Name: Mrunal Sanjay Chaudhari
Roll No: 47006   Class: BE-IT-B
Subject: Distributed Systems

Assignment No. 7

Problem Statement: Create a simple web service and write distributed application (calculator) to consume the Web Service.

Code:

Web Service:

```
/** Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/WebServices/WebService.java to edit this template */
package com.unique;
import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;
/*** * @author MyLappy */
@WebService(serviceName = "Calculator")
public class Calculator {
   /*** Web service operation */
   @WebMethod(operationName = "getNumber")
   public int getNumber(@WebParam(name = "num1") int num1, @WebParam(name = "num2") int num2) {
      int sum=num1+num2;
      return sum;
   }
   /**
    * This is a sample web service operation
    */
}
```

Web Service Client:

```
import com.unique.Calculator_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 MyLappy
 */
public class Calculator extends HttpServlet {
```

```java
    @WebServiceRef(wsdlLocation = "WEB-
INF/wsdl/localhost_8080/CalculatorWebService/Calculator.wsdl")
    private Calculator_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()) {
        int num1=Integer.parseInt(request.getParameter("txt1")),
num2=Integer.parseInt(request.getParameter("txt2"));
        out.println("<!DOCTYPE html>");
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Servlet Calculator</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Servlet Calculator at " + getNumber(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);
    }
```
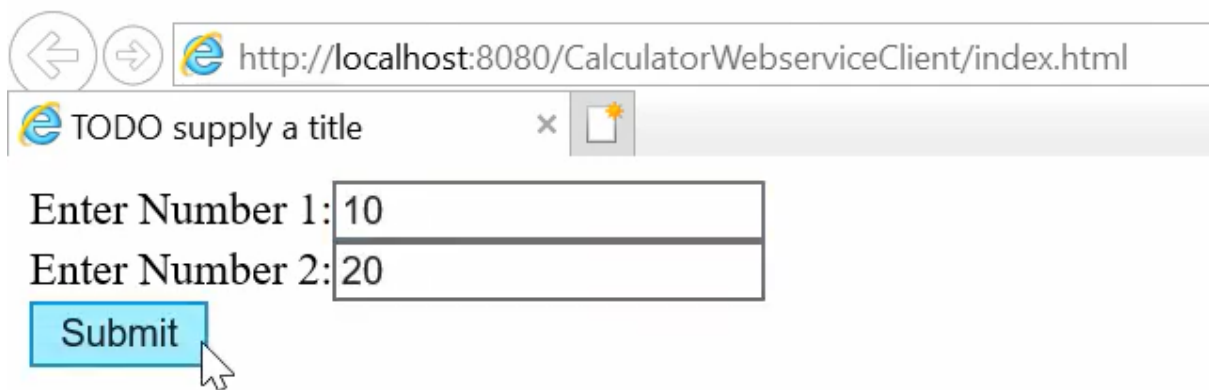
```java
/** * Returns a short description of the servlet. *
 * @return a String containing servlet description
 */
@Override
public String getServletInfo() {
    return "Short description";
}// </editor-fold>
private int getNumber(int num1, int num2) {
    // Note that the injected javax.xml.ws.Service 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.unique.Calculator port = service.getCalculatorPort();
    return port.getNumber(num1, num2);
}
}
```
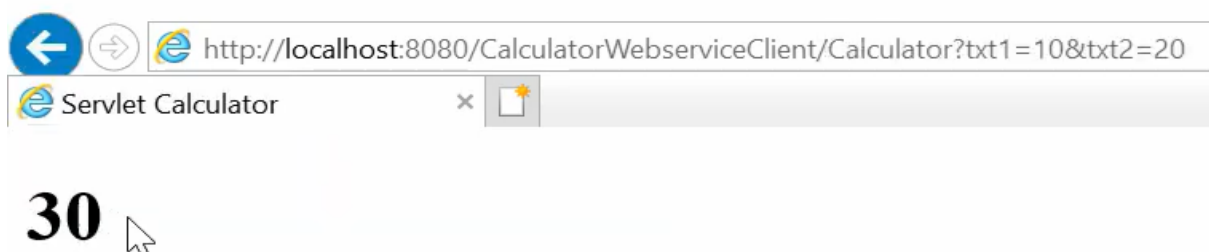
Output: