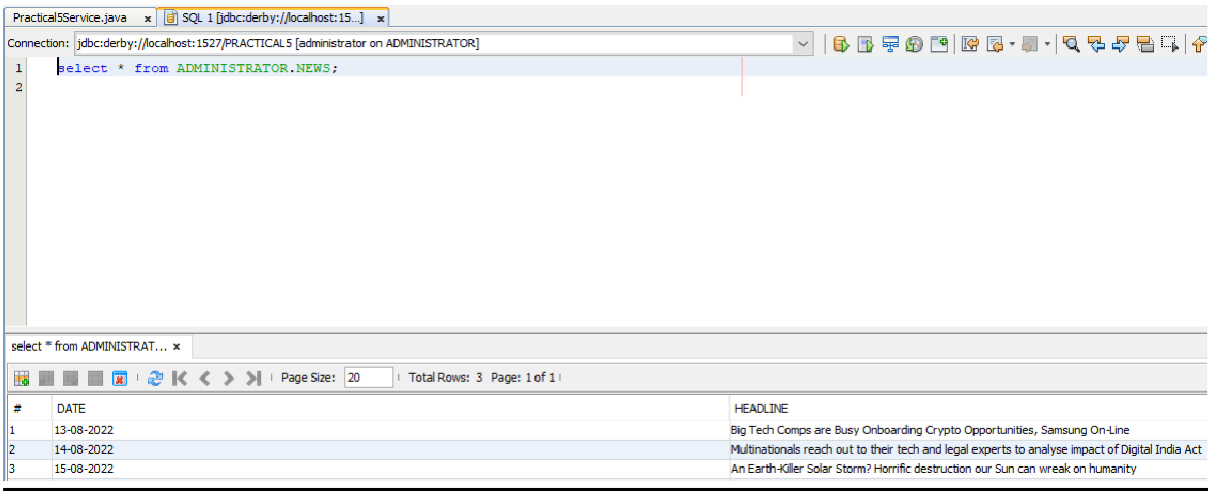
Client:

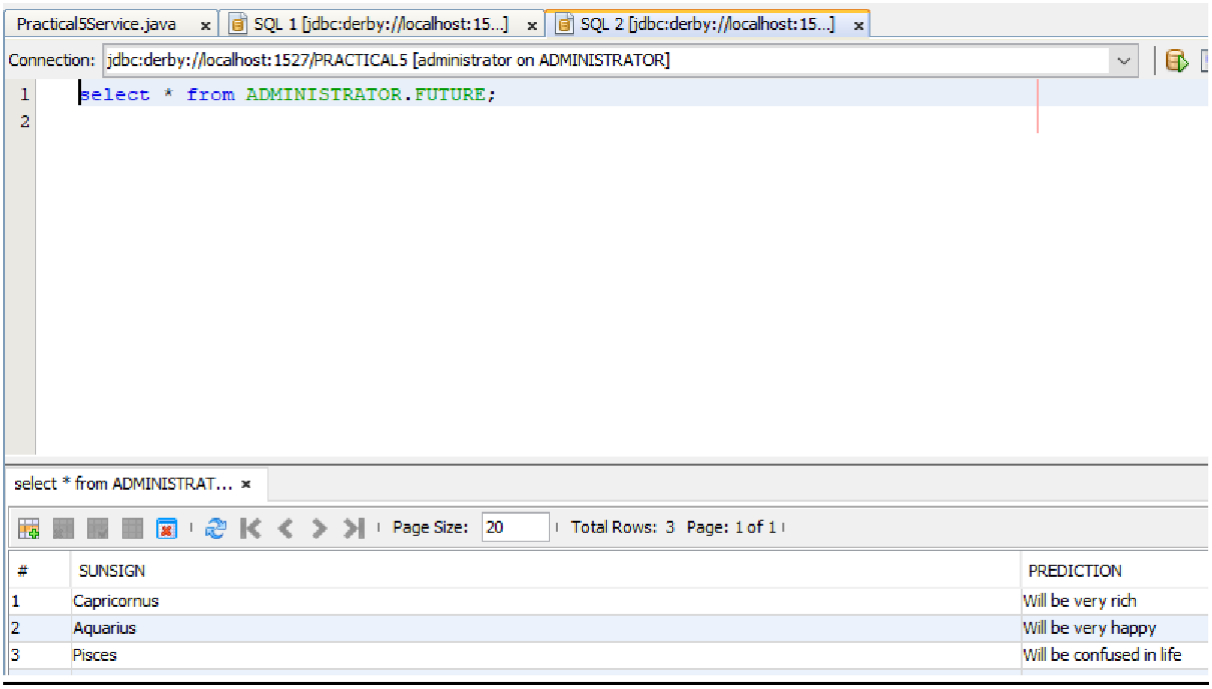
```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>PRACTICAL 5</title>
    </head>
```

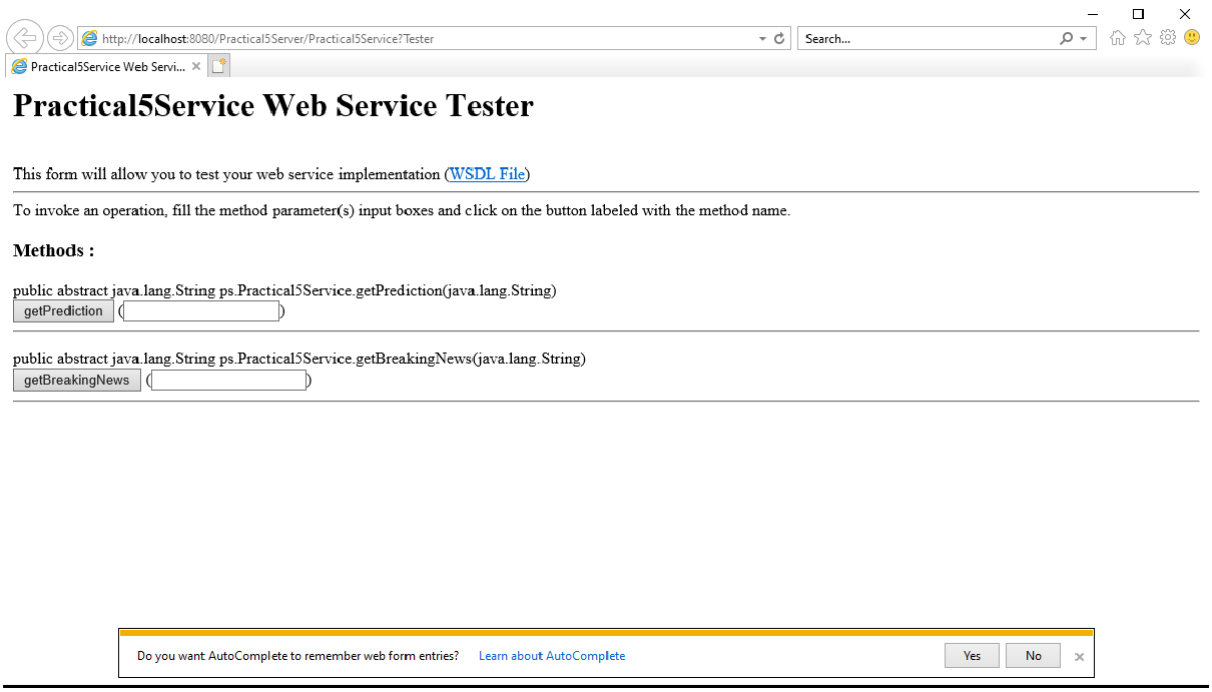
```
    <body>
        <h1>NEWS & PREDICTION</h1>
        <form action="index.jsp" method="POST">
            Enter Date <input type="text" name="Value1"><br>
            Enter Sun Sign <input type="text" name="Value2"><br>
            <input type="submit" name="btnsubmit" value="Submit"/>
        </form>
        <hr/>
        <%
            try {
                ps.Practical5Service_Service service = new
ps.Practical5Service_Service();
                ps.Practical5Service port =
service.getPractical5ServicePort();
                java.lang.String date = request.getParameter("Value1");
                java.lang.String result = port.getBreakingNews(date);
                out.println("Result = " + result);
            } catch (Exception ex) {
            }
        %>
        <hr/>
        <hr/>
        <%
            try {
                ps.Practical5Service_Service service = new
ps.Practical5Service_Service();
                ps.Practical5Service port =
service.getPractical5ServicePort();
                java.lang.String sunsign = request.getParameter("Value2");
                java.lang.String result = port.getPrediction(sunsign);
                out.println("Result = " + result);
            } catch (Exception ex) {
            }
        %>
        <hr/>
    </body>
</html>
```

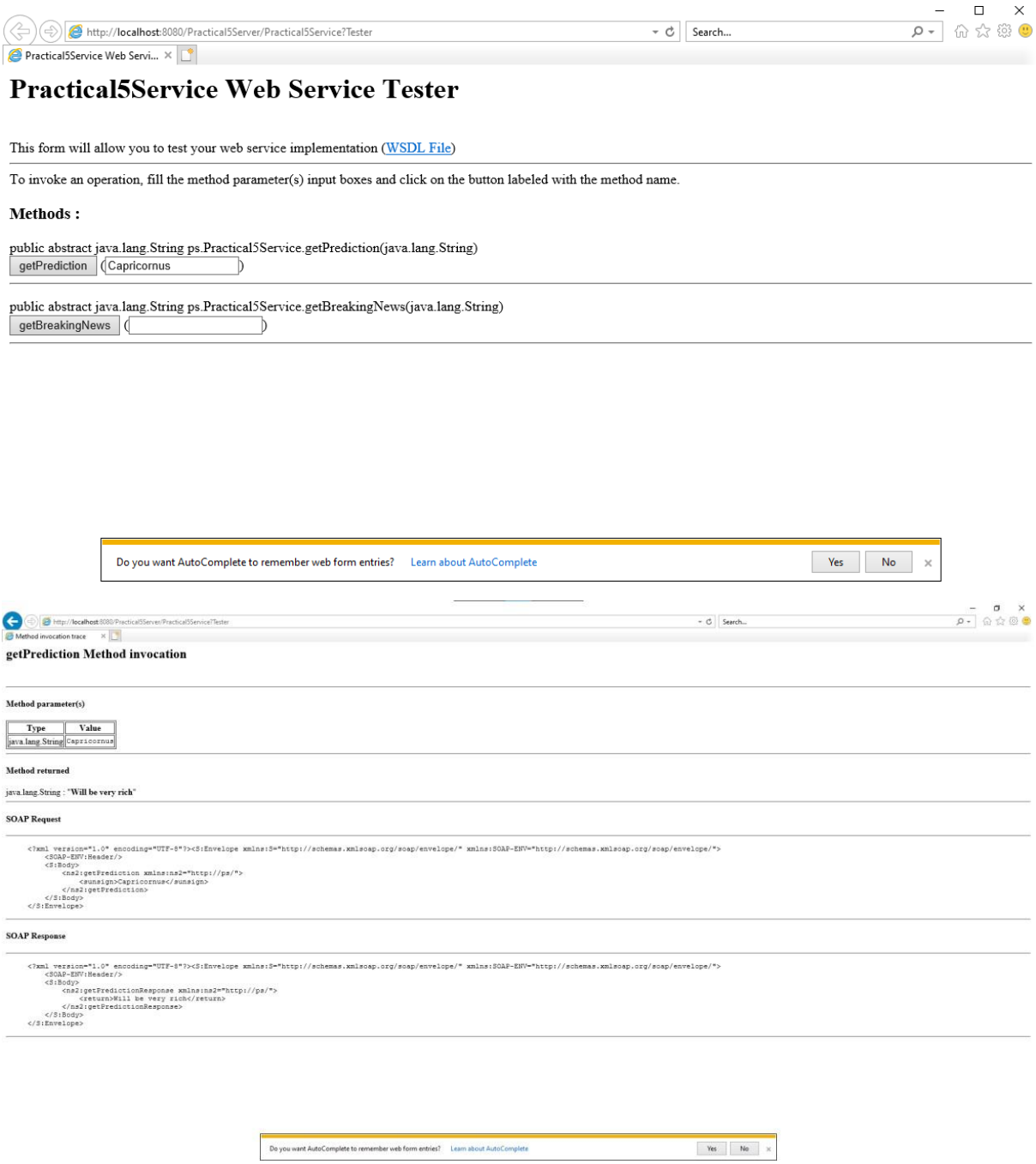
Output:

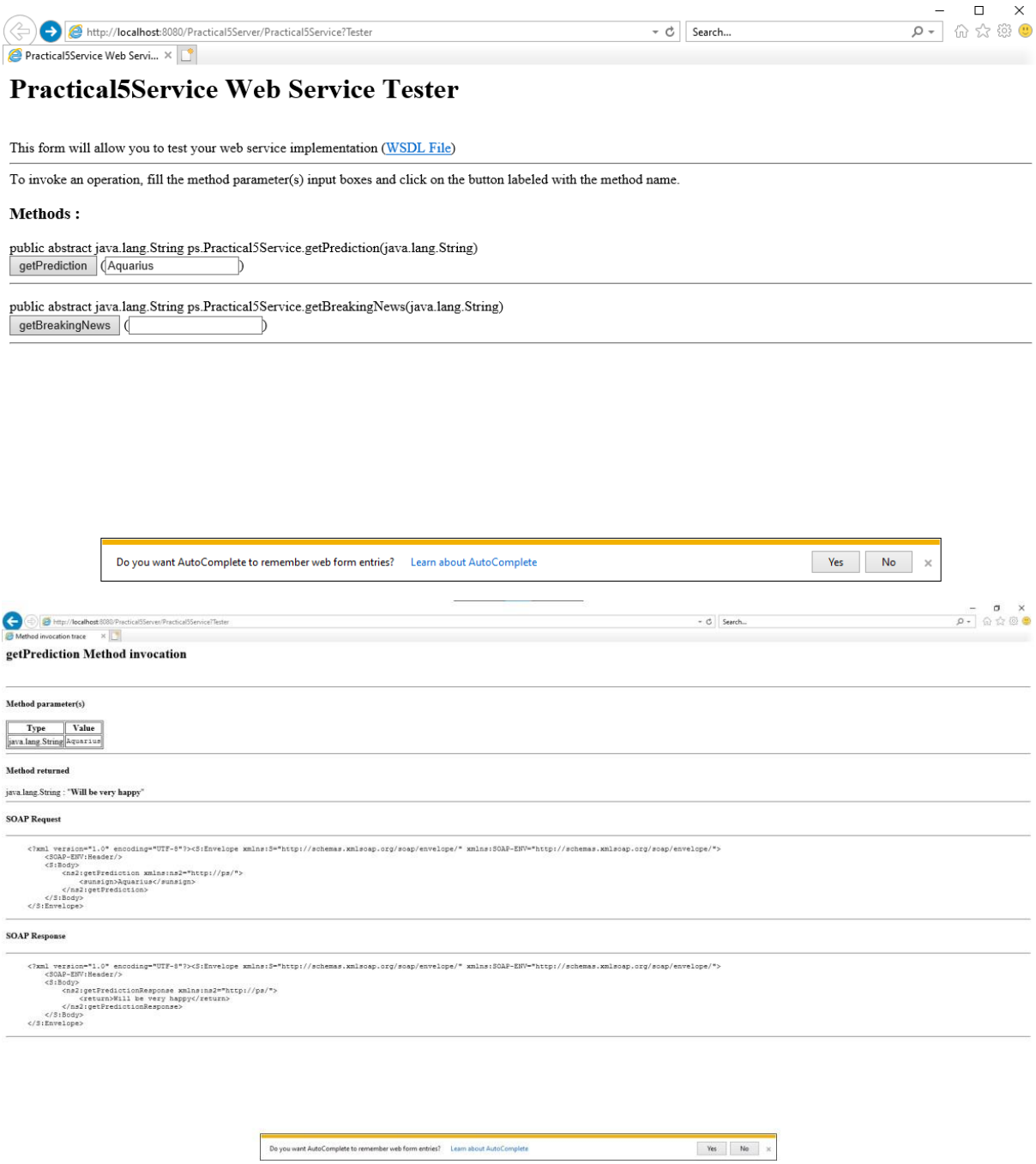

Practical5Service - Navigator
Services
Databases
Java DB
PRACTICAL5
StockData
UGC
Drivers
jdbc:derby://localhost:1527/PRACTICAL5 [administrator on ADMINISTRATOR]
ADMINISTRATOR
Tables
FUTURE
SUNSIGN
PREDICTION
Indexes
Foreign Keys
NEWS
DATE
HEADLINE
Indexes
Foreign Keys
Views
Procedures
Other schemas
jdbc:derby://localhost:1527/sample [app on APP]
jdbc:derby://localhost:1527/StockData [administrator on ADMINISTRATOR]
jdbc:derby://localhost:1527/UGC [administrator on ADMINISTRATOR]
Web Services
Servers
Maven Repositories


Practical5Service.java
SQL 1 [jdbc:derby://localhost:15...]
Connection: jdbc:derby://localhost:1527/PRACTICAL5 [administrator on ADMINISTRATOR]
1 select * from ADMINISTRATOR.NEWS;
2
select * from ADMINISTRAT...
Page Size: 20
Total Rows: 3 Page: 1 of 1
DATE HEADLINE
1 13-08-2022 Big Tech Comps are Busy Onboarding Crypto Opportunities, Samsung On-Line
2 14-08-2022 Multinationals reach out to their tech and legal experts to analyse impact of Digital India Act
3 15-08-2022 An Earth-Killer Solar Storm? Horrific destruction our Sun can wreak on humanity


Practical5Service.java
SQL 1 [jdbc:derby://localhost:15...]
SQL 2 [jdbc:derby://localhost:15...]
Connection: jdbc:derby://localhost:1527/PRACTICAL5 [administrator on ADMINISTRATOR]
1 select * from ADMINISTRATOR.FUTURE;
2
select * from ADMINISTRAT...
Page Size: 20
Total Rows: 3 Page: 1 of 1
SUNSIGN PREDICTION
1 Capricornus Will be very rich
2 Aquarius Will be very happy
3 Pisces Will be confused in life

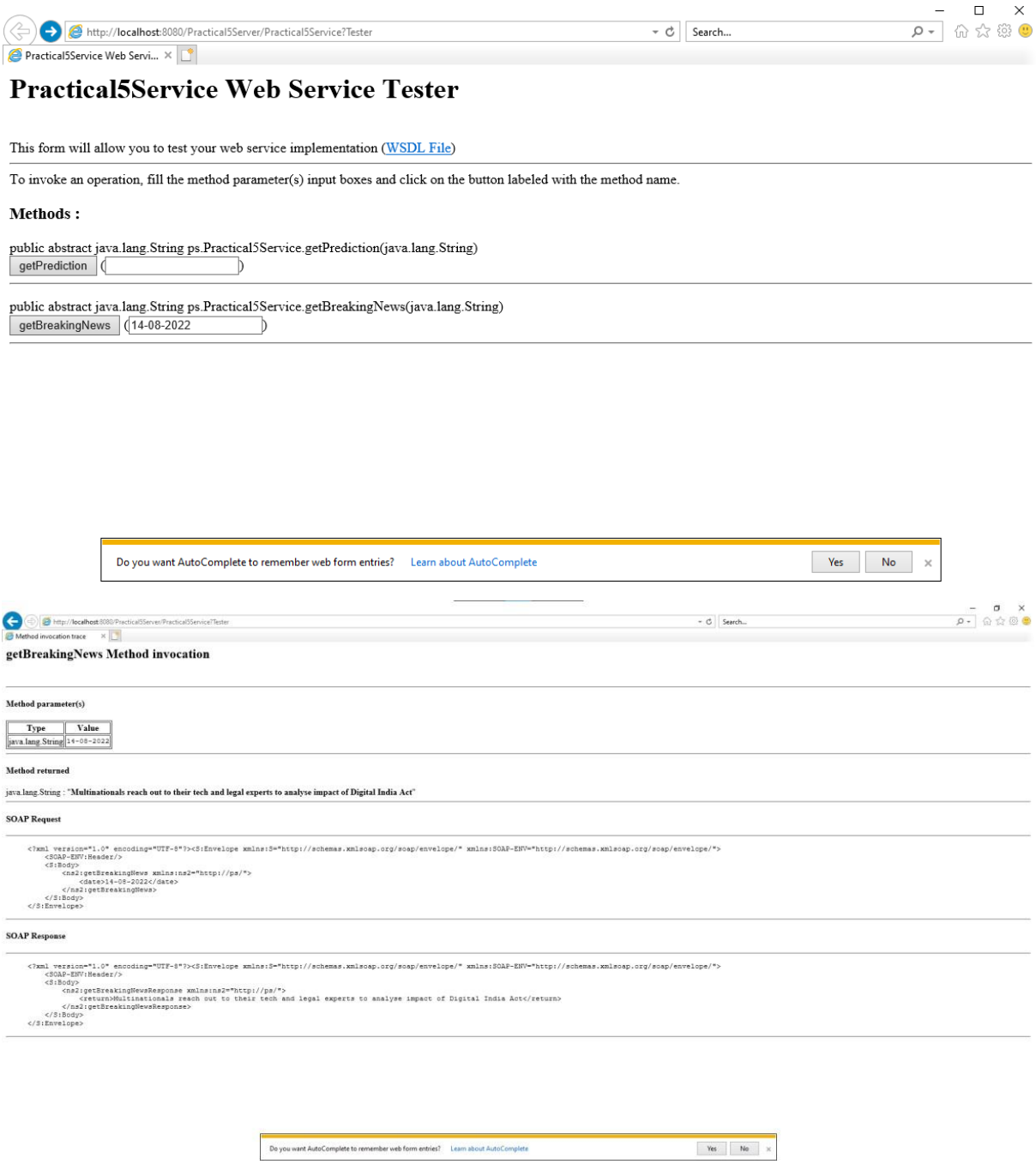

http://localhost:8080/Practical5Server/Practical5Service?Tester
Search...
Practical5Service Web Servi...
Practical5Service Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String ps.Practical5Service.getPrediction(java.lang.String)
getPrediction
public abstract java.lang.String ps.Practical5Service.getBreakingNews(java.lang.String)
getBreakingNews
Do you want AutoComplete to remember web form entries? Learn about AutoComplete
Yes
No


http://localhost:8080/Practical5Server/Practical5Service?Tester
Search...
Practical5Service Web Servi...
Practical5Service Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String ps.Practical5Service.getPrediction(java.lang.String)
getPrediction
Capricornus
public abstract java.lang.String ps.Practical5Service.getBreakingNews(java.lang.String)
getBreakingNews
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No
Method invocation trace
getPrediction Method invocation
Method parameter(s)
Type
Value
java.lang.String
Capricornus
Method returned
java.lang.String : "Will be very rich"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getPrediction xmlns:ns2="http://ps/">
<sunsign>Capricornus</sunsign>
</ns2:getPrediction>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getPredictionResponse xmlns:ns2="http://ps/">
<return>Will be very rich</return>
</ns2:getPredictionResponse>
</S:Body>
</S:Envelope>

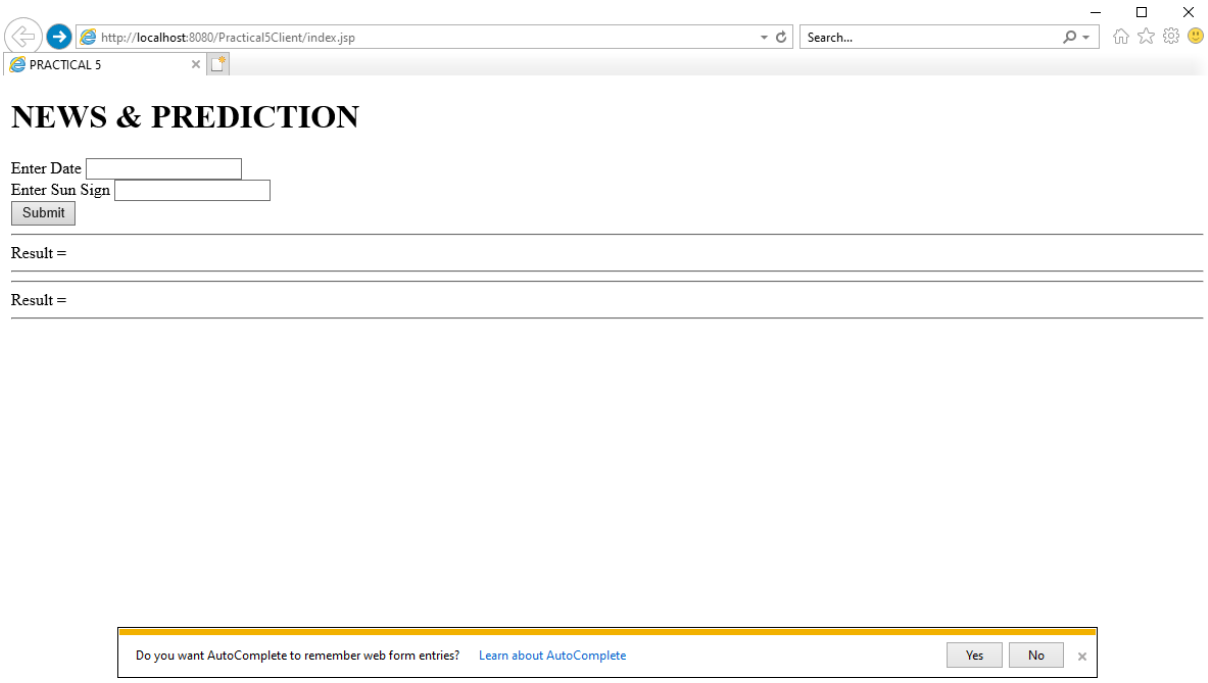

http://localhost:8080/Practical5Server/Practical5Service?Tester
Search...
Practical5Service Web Servi...
Practical5Service Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String ps.Practical5Service.getPrediction(java.lang.String)
getPrediction (Aquarius)
public abstract java.lang.String ps.Practical5Service.getBreakingNews(java.lang.String)
getBreakingNews ()
Do you want AutoComplete to remember web form entries? Learn about AutoComplete
Yes
No
Method invocation trace
getPrediction Method invocation
Method parameter(s)
Type
Value
java.lang.String
Aquarius
Method returned
java.lang.String : "Will be very happy"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getPrediction xmlns:ns2="http://ps/">
<sunsign>Aquarius</sunsign>
</ns2:getPrediction>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getPredictionResponse xmlns:ns2="http://ps/">
<return>Will be very happy</return>
</ns2:getPredictionResponse>
</S:Body>
</S:Envelope>
Do you want AutoComplete to remember web form entries? Learn about AutoComplete
Yes
No

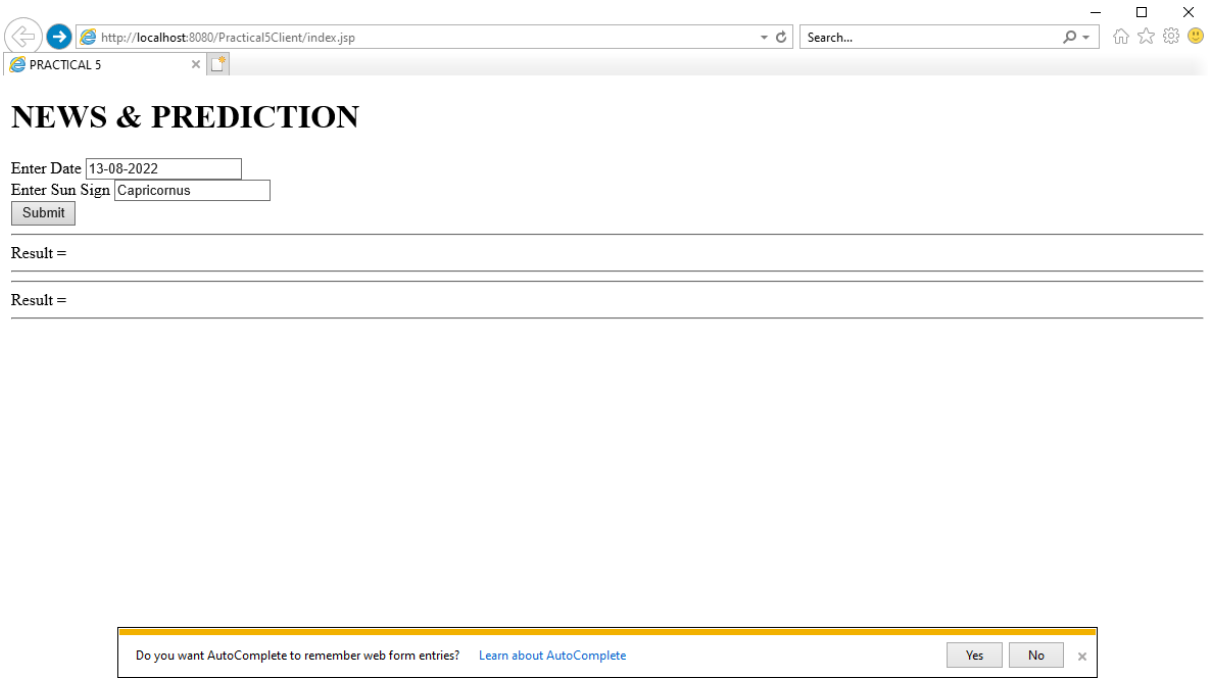

http://localhost:8080/Practical5Server/Practical5Service?Tester
Search...
Practical5Service Web Servi...
Practical5Service Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String ps.Practical5Service.getPrediction(java.lang.String)
getPrediction (Pisces)
public abstract java.lang.String ps.Practical5Service.getBreakingNews(java.lang.String)
getBreakingNews ( )
Do you want AutoComplete to remember web form entries? Learn about AutoComplete
Yes
No
Method invocation trace
getPrediction Method invocation
Method parameter(s)
Type
Value
java.lang.String
Pisces
Method returned
java.lang.String : "Will be confused in life"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getPrediction xmlns:ns2="http://ps/">
<sunsign>Pisces</sunsign>
</ns2:getPrediction>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getPredictionResponse xmlns:ns2="http://ps/">
<return>Will be confused in life</return>
</ns2:getPredictionResponse>
</S:Body>
</S:Envelope>
Do you want AutoComplete to remember web form entries? Learn about AutoComplete
Yes
No

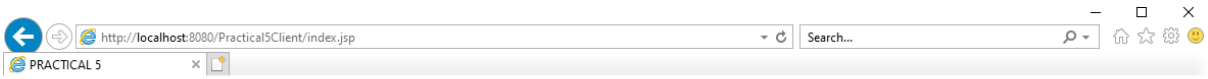

http://localhost:8080/Practical5Server/Practical5Service?Tester
Search...
Practical5Service Web Servi...
Practical5Service Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String ps.Practical5Service.getPrediction(java.lang.String)
getPrediction
public abstract java.lang.String ps.Practical5Service.getBreakingNews(java.lang.String)
getBreakingNews
13-08-2022
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No
Method invocation trace
getBreakingNews Method invocation
Method parameter(s)
Type
Value
java.lang.String
13-08-2022
Method returned
java.lang.String : "Big Tech Comps are Busy Onboarding Crypto Opportunities, Samsung On-Line"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getBreakingNews xmlns:ns2="http://ps/">
<date>13-08-2022</date>
</ns2:getBreakingNews>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getBreakingNewsResponse xmlns:ns2="http://ps/">
<return>Big Tech Comps are Busy Onboarding Crypto Opportunities, Samsung On-Line</return>
</ns2:getBreakingNewsResponse>
</S:Body>
</S:Envelope>


http://localhost:8080/Practical5Server/Practical5Service?Tester
Search...
Practical5Service Web Servi...
Practical5Service Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String ps.Practical5Service.getPrediction(java.lang.String)
getPrediction
public abstract java.lang.String ps.Practical5Service.getBreakingNews(java.lang.String)
getBreakingNews
14-08-2022
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No
Method invocation trace
getBreakingNews Method invocation
Method parameter(s)
Type
Value
java.lang.String
14-08-2022
Method returned
java.lang.String : "Multinationals reach out to their tech and legal experts to analyse impact of Digital India Act"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getBreakingNews xmlns:ns2="http://ps/">
<date>14-08-2022</date>
</ns2:getBreakingNews>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getBreakingNewsResponse xmlns:ns2="http://ps/">
<return>Multinationals reach out to their tech and legal experts to analyse impact of Digital India Act</return>
</ns2:getBreakingNewsResponse>
</S:Body>
</S:Envelope>


http://localhost:8080/Practical5Server/Practical5Service?Tester
Search...
Practical5Service Web Servi...
Practical5Service Web Service Tester
This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods :
public abstract java.lang.String ps.Practical5Service.getPrediction(java.lang.String)
getPrediction
public abstract java.lang.String ps.Practical5Service.getBreakingNews(java.lang.String)
getBreakingNews
15-08-2022
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No
Method invocation trace
getBreakingNews Method invocation
Method parameter(s)
Type
Value
java.lang.String
15-08-2022
Method returned
java.lang.String : "An Earth-Killer Solar Storm? Horrific destruction our Sun can wreak on humanity"
SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getBreakingNews xmlns:ns2="http://ps/">
<date>15-08-2022</date>
</ns2:getBreakingNews>
</S:Body>
</S:Envelope>
SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
<ns2:getBreakingNewsResponse xmlns:ns2="http://ps/">
<return>An Earth-Killer Solar Storm? Horrific destruction our Sun can wreak on humanity</return>
</ns2:getBreakingNewsResponse>
</S:Body>
</S:Envelope>
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/Practical5Client/index.jsp
Search...
PRACTICAL 5
NEWS & PREDICTION
Enter Date
Enter Sun Sign
Submit
Result =
Result =
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/Practical5Client/index.jsp
Search...
PRACTICAL 5
NEWS & PREDICTION
Enter Date 13-08-2022
Enter Sun Sign Capricornus
Submit
Result =
Result =
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

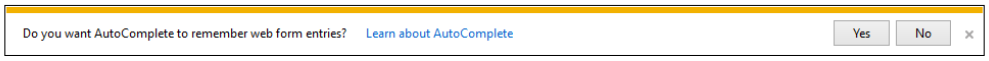


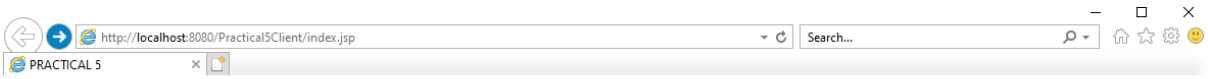
NEWS & PREDICTION

Enter Date
Enter Sun Sign
Submit

Result = Big Tech Comps are Busy Onboarding Crypto Opportunities, Samsung On-Line

Result = Will be very rich



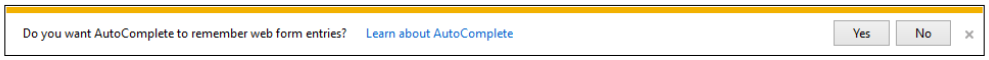


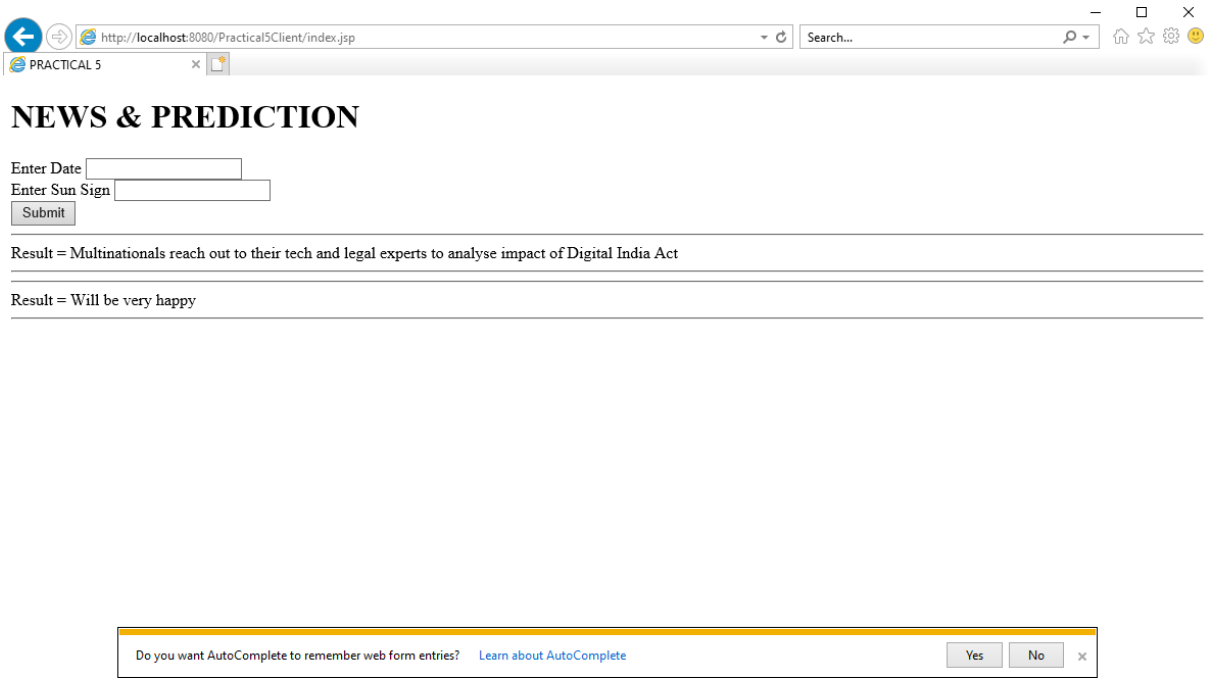
NEWS & PREDICTION

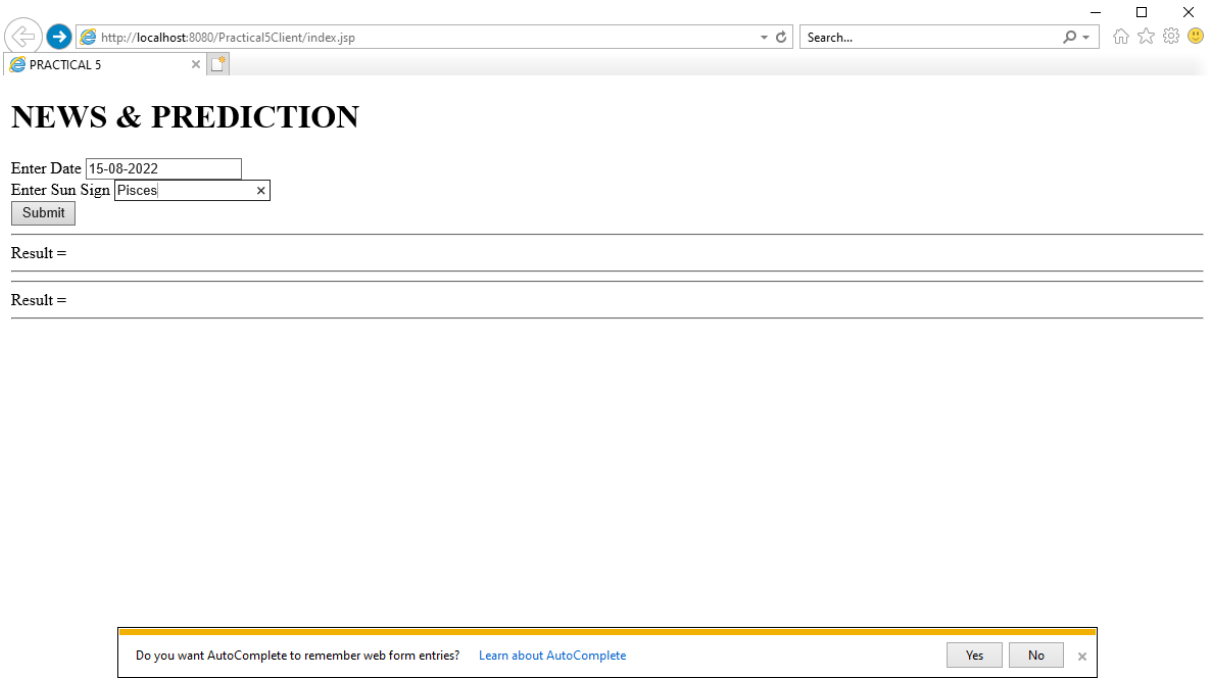
Enter Date 14-08-2022
Enter Sun Sign Aquarius
Submit

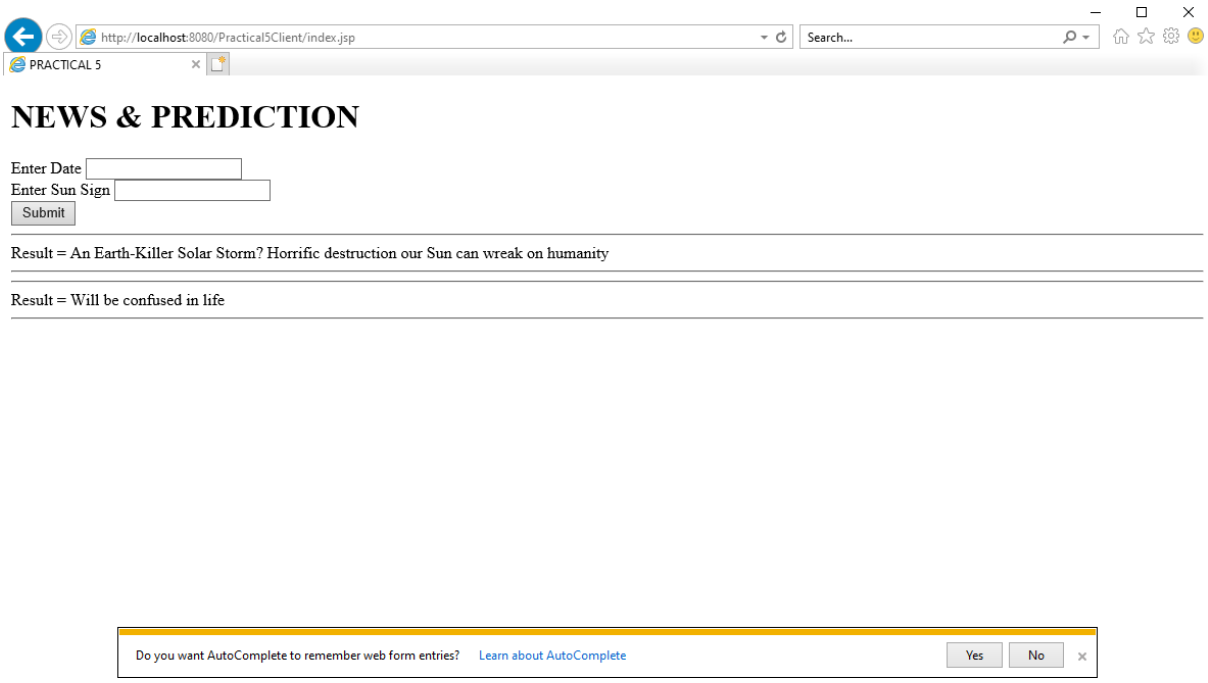
Result =

Result =




http://localhost:8080/Practical5Client/index.jsp
Search...
PRACTICAL 5
NEWS & PREDICTION
Enter Date
Enter Sun Sign
Submit
Result = Multinationals reach out to their tech and legal experts to analyse impact of Digital India Act
Result = Will be very happy
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/Practical5Client/index.jsp
Search...
PRACTICAL 5
NEWS & PREDICTION
Enter Date 15-08-2022
Enter Sun Sign Pisces
Submit
Result =
Result =
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No


http://localhost:8080/Practical5Client/index.jsp
Search...
PRACTICAL 5
NEWS & PREDICTION
Enter Date
Enter Sun Sign
Submit
Result = An Earth-Killer Solar Storm? Horrific destruction our Sun can wreak on humanity
Result = Will be confused in life
Do you want AutoComplete to remember web form entries?
Learn about AutoComplete
Yes
No

WEB SERVICES

PRACTICAL NO. 6

TCS2223013

CHIRAG BHATIA

Aim:

Design a Restful webservice from a database table Employee with columns emp_id, emp_name and emp_designation. Test the webservice for the various http requests

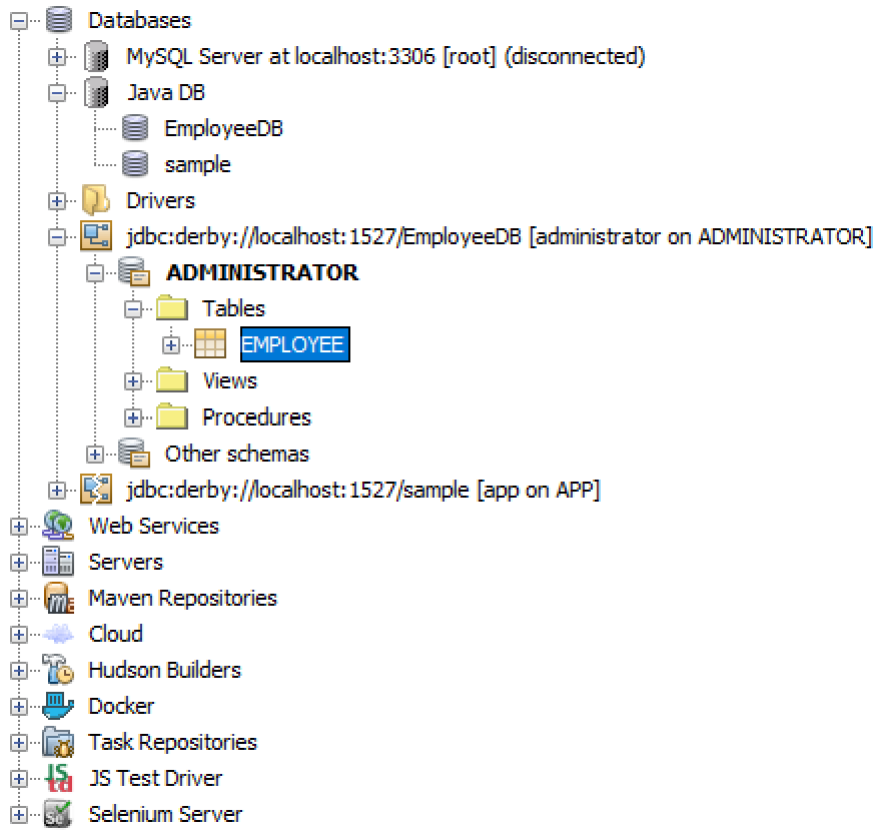
Code:

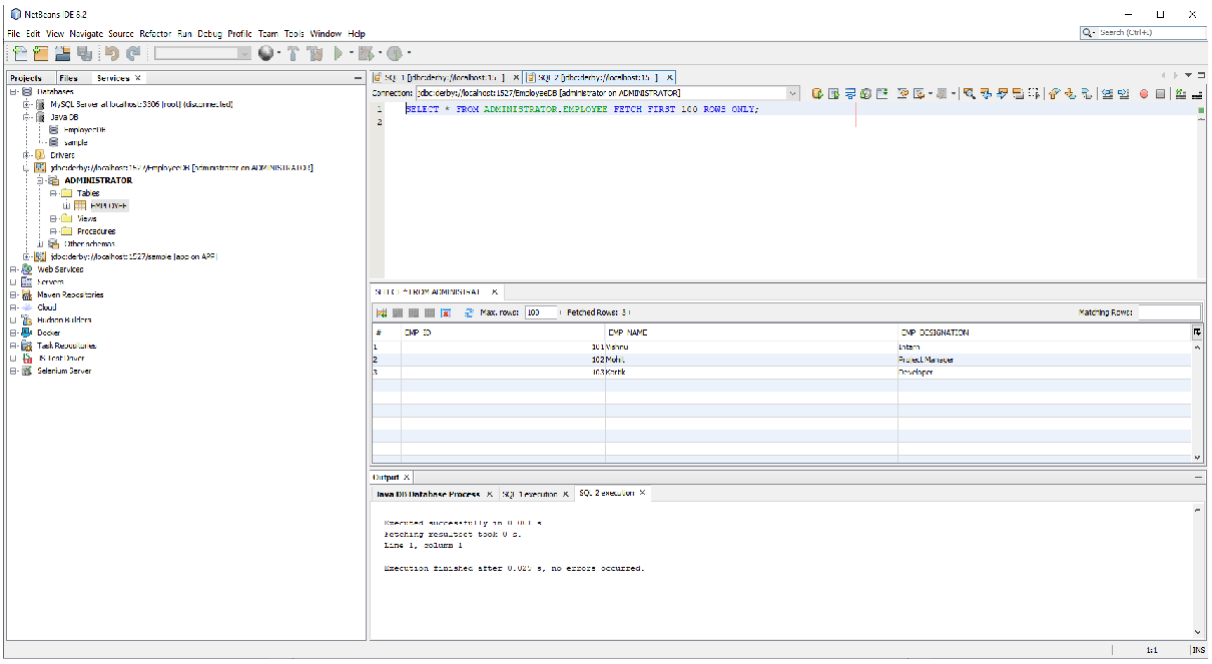
```
INSERT INTO EMPLOYEE VALUES(101,'Vishnu','Intern');
INSERT INTO EMPLOYEE VALUES(102,'Mohit','Project Manager');
INSERT INTO EMPLOYEE VALUES(103,'Kartik','Developer');
```

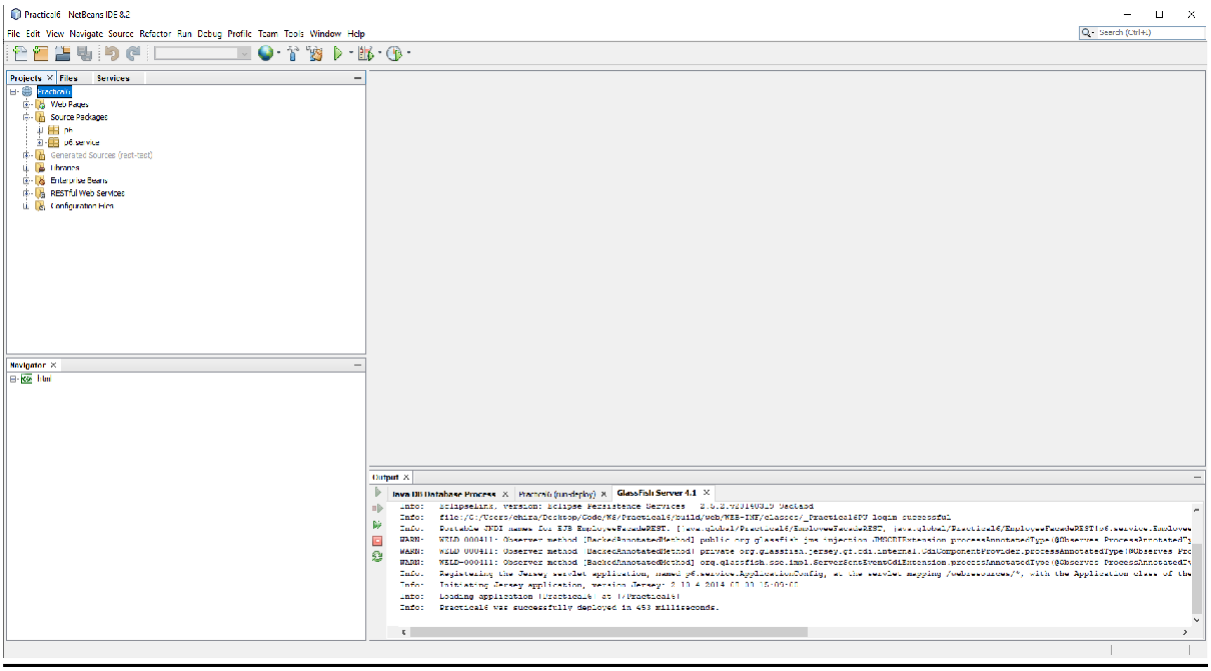
```
<employee>
    <empDesignation>Tester</empDesignation>
    <empId>104</empId>
    <empName>Kevin</empName>
</employee>
```

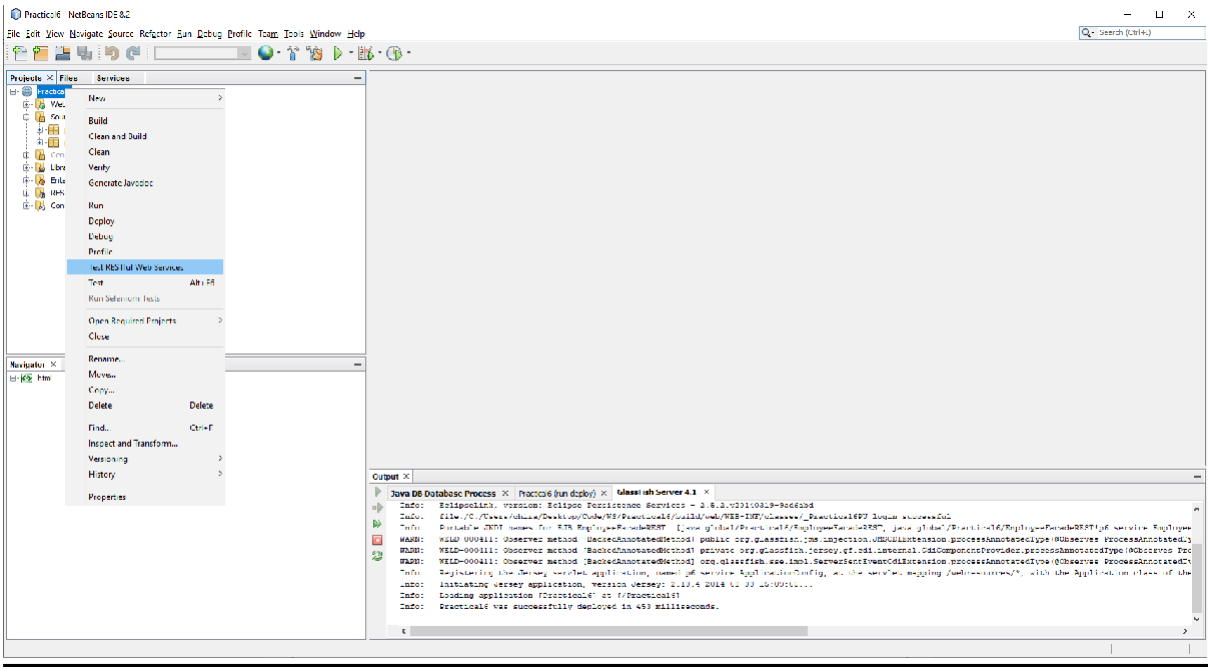
```
{"empDesignation":"UI/UX","empId":105,"empName":"Sachin"}
```

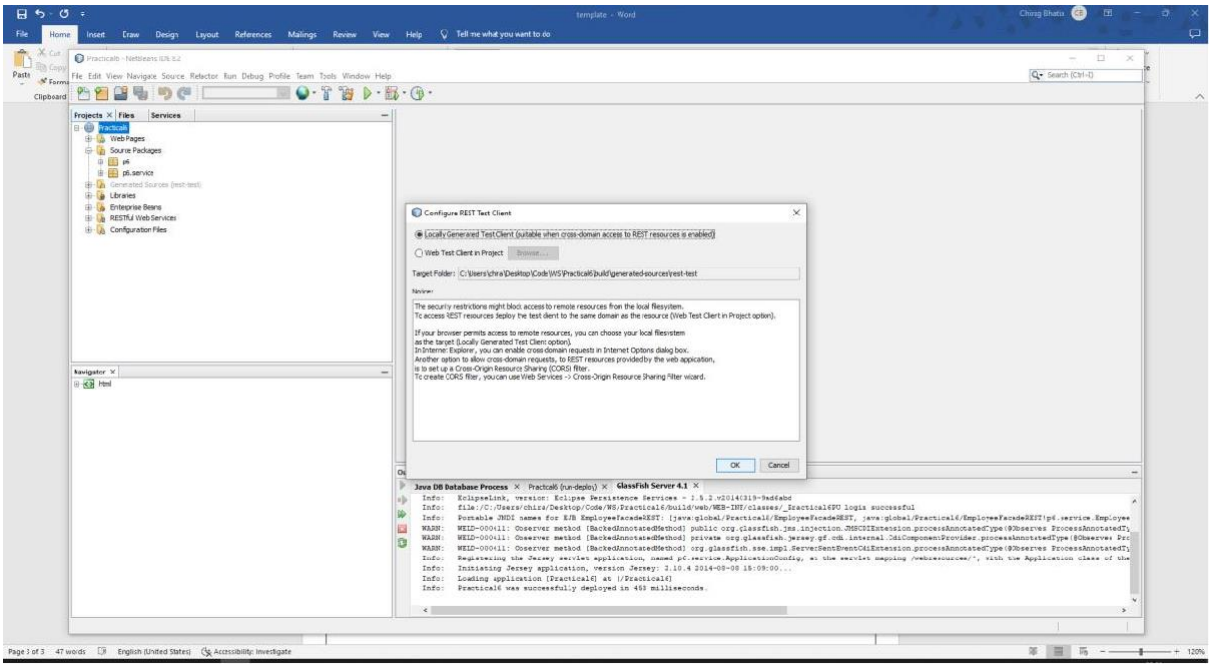
Output:

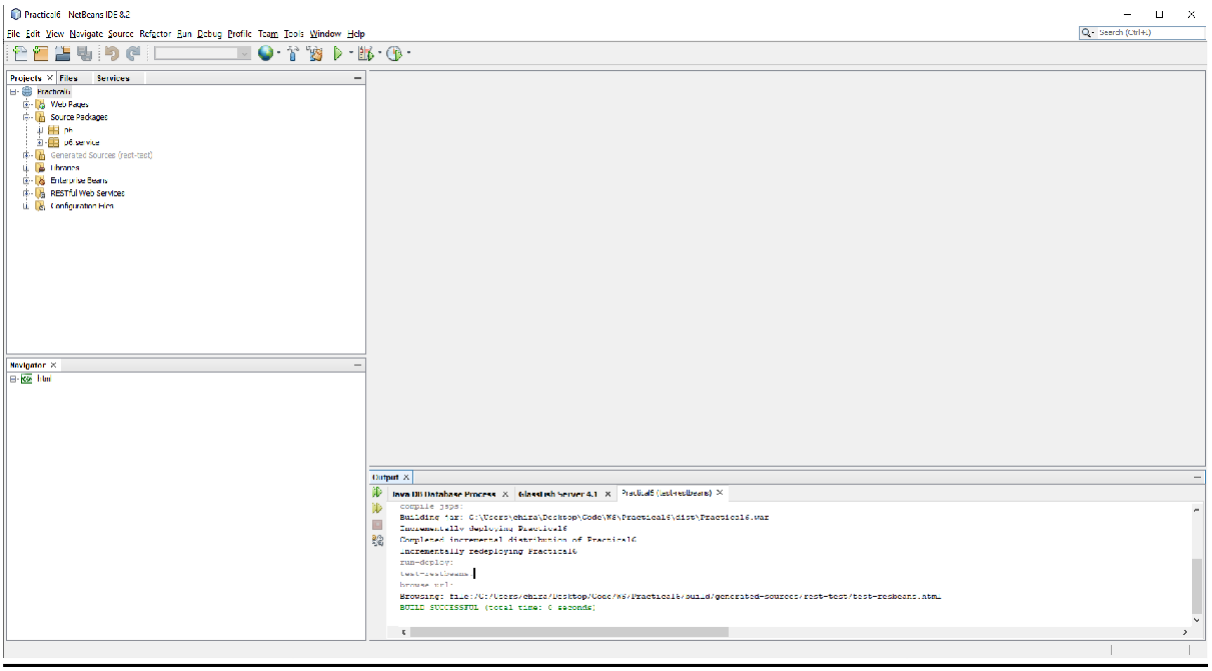

Databases
MySQL Server at localhost:3306 [root] (disconnected)
Java DB
EmployeeDB
sample
Drivers
jdbc:derby://localhost:1527/EmployeeDB [administrator on ADMINISTRATOR]
ADMINISTRATOR
Tables
EMPLOYEE
Views
Procedures
Other schemas
jdbc:derby://localhost:1527/sample [app on APP]
Web Services
Servers
Maven Repositories
Cloud
Hudson Builders
Docker
Task Repositories
JS Test Driver
Selenium Server

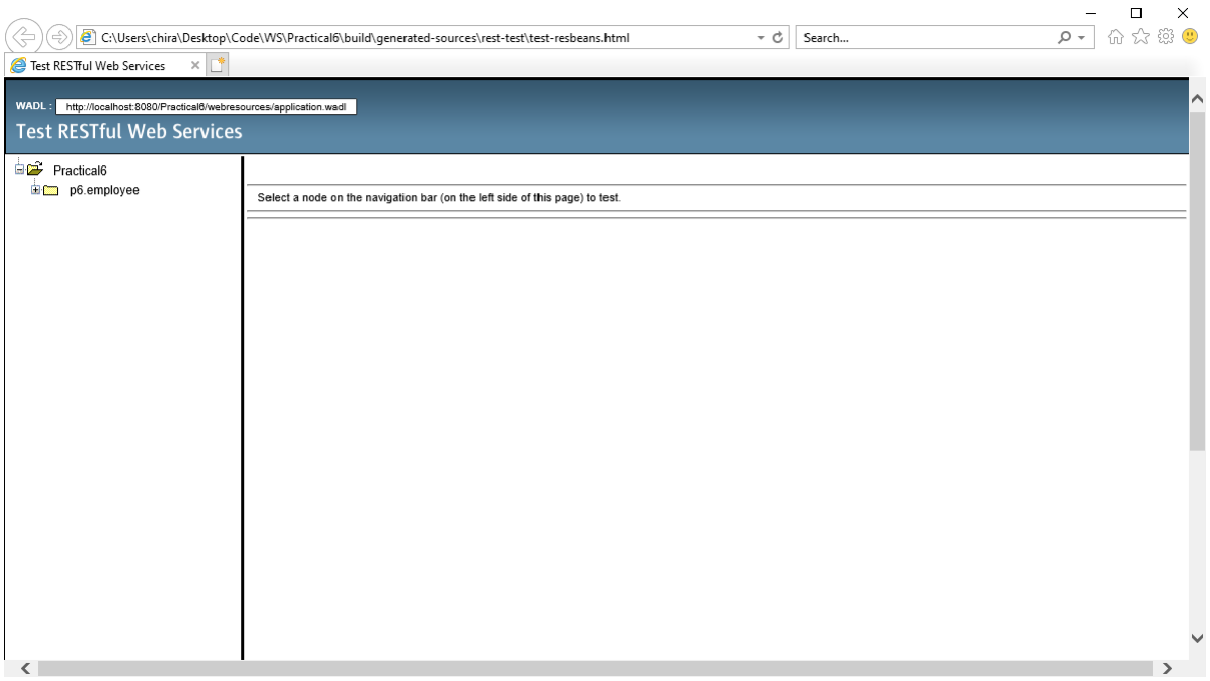


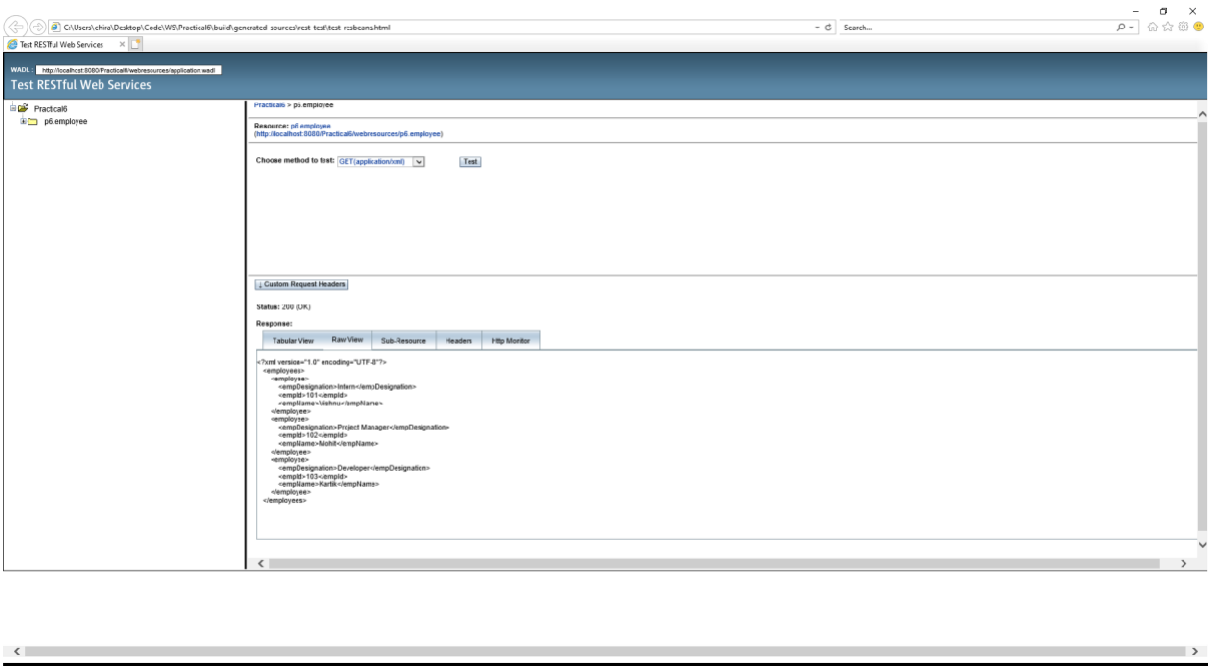


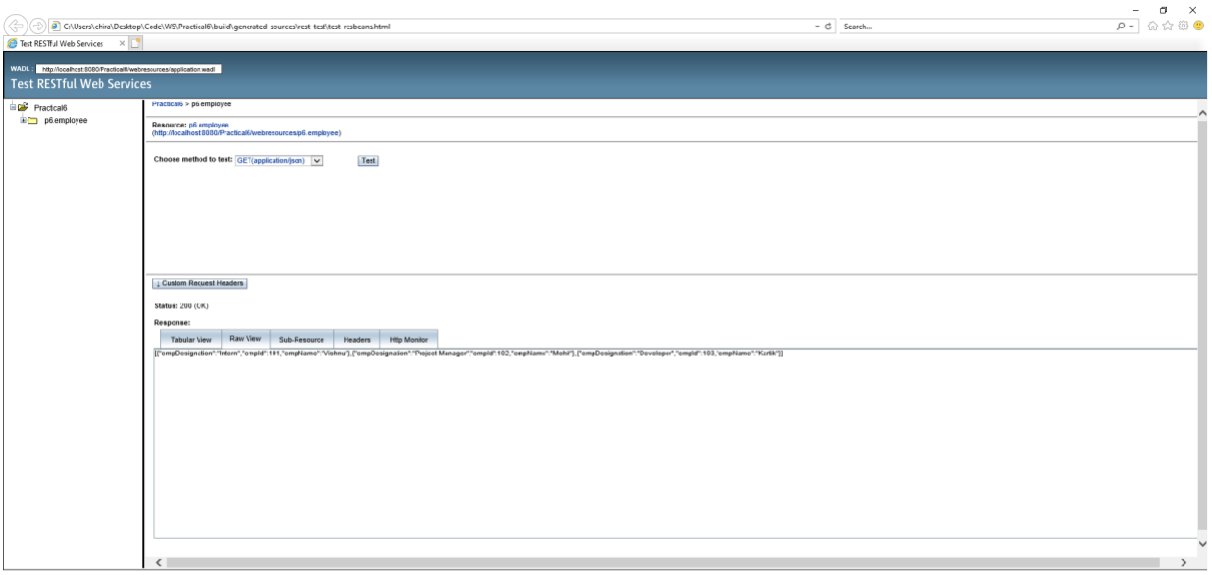


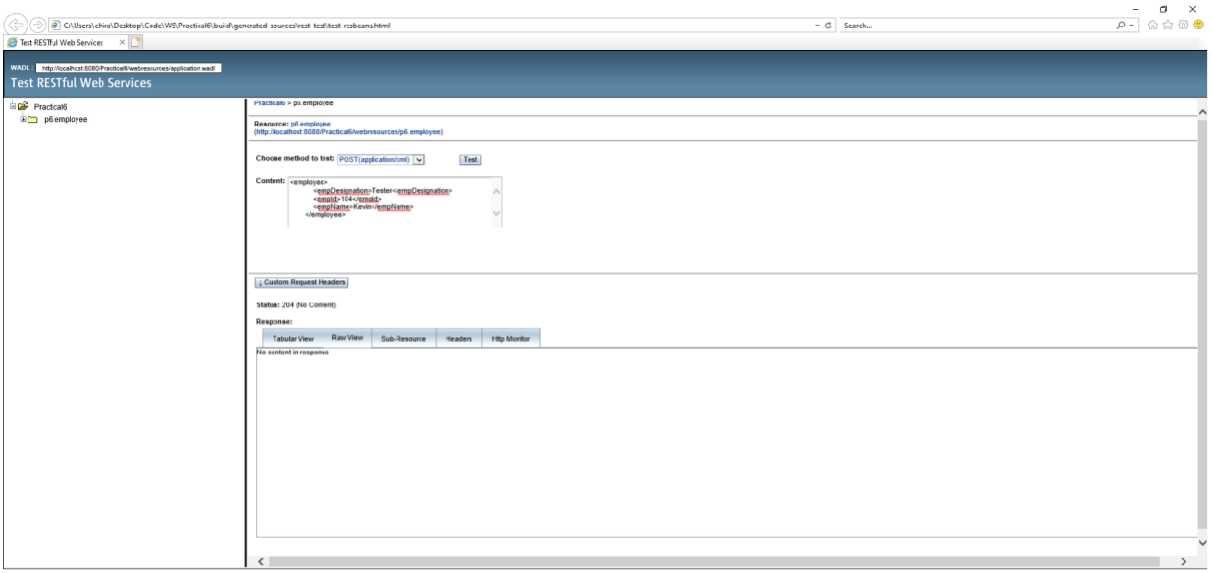


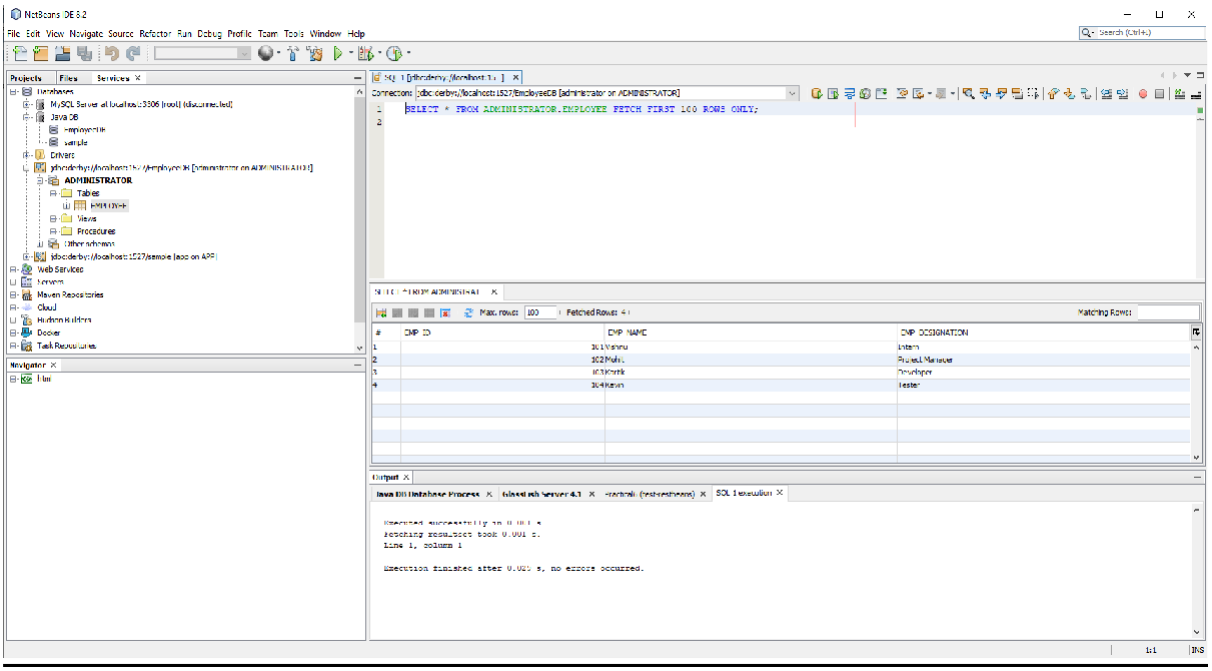


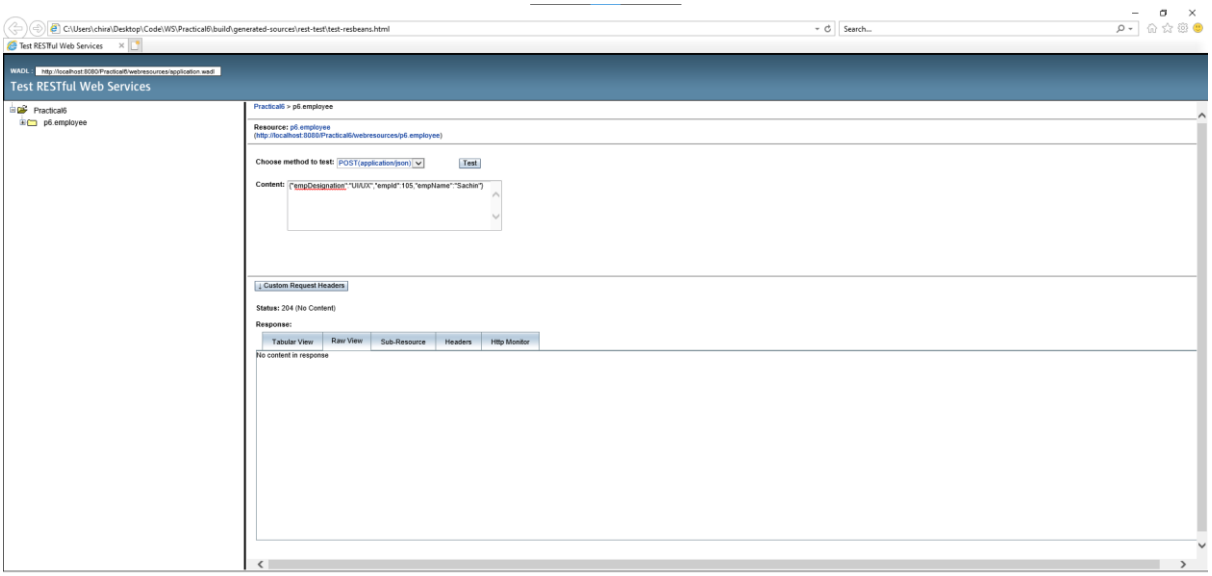

C:\Users\chira\Desktop\Code\WS\Practical6\build\generated-sources\rest-test\test-resbeans.html
Search...
Test RESTful Web Services
WADL : http://localhost:8080/Practical6/webresources/application.wadl
Test RESTful Web Services
Practical6
p6.employee
Select a node on the navigation bar (on the left side of this page) to test.

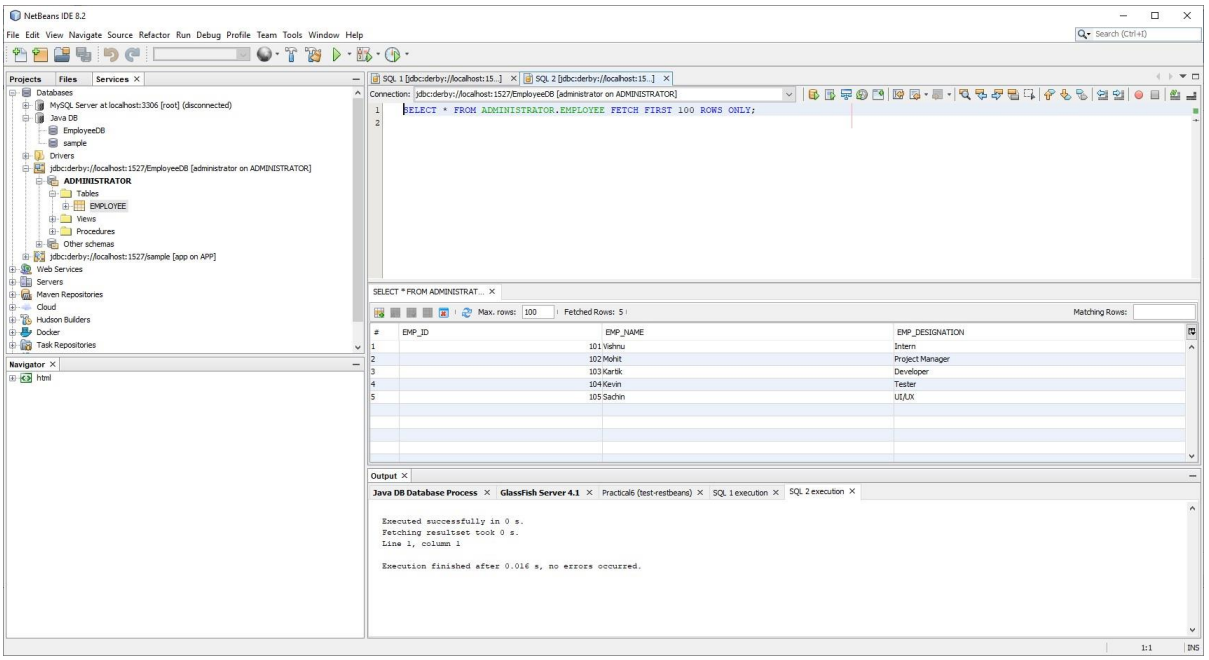

Test RESTful Web Services
Practical6
p6.employee
Choose method to test:
GET (application/xml)
Test
Custom Request Headers
Status: 200 (OK)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor


Test RESTful Web Services
Practical6
p6.employee
Resource: p6.employee
Choose method to test: GET(application/json)
Test
Custom Request Headers
Status: 200 (OK)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor


Test RESTful Web Services
Practical6
p6.employee
Resource: p6.employee
Choose method to test: POST(application/xml)
Test
Content:
<employee>
<empDesignation>Tester</empDesignation>
<empId>114</empId>
<empName>Kevin</empName>
</employee>
Custom Request Headers
Status: 204 (No Content)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor
No content in response


NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects
Files
Services
Navigator
Output
SELECT * FROM ADMINISTRATOR.EMPLOYEE FETCH FIRST 100 ROWS ONLY;
EMP_ID
EMP_NAME
EMP_DESIGNATION
Intern
Project Manager
Developer
Tester


Test RESTful Web Services
Test RESTful Web Services
Practical6
p6.employee
Resource: p6.employee
(http://localhost:8080/Practical6/webresources/p6.employee)
Choose method to test: POST(application/json)
Test
Content: {"empDesignation":"UI/UX","emplId":105,"empName":"Sachin"}
Custom Request Headers
Status: 204 (No Content)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor
No content in response


NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects Files Services
Databases
MySQL Server at localhost:3306 [root] (disconnected)
Java DB
EmployeeDB
sample
Drivers
jdbc:derby://localhost:1527/EmployeeDB [administrator on ADMINISTRATOR]
ADMINISTRATOR
Tables
EMPLOYEE
Views
Procedures
Other schemas
jdbc:derby://localhost:1527/sample [app on APP]
Web Services
Servers
Maven Repositories
Cloud
Hudson Builders
Docker
Task Repositories
Navigator
html
SQL 1 [jdbc:derby://localhost:15..]
SQL 2 [jdbc:derby://localhost:15..]
Connection: jdbc:derby://localhost:1527/EmployeeDB [administrator on ADMINISTRATOR]
SELECT * FROM ADMINISTRATOR.EMPLOYEE FETCH FIRST 100 ROWS ONLY;
SELECT * FROM ADMINISTRAT...
Max. rows: 100
Fetched Rows: 5
Matching Rows:
EMP_ID EMP_NAME EMP_DESIGNATION
1 101 Vishnu Intern
2 102 Mohit Project Manager
3 103 Kartik Developer
4 104 Kevin Tester
5 105 Sachin UI/UX
Output
Java DB Database Process
GlassFish Server 4.1
Practical6 (test-restbeans)
SQL 1 execution
SQL 2 execution
Executed successfully in 0 s.
Fetching resultset took 0 s.
Line 1, column 1
Execution finished after 0.016 s, no errors occurred.
1:1
INS

WEB SERVICES

PRACTICAL NO. 7

TCS2223013

CHIRAG BHATIA

Aim:

Design a Restful webservice from a database table Student with columns roll no, name and total marks. Create a restful client that displays the data by accessing restful service.

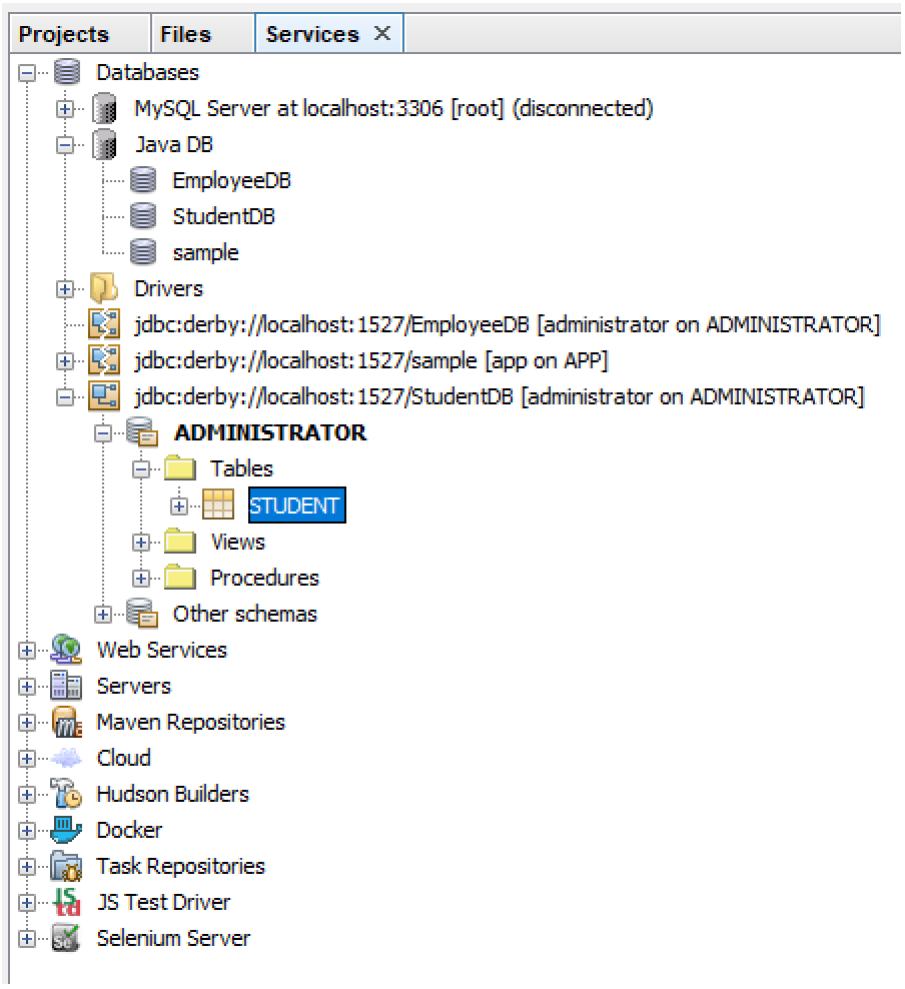
Code:

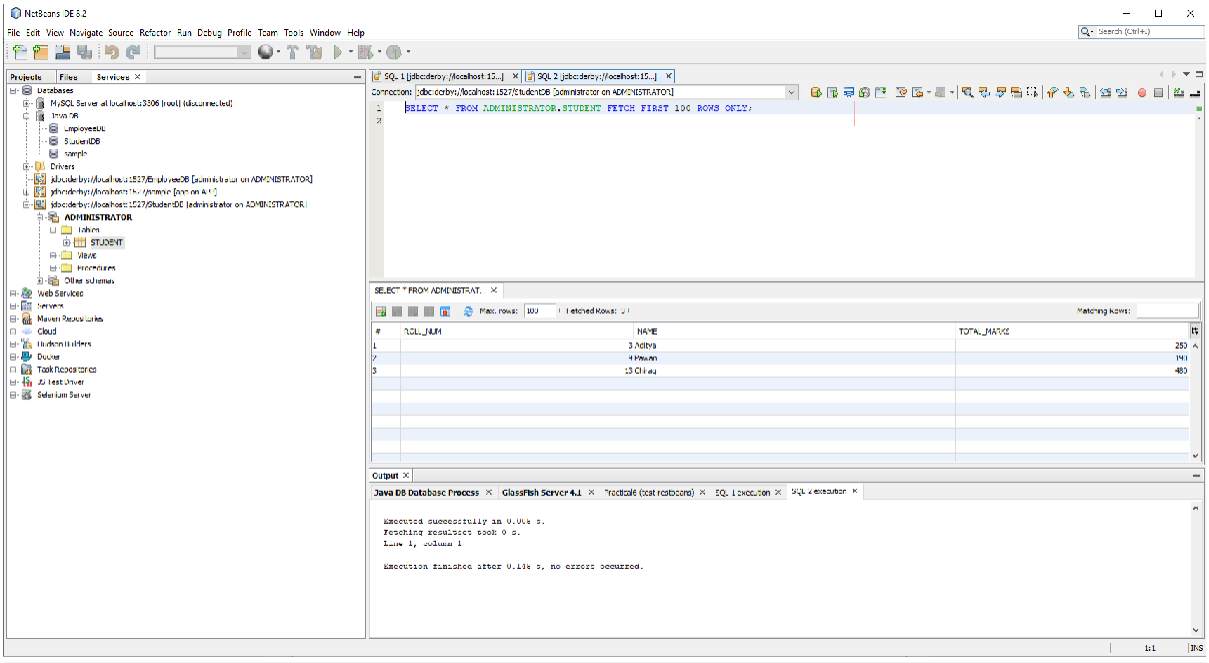
```
insert into STUDENT values(3, 'Aditya', 250);
insert into STUDENT values(9, 'Pawan', 190);
insert into STUDENT values(13, 'Chirag', 480);
```

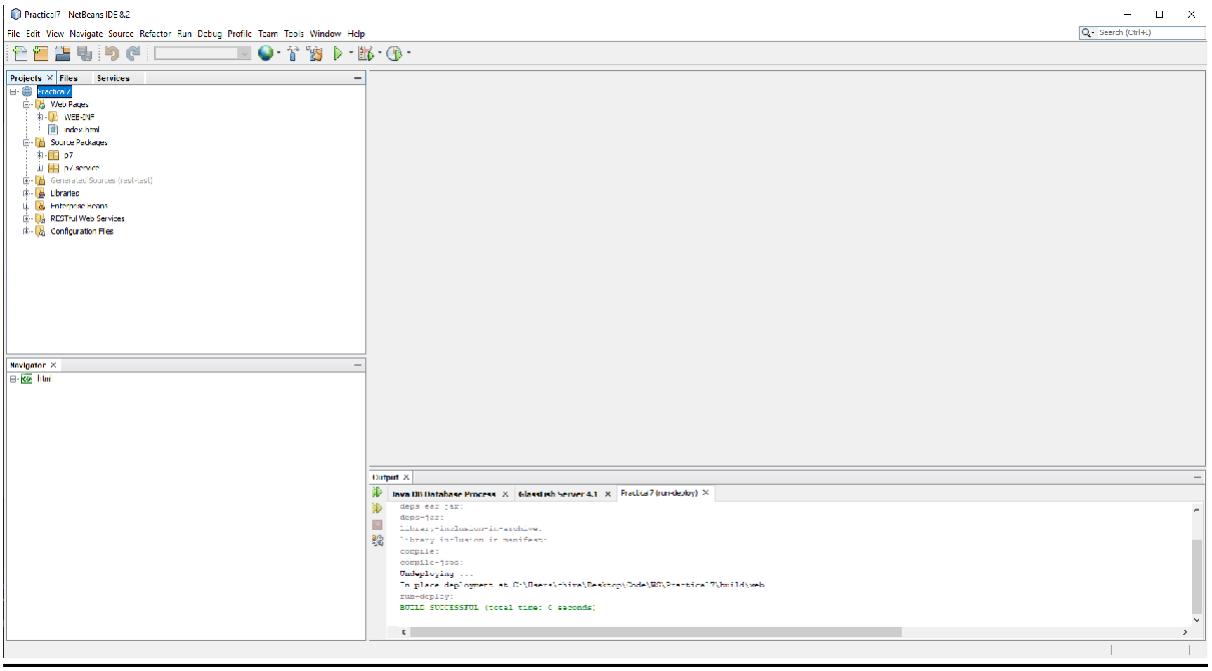
```
<student>
    <name>Rohan</name>
    <rollNum>4</rollNum>
    <totalMarks>270</totalMarks>
 </student>
```

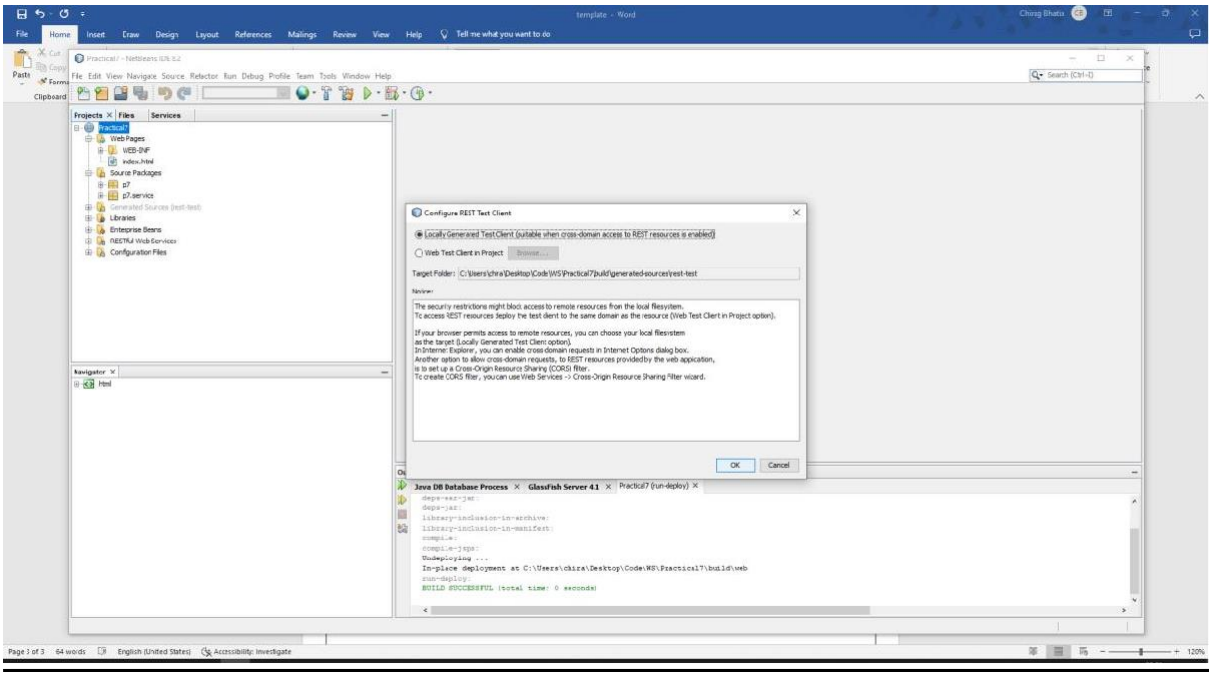
```
{"name":"Nishant","rollNum":5,"totalMarks":300}
```

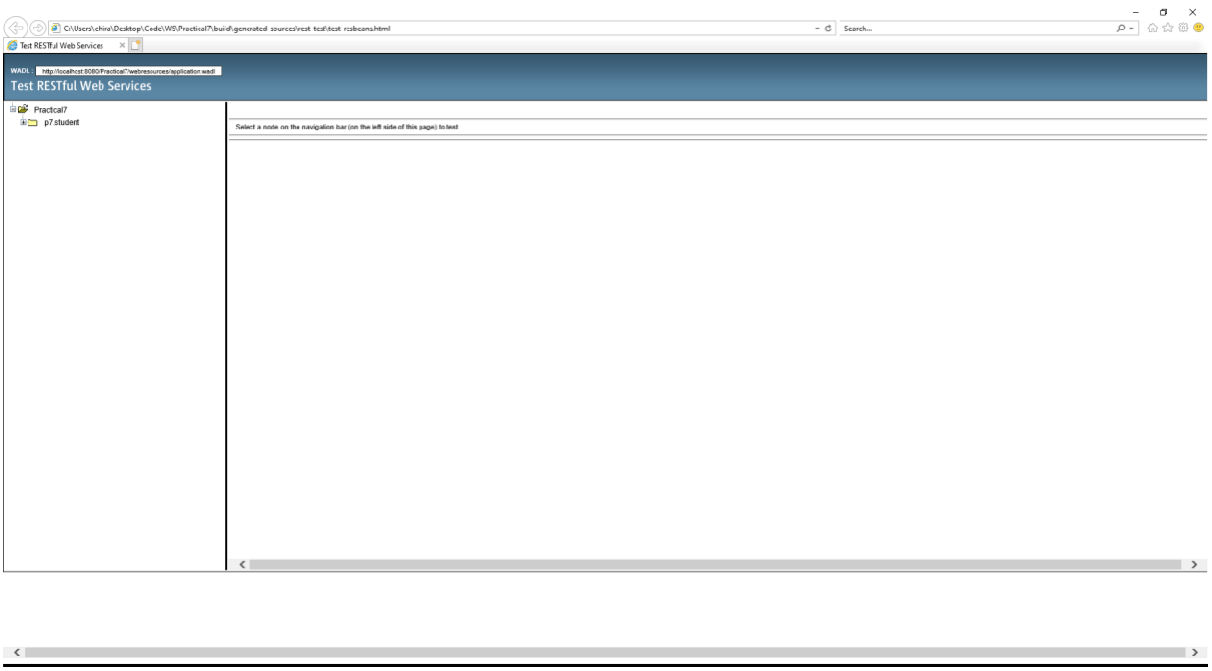
Output:

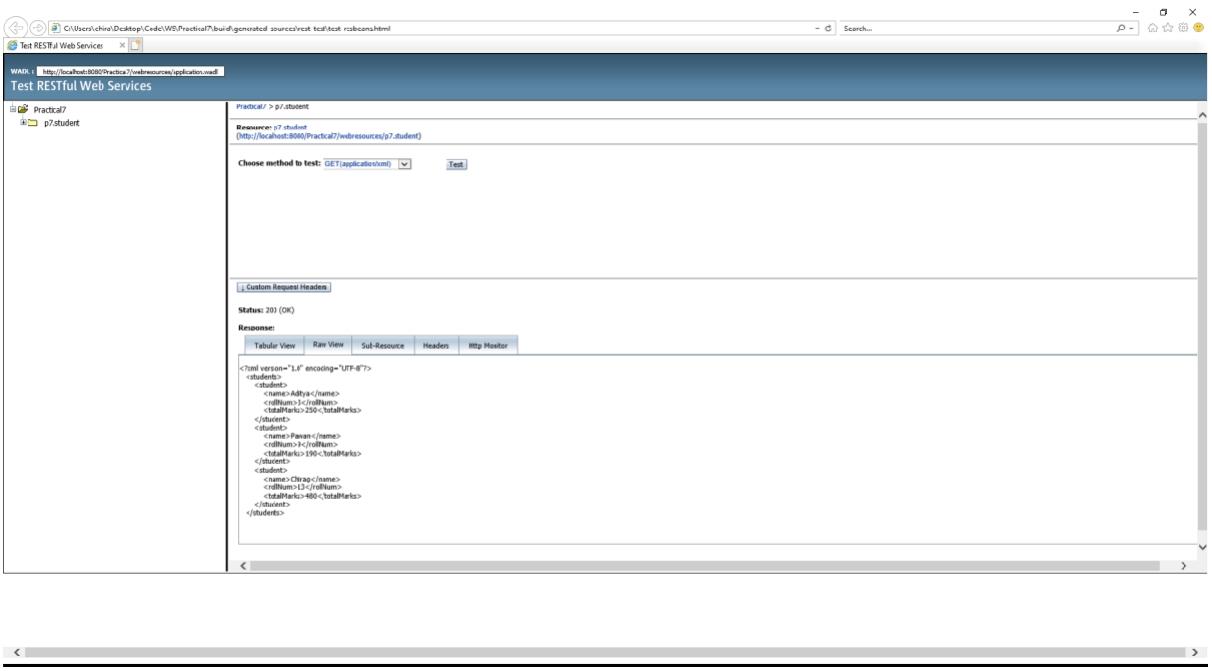

Projects
Files
Services
Databases
MySQL Server at localhost:3306 [root] (disconnected)
Java DB
EmployeeDB
StudentDB
sample
Drivers
jdbc:derby://localhost:1527/EmployeeDB [administrator on ADMINISTRATOR]
jdbc:derby://localhost:1527/sample [app on APP]
jdbc:derby://localhost:1527/StudentDB [administrator on ADMINISTRATOR]
ADMINISTRATOR
Tables
STUDENT
Views
Procedures
Other schemas
Web Services
Servers
Maven Repositories
Cloud
Hudson Builders
Docker
Task Repositories
JS Test Driver
Selenium Server

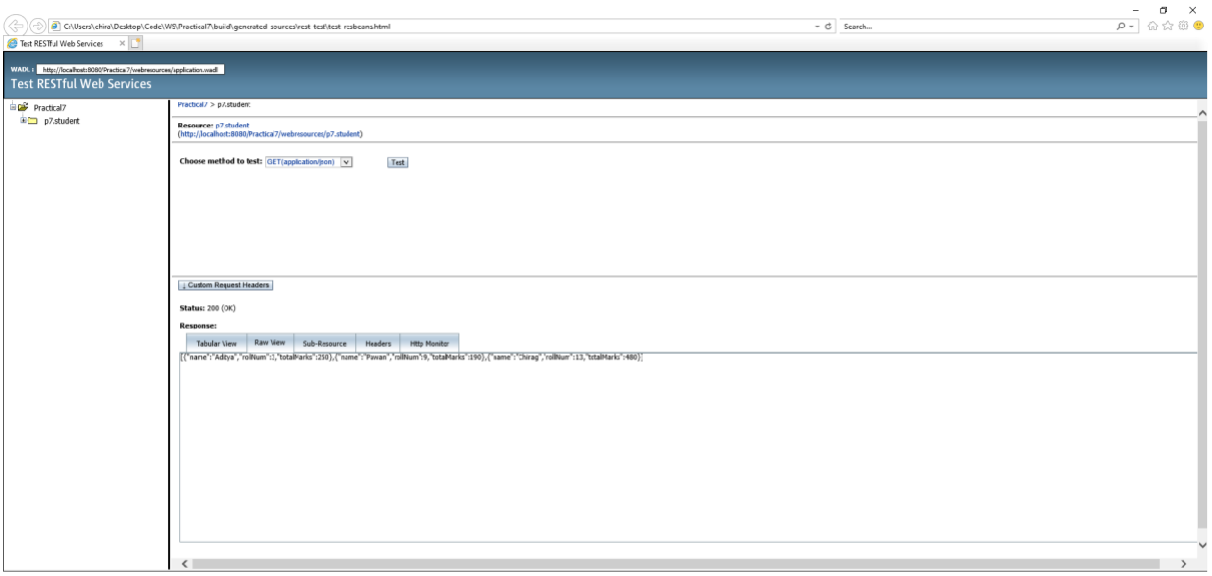

NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
SELECT * FROM ADMINISTRATOR.STUDENT FETCH FIRST 100 ROWS ONLY;

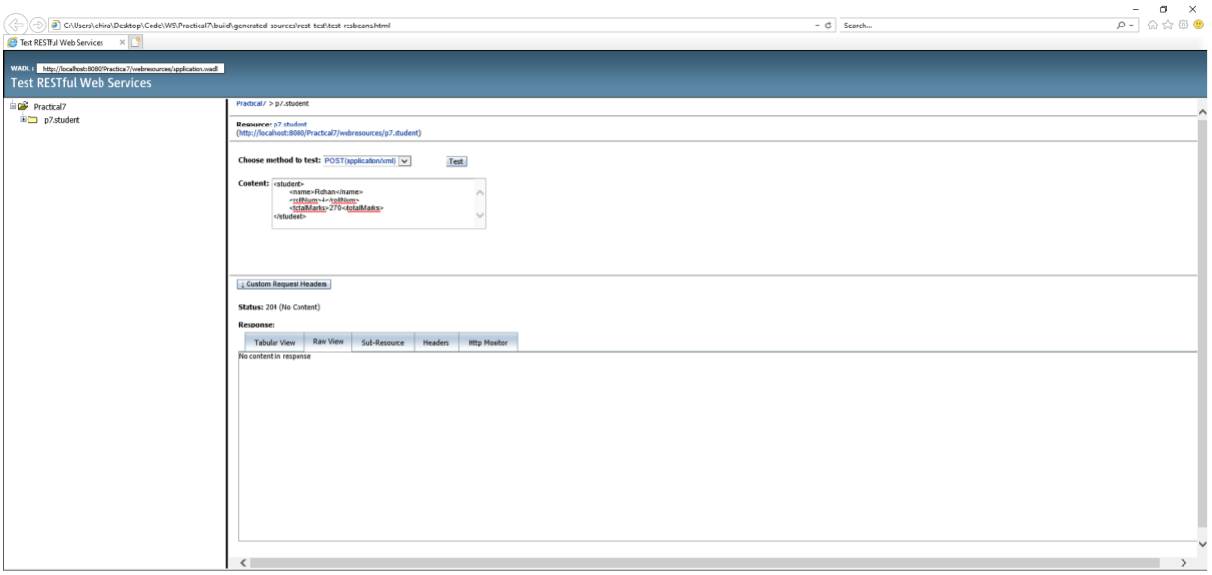

Practical7 - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects
Files
Services
Navigator
Output
BUILD SUCCESSFUL (total time: 0 seconds)

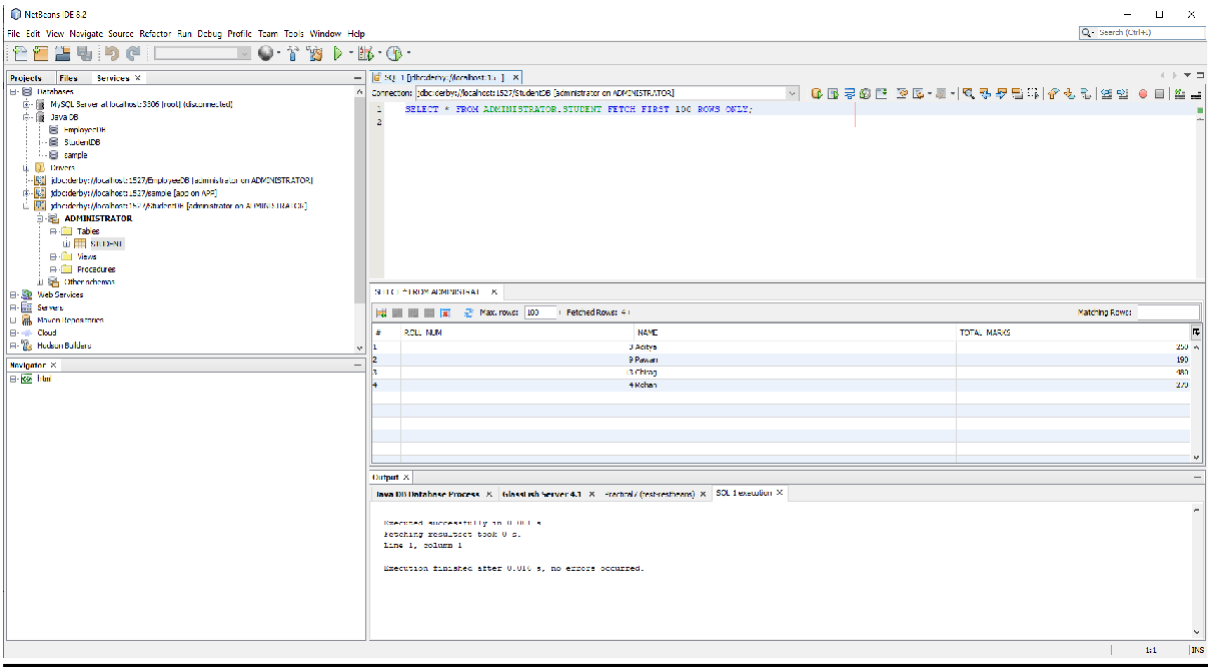

Configure REST Test Client
Locally Generated Test Client (suitable when cross-domain access to REST resources is enabled)
Web Test Client in Project
Target Folder:
OK
Cancel
Java DB Database Process
GlassFish Server 4.1
Practical7 (run-deploy)
BUILD SUCCESSFUL (total time: 0 seconds)

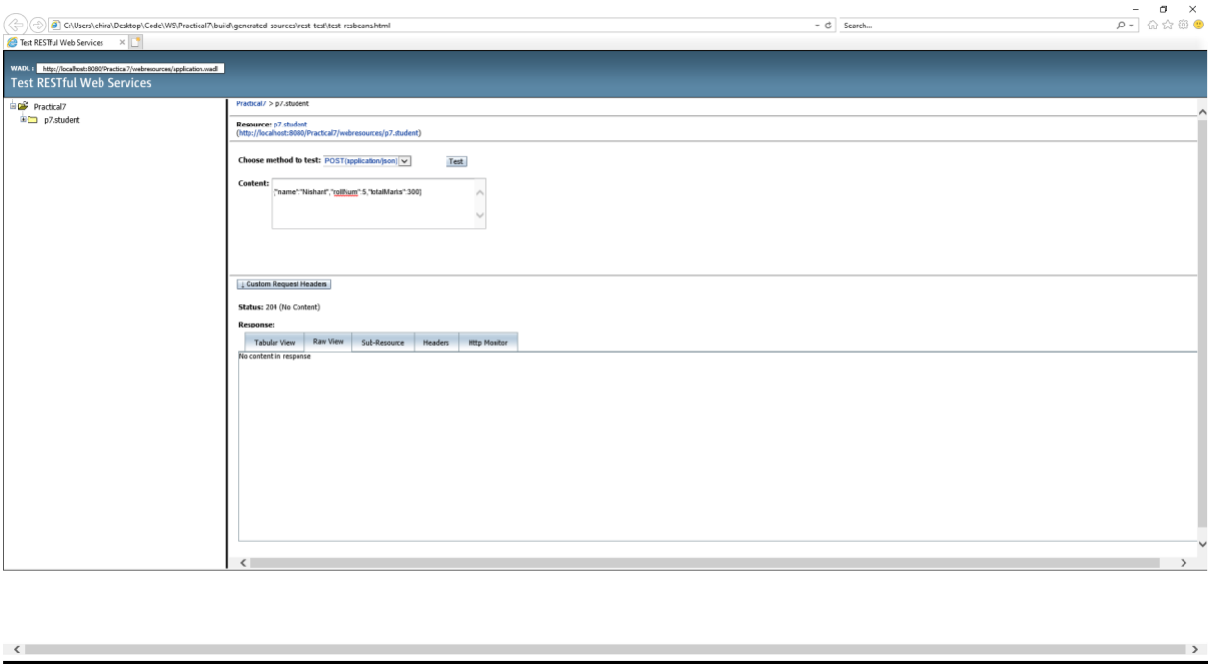

Test RESTful Web Services
Practical7
p7.student
Select a node on the navigation bar (on the left side of this page) to test

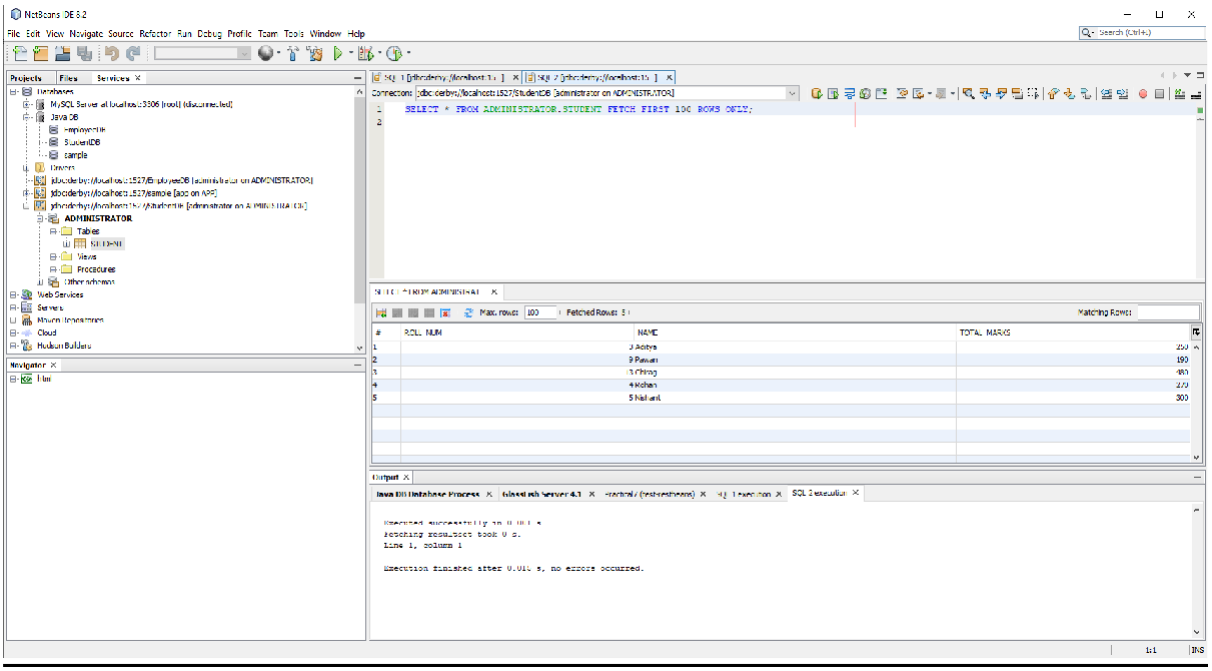

Test RESTful Web Services
Practical7
p7.student
Practical7 > p7.student
Resource: p7.student
(http://localhost:8080/Practical7/webresources/p7.student)
Choose method to test: GET(application/xml)
Test
Custom Request Headers
Status: 200 (OK)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor
<?xml version="1.0" encoding="UTF-8"?>
<students>
<student>
<name>Aditya</name>
<rollNum>1</rollNum>
<totalMarks>250</totalMarks>
</student>
<student>
<name>Pavan</name>
<rollNum>2</rollNum>
<totalMarks>190</totalMarks>
</student>
<student>
<name>Chirag</name>
<rollNum>3</rollNum>
<totalMarks>480</totalMarks>
</student>
</students>


Test RESTful Web Services
Practical7
p7.student
Choose method to test:
GET(application/json)
Test
Custom Request Headers
Status: 200 (OK)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor


Test RESTful Web Services
Practical7
p7.student
Choose method to test:
POST(application/xml)
Test
Content:
<name>Rohan</name>
<totalMarks>270</totalMarks>
Custom Request Headers
Status: 204 (No Content)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor
No content in response


NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects
Files
Services
Navigator
SELECT * FROM ADMINISTRATOR.STUDENT FETCH FIRST 100 ROWS ONLY;
ROLL_NUM
NAME
TOTAL_MARKS
1 Ashra 250
2 Pawan 190
3 Chirag 480
4 Mohan 270
Output
Fetching result set row 0 s.
Line 1, column 1
Execution finished after 0.016 s, no errors occurred.


Test RESTful Web Services
Test RESTful Web Services
Practical7
p7.student
Practical7 > p7.student
Resource: p7.student
(http://localhost:8080/Practical7/webresources/p7.student)
Choose method to test: POST(application/json)
Test
Content:
{"name":"Nishant","rollNum":5,"totalMarks":300}
Custom Request Headers
Status: 204 (No Content)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor
No content in response


NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects Files Services
Navigator
SELECT * FROM ADMINISTRATOR.STUDENT FETCH FIRST 100 ROWS ONLY;
ROLL_NUM
NAME
TOTAL_MARKS
Output

WEB SERVICES

PRACTICAL NO. 8

TCS2223013

CHIRAG BHATIA

Aim:

Create a WCF service to perform calculations like Addition, Subtraction, Multiplication and Division. Create a client for WCF which invokes the various operations.

Code:

IService1.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.Text;

namespace Practical8
{
    // NOTE: You can use the "Rename" command on the "Refactor" menu to change
the interface name "IService1" in both code and config file together.
    [ServiceContract]
    public interface IService1
    {
        [OperationContract]
        double Sum(double a, double b);

        [OperationContract]
        double Mul(double a, double b);
```

```
        [OperationContract]
        double Sub(double a, double b);

        [OperationContract]
        double Div(double a, double b);
    }
}
```

Service1.svc.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.Text;

namespace Practical8
{
    // NOTE: You can use the "Rename" command on the "Refactor" menu to change the class name "Service1" in code, svc and config file together.
    // NOTE: In order to launch WCF Test Client for testing this service, please select Service1.svc or Service1.svc.cs at the Solution Explorer and start debugging.
    public class Service1 : IService1
    {
        public double Mul(double a, double b)
        {
            double result = a * b;
            return result;
        }

        public double Sum(double a, double b)
        {
            double result = a + b;
            return result;
        }

        public double Sub(double a, double b)
        {
            double result = a - b;
            return result;
        }

        public double Div(double a, double b)
        {
            double result = a / b;
            return result;
        }
    }
}
```

WebForm1.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inherits="WebApplication1.WebForm1" %>
```

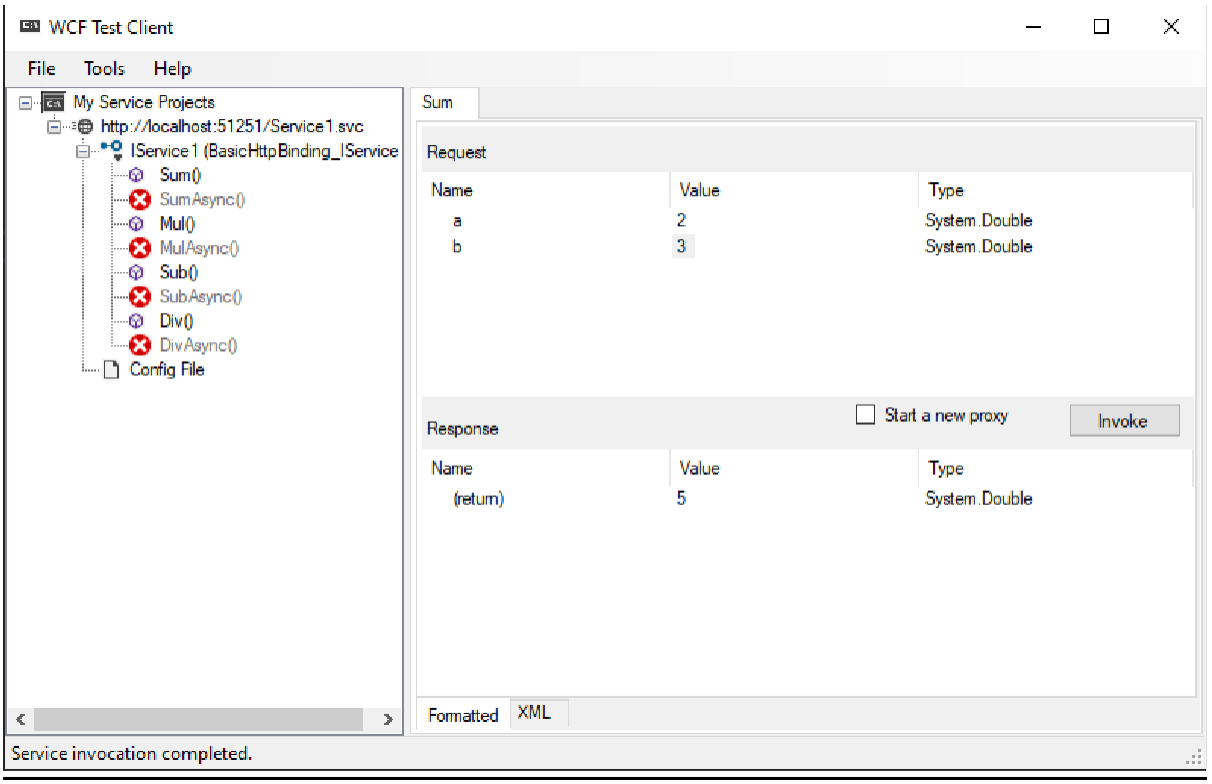
```
<!DOCTYfIE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            Number 1:
            <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
             <br />
            <br />
            Number2:
            <asp:TextBox ID="TextBox2" runat="server"
Width="127px"></asp:TextBox>
            <br />
            <asp:Button ID="Button1" runat="server" OnClick="Button1_Click"
Text="Add" />
             <asp:Button ID="Button2" runat="server" OnClick="Button2_Click"
Text="Multiplication" />
            <asp:Button ID="Button3" runat="server" OnClick="Button3_Click"
Text="Divide" />
            <asp:Button ID="Button4" runat="server" OnClick="Button4_Click"
Text="Subtract" />
            <br />
            <asp:Label ID="Label1" runat="server"></asp:Label>

        </div>
    </form>
</body>
</html>
```

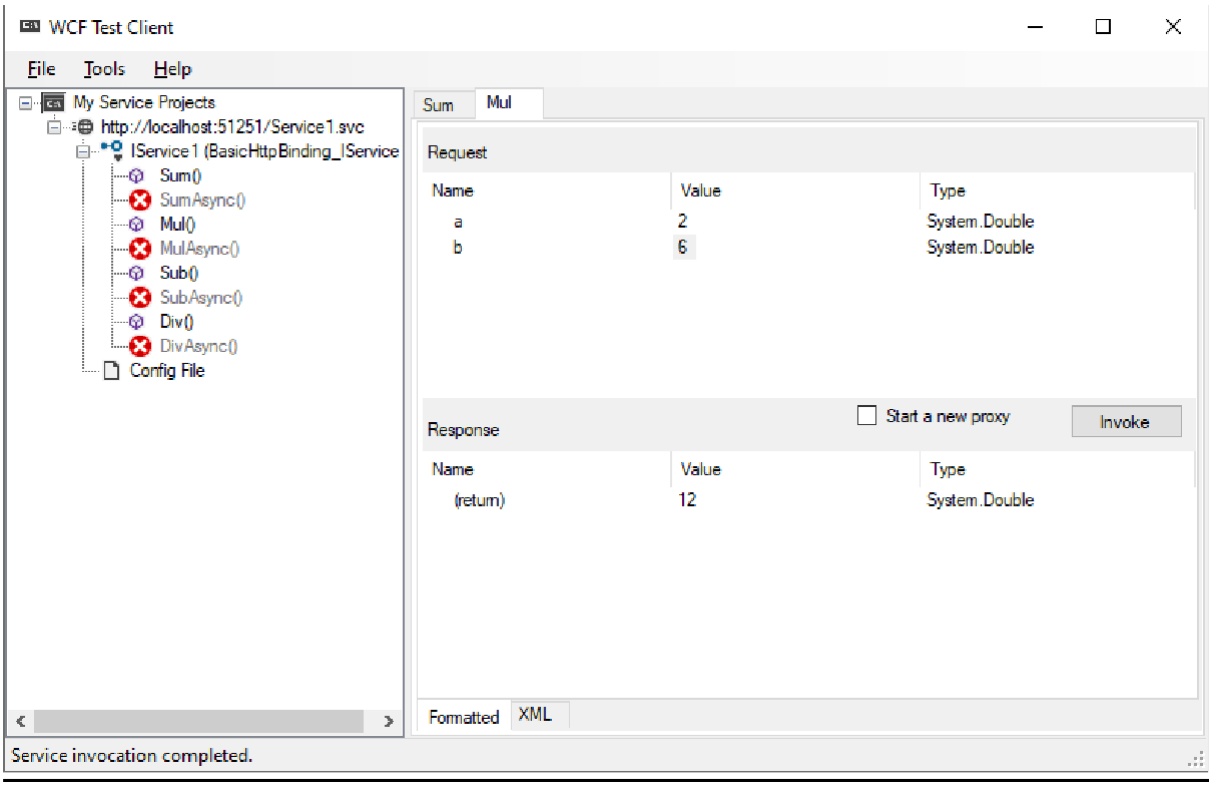
WebForm1.aspx.cs

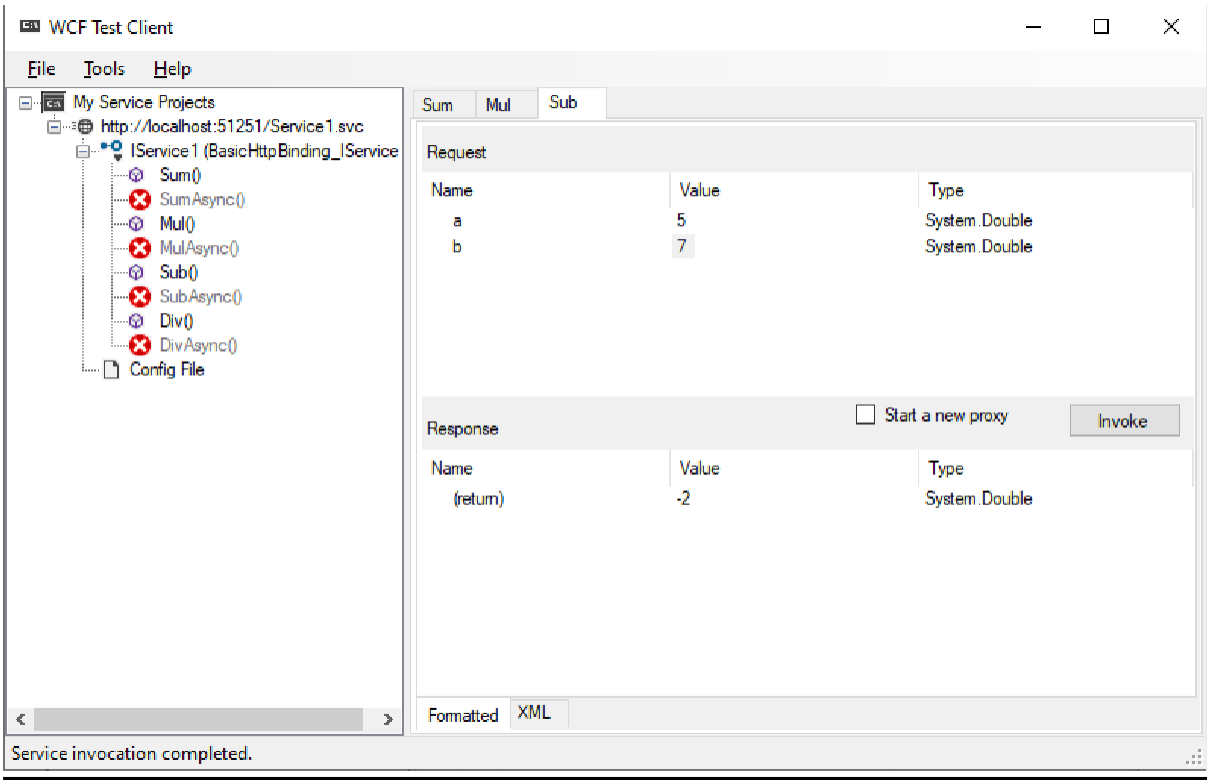
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace WebApplication1
{
    public partial class WebForm1 : System.Web.UI.flage
    {
        protected void flage_Load(object sender, EventArgs e)
        {

        }
        protected void Button1_Click(object sender, EventArgs e)
        {
            ServiceReference1.Service1Client client = new
ServiceReference1.Service1Client();
            double a = double.flarse(TextBox1.Text);
            double b = double.flarse(TextBox2.Text);
            Label1.Text = "Sum: " + client.Sum(a, b).ToString();
        }
```

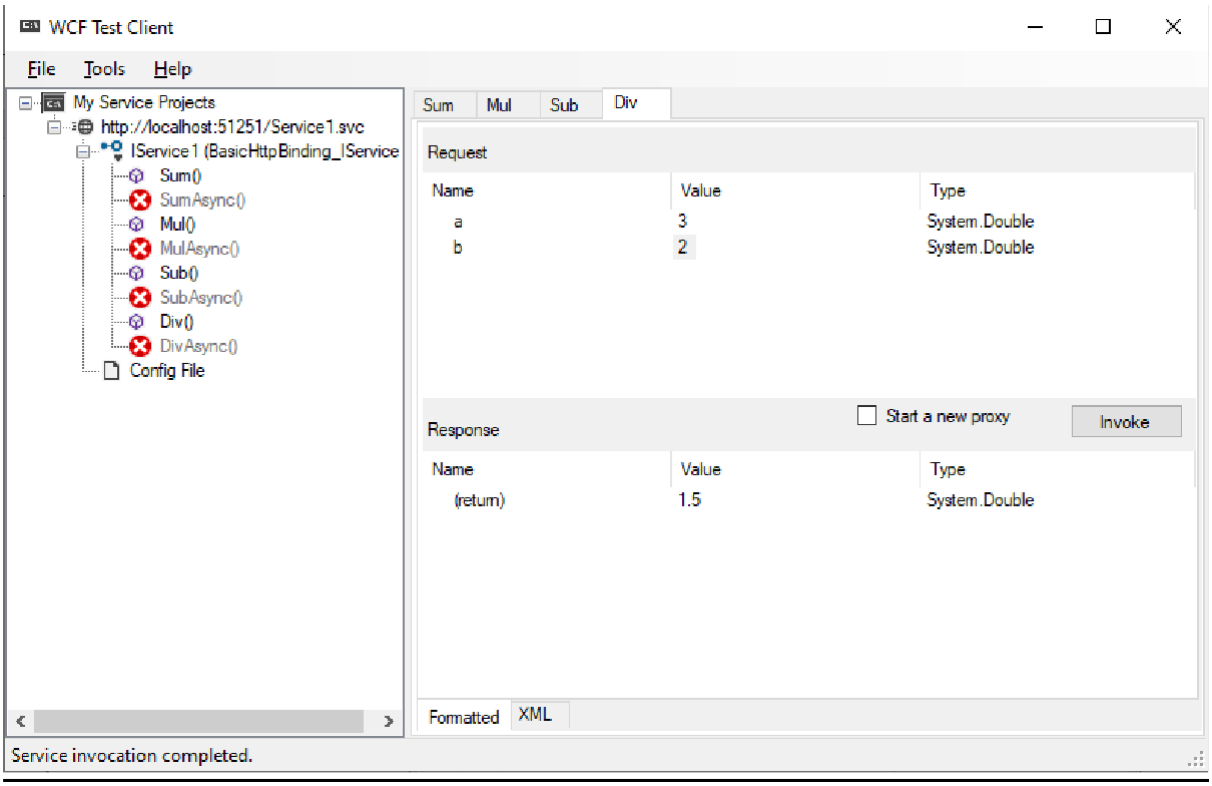
```
        protected void Button2_Click(object sender, EventArgs e)
        {
            ServiceReference1.Service1Client client = new
ServiceReference1.Service1Client();
            double a = double.flarse(TextBox1.Text);
            double b = double.flarse(TextBox2.Text);
            Label1.Text = "flroduct: " + client.Mul(a, b).ToString();
        }

        protected void Button3_Click(object sender, EventArgs e)
        {
            ServiceReference1.Service1Client client = new
ServiceReference1.Service1Client();
            double a = double.flarse(TextBox1.Text);
            double b = double.flarse(TextBox2.Text);
            Label1.Text = "Quotient: " + client.Div(a, b).ToString();
        }

        protected void Button4_Click(object sender, EventArgs e)
        {
            ServiceReference1.Service1Client client = new
ServiceReference1.Service1Client();
            double a = double.flarse(TextBox1.Text);
            double b = double.flarse(TextBox2.Text);
            Label1.Text = "Difference: " + client.Sub(a, b).ToString();
        }
    }
}
```

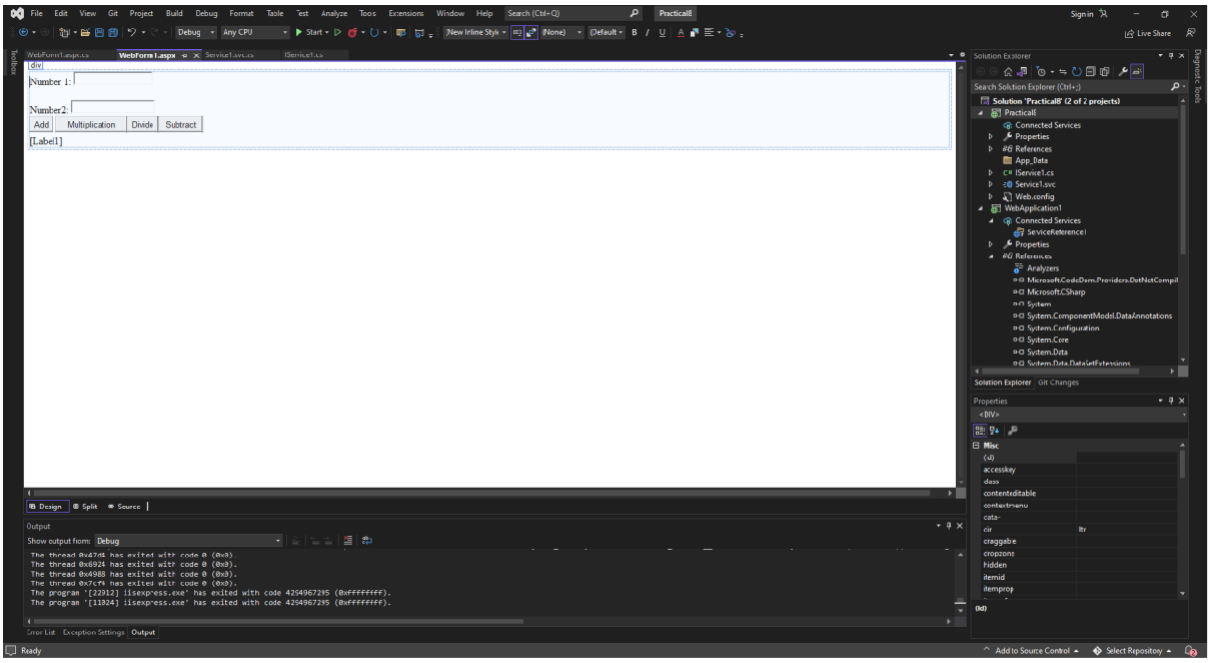
Output:


WCF Test Client
File
Tools
Help
My Service Projects
http://localhost:51251/Service1.svc
IService1 (BasicHttpBinding_IService
Sum()
SumAsync()
Mul()
MulAsync()
Sub()
SubAsync()
Div()
DivAsync()
Config File
Sum
Request
Name
Value
Type
a
2
System.Double
b
3
System.Double
Response
Start a new proxy
Invoke
Name
Value
Type
(return)
5
System.Double
Formatted
XML
Service invocation completed.


WCF Test Client
File
Tools
Help
My Service Projects
http://localhost:51251/Service1.svc
IService1 (BasicHttpBinding_IService
Sum()
SumAsync()
Mul()
MulAsync()
Sub()
SubAsync()
Div()
DivAsync()
Config File
Sum
Mul
Request
Name
Value
Type
a
2
System.Double
b
6
System.Double
Response
Start a new proxy
Invoke
Name
Value
Type
(return)
12
System.Double
Formatted
XML
Service invocation completed.


WCF Test Client
File
Tools
Help
My Service Projects
http://localhost:51251/Service1.svc
IService1 (BasicHttpBinding_IService
Sum()
SumAsync()
Mul()
MulAsync()
Sub()
SubAsync()
Div()
DivAsync()
Config File
Sum
Mul
Sub
Request
Name
Value
Type
a
5
System.Double
b
7
System.Double
Response
Start a new proxy
Invoke
Name
Value
Type
(return)
-2
System.Double
Formatted
XML
Service invocation completed.


WCF Test Client
File
Tools
Help
My Service Projects
http://localhost:51251/Service1.svc
IService1 (BasicHttpBinding_IService
Sum()
SumAsync()
Mul()
MulAsync()
Sub()
SubAsync()
Div()
DivAsync()
Config File
Sum
Mul
Sub
Div
Request
Name
Value
Type
a
3
System.Double
b
2
System.Double
Response
Start a new proxy
Invoke
Name
Value
Type
(return)
1.5
System.Double
Formatted
XML
Service invocation completed.

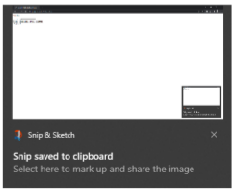

Number 1:
Number2:
Add
Multiplication
Divide
Subtract
[Label1]


localhost:54610/WebForm1.aspx
Number 1:
Number2:
Add
Multiplication
Divide
Subtract

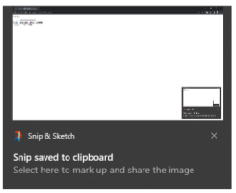

localhost:54610/WebForm1.aspx
Number 1: 4
Number2: 7
Add
Multiplication
Divide
Subtract
Sum: 11


localhost:54610/WebForm1.aspx
Number 1: 4
Number2: 7
Add
Multiplication
Divide
Subtract
Product: 28
Snip & Sketch
Snip saved to clipboard
Select here to mark up and share the image


localhost:54610/WebForm1.aspx
Number 1: 4
Number2: 7
Add
Multiplication
Divide
Subtract
Quotient: 0.571428571428571


Snip & Sketch
Snip saved to clipboard
Select here tc mark up and share the image


localhost:54610/WebForm1.aspx
Number 1: 4
Number2: 7
Add
Multiplication
Divide
Subtract
Difference -3


Snip & Sketch
Snip saved to clipboard
Select here tc mark up and share the image

WEB SERVICES

PRACTICAL NO. 9

TCS2223013

CHIRAG BHATIA

Aim:

Create a WCF service with different endpoints for Soap based and Rest based implementation

Code:

IService1.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.Text;

namespace WCFflractical
{
    // NOTE: You can use the "Rename" command on the "Refactor" menu to change
the interface name "IService1" in both code and config file together.
    [ServiceContract]
    public interface IService1
    {
        [OperationContract]
        [System.ServiceModel.Web.WebInvoke(Method="GET",UriTemplate
="/SayHello/{value}",RequestFormat
=System.ServiceModel.Web.WebMessageFormat.Json,ResponseFormat
=System.ServiceModel.Web.WebMessageFormat.Json)]
        string SayHello(string value);
    }
```

```
}
```

Service1.svc.cs

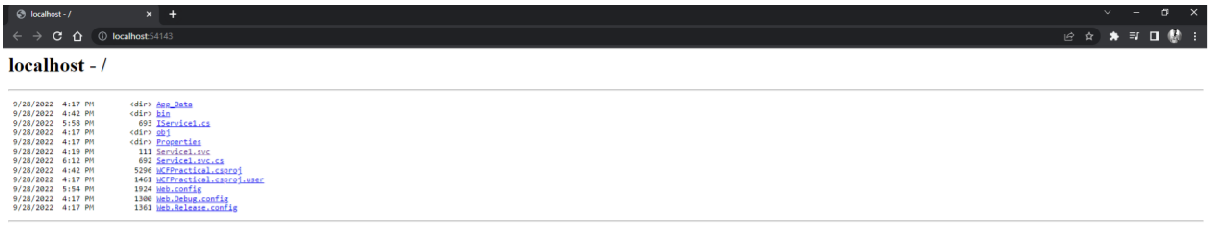
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.Text;

namespace WCFfIractical
{
    // NOTE: You can use the "Rename" command on the "Refactor" menu to change
the class name "Service1" in code, svc and config file together.
    // NOTE: In order to launch WCF Test Client for testing this service, please
select Service1.svc or Service1.svc.cs at the Solution Explorer and start
debugging.
    public class Service1 : IService1
    {
        public string SayHello(string value)
        {
            return string.Format($"Hello {value}! Welcome to WCF");
        }
    }
}
```

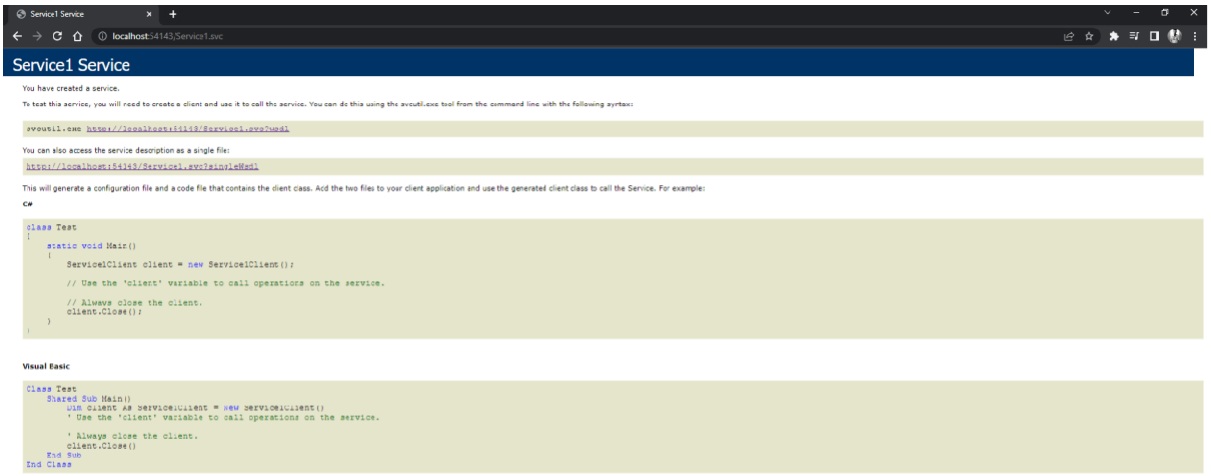
Web.config

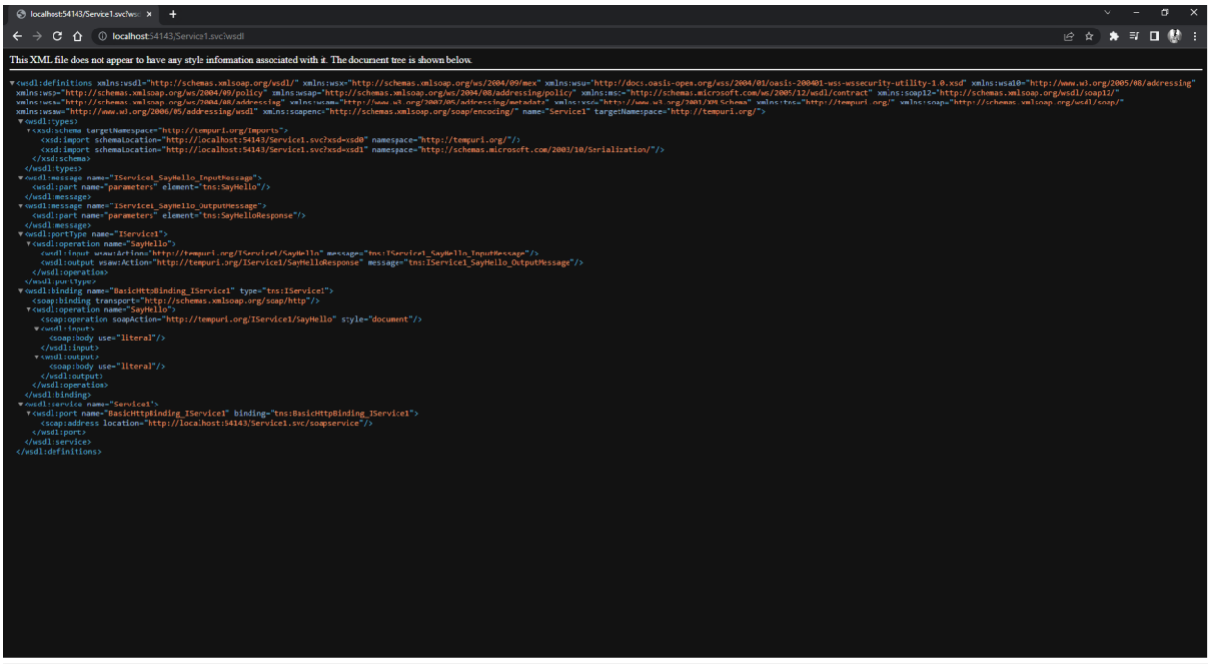
```
<oxml version="1.0"o>
<configuration>

  <appSettings>
    <add key="aspnet:UseTaskFriendlySynchronizationContext" value="true" />
  </appSettings>
  <system.web>
    <compilation debug="true" targetFramework="4.8" />
    <httpRuntime targetFramework="4.8"/>
  </system.web>
  <system.serviceModel>
        <services>
              <service name="WCFfIractical.Service1">
                    <endpoint address="jsonservice" binding="webHttpBinding"
contract="WCFfIractical.IService1" behaviorConfiguration="web"></endpoint>
                    <endpoint address="soapservice" binding="basicHttpBinding"
contract="WCFfIractical.IService1"></endpoint>
              </service>
        </services>
    <behaviors>
      <serviceBehaviors>
        <behavior>
          <!-- To avoid disclosing metadata information, set the values below to
false before deployment -->
          <serviceMetadata httpGetEnabled="true" httpsGetEnabled="true"/>
          <!-- To receive exception details in faults for debugging purposes, set
the value below to true.  Set to false before deployment to avoid disclosing
exception information -->
```

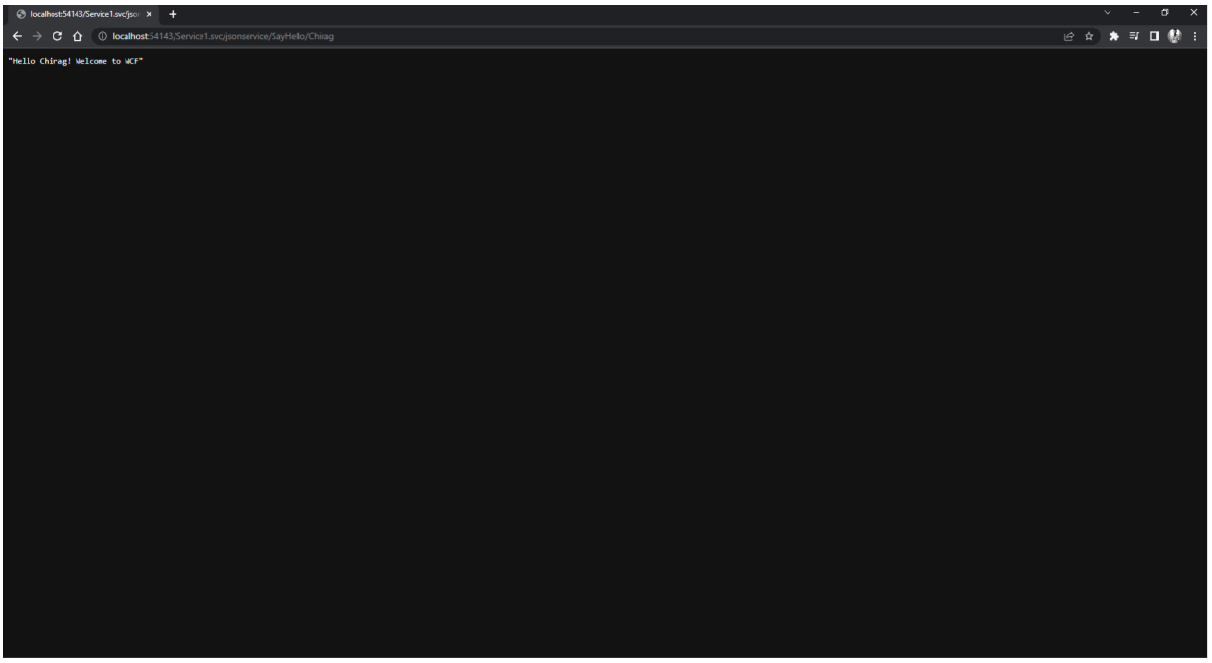
```
            <serviceDebug includeExceptionDetailInFaults="false"/>
        </behavior>
      </serviceBehaviors>
            <endpointBehaviors>
                  <behavior name="web">
                        <webHttp/>
                  </behavior>
            </endpointBehaviors>
    </behaviors>
    <protocolMapping>
        <add binding="basicHttpsBinding" scheme="https" />
    </protocolMapping>
    <serviceHostingEnvironment aspNetCompatibilityEnabled="true"
multipleSiteBindingsEnabled="true" />
  </system.serviceModel>
  <system.webServer>
    <modules runAllManagedModulesForAllRequests="true"/>
    <!--
        To browse web app root directory during debugging, set the value below to
true.
        Set to false before deployment to avoid disclosing web app folder
information.
      -->
    <directoryBrowse enabled="true"/>
  </system.webServer>

</configuration>
```

Output:




Service1 Service
localhost:54143/Service1.svc
Service1 Service
You have created a service.
To test this service, you will need to create a client and use it to call the service. You can do this using the svcutil.exe tool from the command line with the following syntax:
svcutil.exe http://localhost:54143/Service1.svc?wsdl
You can also access the service description as a single file:
http://localhost:54143/Service1.svc?singleWsdl
This will generate a configuration file and a code file that contains the client class. Add the two files to your client application and use the generated client class to call the Service. For example:
C#
class Test
{
static void Main()
{
Service1Client client = new Service1Client();
// Use the 'client' variable to call operations on the service.
// Always close the client.
client.Close();
}
}
Visual Basic
Class Test
Shared Sub Main()
Dim client As Service1Client = New Service1Client()
' Use the 'client' variable to call operations on the service.
' Always close the client.
client.Close()
End Sub
End Class


localhost:54143/Service1.svc?wsdl
This XML file does not appear to have any style information associated with it. The document tree is shown below.


localhost:54143/Service1.svc/jso
localhost:54143/Service1.svc/jsonservice/SayHello/Chirag
"Hello Chirag! Welcome to WCF"

WEB SERVICES

PRACTICAL NO. 10

TCS2223013

CHIRAG BHATIA

Aim:

Design a Restful webservice and create a restful client that displays the data by fetching it using HTTP GET method

Code:

Server

```
INSERT INTO AIRLINES VALUES(100,'AirIndia','23 September 2022','14 : 20
Hrs');
INSERT INTO AIRLINES VALUES(200,'Vistara','24 September 2022','15 : 40
Hrs');
INSERT INTO AIRLINES VALUES(300,'Qatar','25 September 2022','16 : 00 Hrs');
package as.service;

import as.Airlines;
import java.util.List;
import javax.ejb.Stateless;
import javax.persistence.EntityManager;
import javax.persistence.PersistenceContext;
import javax.ws.rs.Consumes;
import javax.ws.rs.DELETE;
import javax.ws.rs.GET;
import javax.ws.rs.POST;
import javax.ws.rs.PUT;
import javax.ws.rs.Path;
import javax.ws.rs.PathParam;
import javax.ws.rs.Produces;
import javax.ws.rs.core.MediaType;
```

```
@Stateless
@Path("as.airlines")
public class AirlinesFacadeREST extends AbstractFacade<Airlines> {

    @PersistenceContext(unitName = "AirlineServerPU")
    private EntityManager em;

    public AirlinesFacadeREST() {
        super(Airlines.class);
    }

    @POST
    @Override
    @Consumes({ MediaType.APPLICATION_JSON})
    public void create(Airlines entity) {
        super.create(entity);
    }

    @PUT
    @Path("{id}")
    @Consumes({ MediaType.APPLICATION_JSON})
    public void edit(@PathParam("id") Integer id, Airlines entity) {
        super.edit(entity);
    }

    @DELETE
    @Path("{id}")
    public void remove(@PathParam("id") Integer id) {
        super.remove(super.find(id));
    }

    @GET
    @Path("{id}")
    @Produces({ MediaType.APPLICATION_JSON})
    public Airlines find(@PathParam("id") Integer id) {
        return super.find(id);
    }

    @GET
    @Override
    @Produces({ MediaType.APPLICATION_JSON})
    public List<Airlines> findAll() {
        return super.findAll();
    }

    @GET
    @Path("{from}/{to}")
    @Produces({ MediaType.APPLICATION_JSON})
    public List<Airlines> findRange(@PathParam("from") Integer from,
@PathParam("to") Integer to) {
        return super.findRange(new int[]{from, to});
    }

    @GET
    @Path("count")
    @Produces(MediaType.TEXT_PLAIN)
    public String countREST() {
        return String.valueOf(super.count());
```

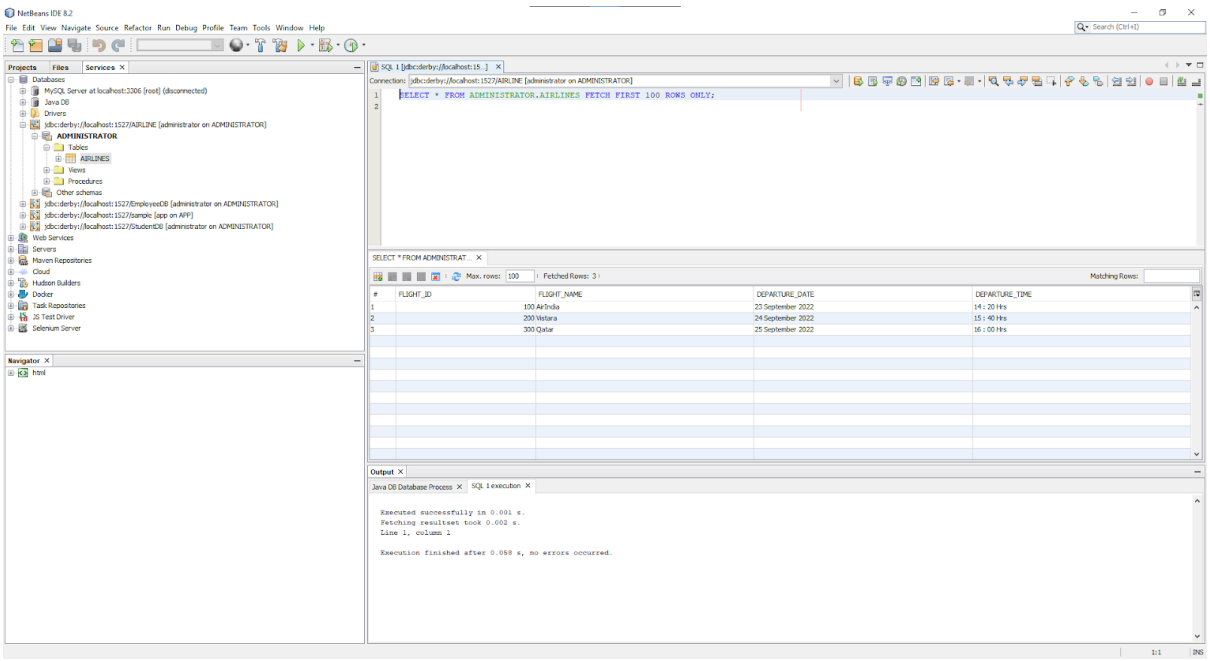
```
    }

    @Override
    protected EntityManager getEntityManager() {
        return em;
    }

}
```

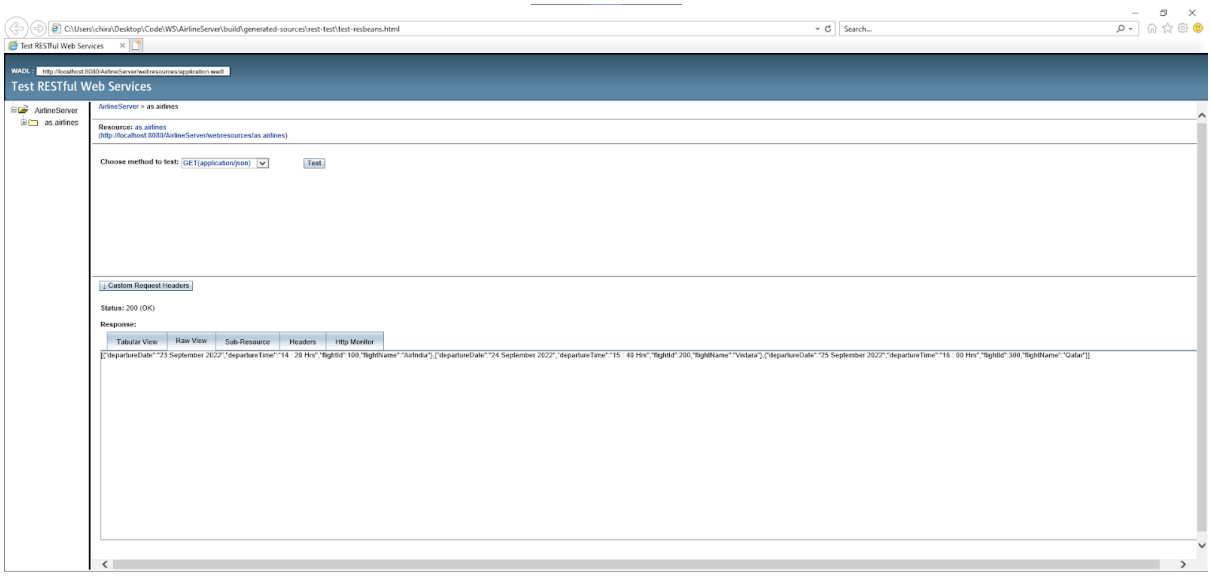
Client

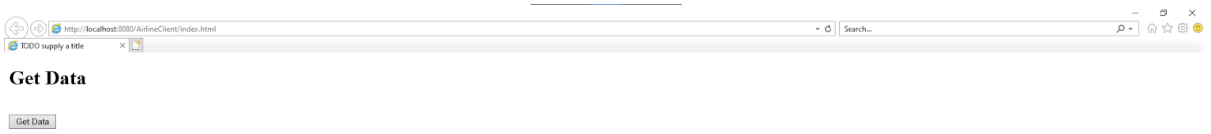
```
<!DOCTYPE html>
<html>
    <head>
        <title>TODO supply a title</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    </head>
    <body>
        <form >
            <h1>Get Data</h1>
            <br>
            <input type="submit" formaction="getData.jsp" value="Get
Data">
        </form>
    </body>
</html>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <title>TODO</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-
scale=1.0">
        <style> table {
                font-family: arial, sans-serif; border-collapse: collapse;
            }
            td, th {
                border: 1px solid #000000; text-align: center; padding:
8px;
            }
        </style>
        <script>
            var request = new XMLHttpRequest();
            request.open('GET',
                    'http://localhost:8080/AirlineServer/webresources/as.ai
rlines', true);
            request.onload = function () {
                var data = JSON.parse(this.response);
                console.log(this.response);
                for (var i = 0; i < data.length; i++) {
                    var table = document.getElementById("myTable");
                    var row = table.insertRow();
                    var cell1 = row.insertCell(0);
                    var cell2 = row.insertCell(1);
                    var cell3 = row.insertCell(2);
                    var cell4 = row.insertCell(3);
                    cell1.innerHTML = data[i].flightId;
                    cell2.innerHTML = data[i].flightName;
```

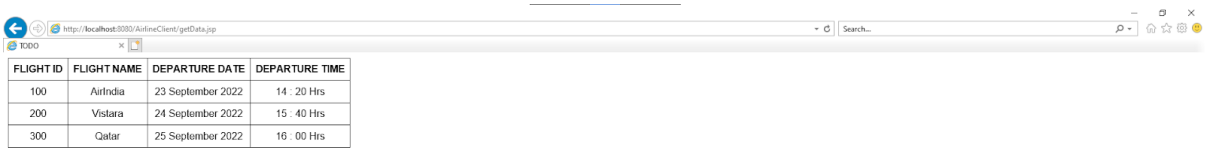
```
                    cell3.innerHTML = data[i].departureDate;
                    cell4.innerHTML = data[i].departureTime;
                }
            };
            request.send();
        </script>
    </head>
    <body>
        <table id="myTable">
            <tr>
                <th>FLIGHT ID</th>
                <th>FLIGHT NAME</th>
                <th>DEPARTURE DATE</th>
                <th>DEPARTURE TIME</th>
            </tr>
        </table>
    </body>
</html>
```

Output:




Test RESTful Web Services
Test RESTful Web Services
AirlineServer
as.airlines
AirlineServer > as.airlines
Resource: as.airlines
Choose method to test: GET(application/json)
Test
Custom Request Headers
Status: 200 (OK)
Response:
Tabular View
Raw View
Sub-Resource
Headers
Http Monitor


TODO supply a title
Get Data
Get Data



FLIGHT ID	FLIGHT NAME	DEPARTURE DATE	DEPARTURE TIME
100	AirIndia	23 September 2022	14 : 20 Hrs
200	Vistara	24 September 2022	15 : 40 Hrs
300	Qatar	25 September 2022	16 : 00 Hrs