It uses WSDL i.e. Web Service Definition Language.

Steps :- >

Two Projects: Server
Client

**Server Part**

1. First Go to Files – > New Project or press Ctrl + shift+ N.
2. Now select **Java Web** from categories and **Web Application** from projects
3. Now click **Next.**We are now first going to create server. So give name as **Server(**Preferably). I am going to name as **AparajitServer**

 And now click next and Click Finish.

1. Now **Right click Project (i.e. AparajitServer)**and**New -> Web service. If you cant find in the pop up click other and find there.**
2. Give Web service name and Package name. Remember this name as we will be using it in further areas
3. Click **Finish.**Sometimes you might have a pop box. Click **yes.**

Your Web Service is ready

1. Now we need to add operations for the web service we have created. So **Right Click the web service and Select Add operations.(Ensure you click the createdweb service)**
2. Now add operation. For now I will teach you to do addition alone.
3. Give name of operation. As we are dealing with Integer return type is Integer and Click **Add**and give name as num1 and Type and also create another number as num2 for number 2. (Since addition requires two numbers). Check at left bottom for the operation signature.
4. Now After deploying (It will take few minutes to deploy, be patient) roght click theweb service and click **Test Web Service.**
5. Your browser will be opened and now copy the WSDL and you will be in need of it later.

Client Service Creation:

Steps

1. First Go to Files – > New Project or press Ctrl + shift+ N.

2. Now select **Java Web** from categories and **Web Application** from projects.

3.Now click **Next.**We are now first going to create server. So give name as **Server(**Preferably). I am going to name as **AparajitClient. And click Next and Finish.**

4. Now right click the **Project** and **select** **web service client**. If you cant find in the pop up click other and find there.

5. Now select WSDL and paste the URL that you have copied.( Looks like this :http://localhost:8084/AparajitServer/Papervisions?wsdl)

And then enter the package name of the server. Refer Step 5 In server.

6. Right Click the Project And Create New HTML file with GUI(Form).

Give a name for HTML

Now we are going to design the front end.

Just create two text boxes and label with Number1 and Number2.

Create HTML code.

Ensure that code has action=index.jsp. and now create a new JSP fil(Right clickAparajitClient project and select JSP).

Give name as index.

7. Go to index.jsp page and delete Hello world line.

Call Web Service Operation and select your operation .i.e. add.

You have code automatically generated .

Now num1 and num2 have 0 as value. But we need to get from user and perform this add operation.

8. So delete int num1 and int num2 value and add as follows :

int num1 = Integer.parseInt(request.getParameter(“i1″));
int num2 = Integer.parseInt(request.getParameter(“i2″));;

i1 and i2 is the reference to the HTML document where you gave name for text box.

9. Now you need to edit the java file in server that performs the actual web serviceoperation.

In the operation add, change the return statement as **return num1+num2.**

10. Now run the calc.html file.

//webservice server

package serv;

import javax.jws.WebService;

import javax.jws.WebMethod;

import javax.jws.WebParam;

/\*\*

 \*

 \* @author Admin

 \*/

@WebService(serviceName = "ServerWS")

public class ServerWS {

 /\*\*

 \* This is a sample web service operation

 \*/

 @WebMethod(operationName = "hello")

 public String hello(@WebParam(name = "name") String txt) {

 return "Hello " + txt + " !";

 }

 /\*\*

 \* Web service operation

 \*/

 @WebMethod(operationName = "add")

 public Integer add(@WebParam(name = "num1") int num1, @WebParam(name = "num2") int num2) {

 //TODO write your implementation code here:

 return null;

 }

}



<http://localhost:8084/myServer/ServerWS?wsdl>

<%@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>

 <%-- start web service invocation --%><hr/>

 <%

 try {

 wsClient.ServerWS\_Service service = new wsClient.ServerWS\_Service();

 wsClient.ServerWS port = service.getServerWSPort();

 // TODO initialize WS operation arguments here

 int num1 = 0;

 int num2 = 0;

 // TODO process result here

 java.lang.Integer result = port.add(num1, num2);

 out.println("Result = "+result);

 } catch (Exception ex) {

 // TODO handle custom exceptions here

 }

 %>

 <%-- end web service invocation --%><hr/>

 </body>

</html>

**Replace:**

int num1 = 0;

 int num2 = 0;

With,

int num1 = Integer.parseInt(request.getParameter("i1"));

 int num2 = Integer.parseInt(request.getParameter("i2"));;

**Modify Server Code**:

@WebMethod(operationName = "add")

 public Integer add(@WebParam(name = "num1") int num1, @WebParam(name = "num2") int num2) {

 //TODO write your implementation code here:

 **return num1+num2;**

**//calc.html**

<html>

 <head>

 <title>Calculator</title>

 <meta charset="UTF-8">

 <meta name="viewport" content="width=device-width">

 </head>

 <body>

 <center>

 Calculator<sup>Beta version</sup>

 <hr>

 <br><br><br><br><br>

 <form name ="webservice" action="index.jsp" method="post">

 Number 1 :<input type="text" name="i1" value="" />

 <br>

 Number 2 :<input type="text" name="i2" value="" />

 <br>

 <input type="submit" value="add" name="b1" />

 </center>

 <br><br><br><br><br><br><br>

 <hr>

 </body>

</html>

**//Index.jsp**

 </head>

 <body>

 <%-- start web service invocation --%><hr/>

 <%

 try {

 wsClient.ServerWS\_Service service = new wsClient.ServerWS\_Service();

 wsClient.ServerWS port = service.getServerWSPort();

 // TODO initialize WS operation arguments here

 int num1 = Integer.parseInt(request.getParameter("i1"));

 int num2 = Integer.parseInt(request.getParameter("i2"));;

 // TODO process result here

 java.lang.Integer result = port.add(num1, num2);

 out.println("Result = "+result);

 } catch (Exception ex) {

 // TODO handle custom exceptions here

 }

 %>

 <%-- end web service invocation --%><hr/>

 </body>

</html>

Output:





