**CalculatorServlet:**

/\*

\* 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.

\*/

import com.myservice.MyCalculatorWebService\_Service;

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.xml.ws.WebServiceRef;

/\*\*

\*

\* @author ASUS

\*/

public class CalculatorServlet extends HttpServlet {

@WebServiceRef(wsdlLocation = "WEB-INF/wsdl/localhost\_8080/Assignment7/MyCalculatorWebService.wsdl")

private MyCalculatorWebService\_Service service;

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code>

\* methods.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

double num1,num2;

num1 =Double.parseDouble(request.getParameter("number1"));

num2 =Double.parseDouble(request.getParameter("number2"));

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>CalculatorServlet Output</title>");

out.println("</head>");

out.println("<body>");

out.println("<h1> Addition is : " +addition(num1,num2)+ "</h1>");

out.println("<h1> Substraction is : " +substraction(num1,num2)+ "</h1>");

out.println("<h1> Multiplication is : " +multiplication(num1,num2)+ "</h1>");

out.println("<h1> Division is : " +division(num1,num2)+ "</h1>");

out.println("</body>");

out.println("</html>");

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\*

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

private double addition(double num1, double num2) {

// Note that the injected javax.xml.ws.Service reference as well as port objects are not thread safe.

// If the calling of port operations may lead to race condition some synchronization is required.

com.myservice.MyCalculatorWebService port = service.getMyCalculatorWebServicePort();

return port.addition(num1, num2);

}

private double substraction(double num1, double num2) {

// Note that the injected javax.xml.ws.Service reference as well as port objects are not thread safe.

// If the calling of port operations may lead to race condition some synchronization is required.

com.myservice.MyCalculatorWebService port = service.getMyCalculatorWebServicePort();

return port.substraction(num1, num2);

}

private double multiplication(double num1, double num2) {

// Note that the injected javax.xml.ws.Service reference as well as port objects are not thread safe.

// If the calling of port operations may lead to race condition some synchronization is required.

com.myservice.MyCalculatorWebService port = service.getMyCalculatorWebServicePort();

return port.multiplication(num1, num2);

}

private double division(double num1, double num2) {

// Note that the injected javax.xml.ws.Service reference as well as port objects are not thread safe.

// If the calling of port operations may lead to race condition some synchronization is required.

com.myservice.MyCalculatorWebService port = service.getMyCalculatorWebServicePort();

return port.division(num1, num2);

}

}

**index.html:**

Top of Form

<!DOCTYPE html>

<!--

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.

-->

<html>

<head>

<title>Calculator Web Service Client</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

<form action="CalculatorServlet">

Enter Number-1:<input type="text" name="number1" value=""/><br>

Enter Number-2:<input type="text" name="number2" value=""/><br>

<input type="submit" value="Submit" />

</form>

</body>

</html>

Bottom of Form

Bottom of Form