

PRACTICAL-4

Aim: Develop client which consumes web services developed in different platform.

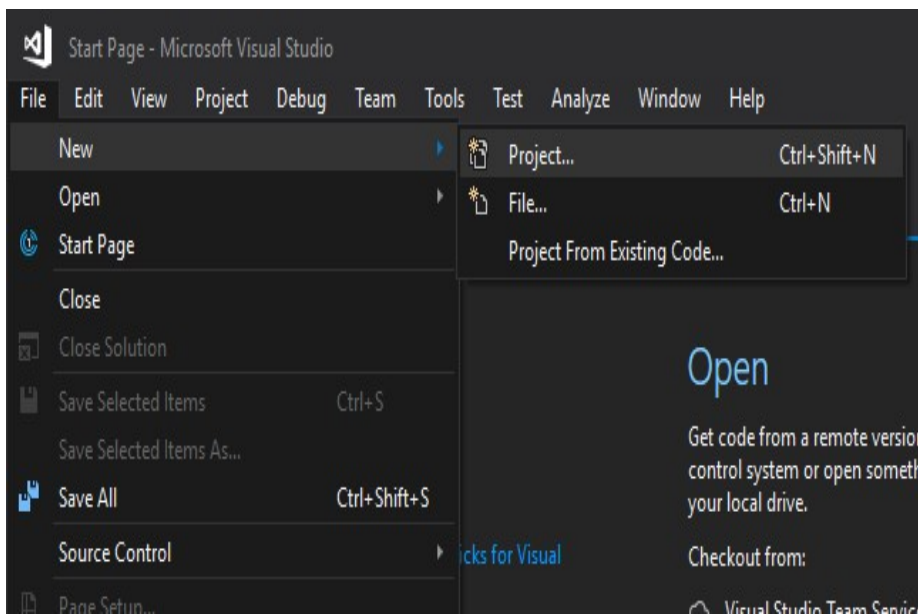
Requirement:

1. Visual Studio Community 2017
2. Version : 15.8 or latest

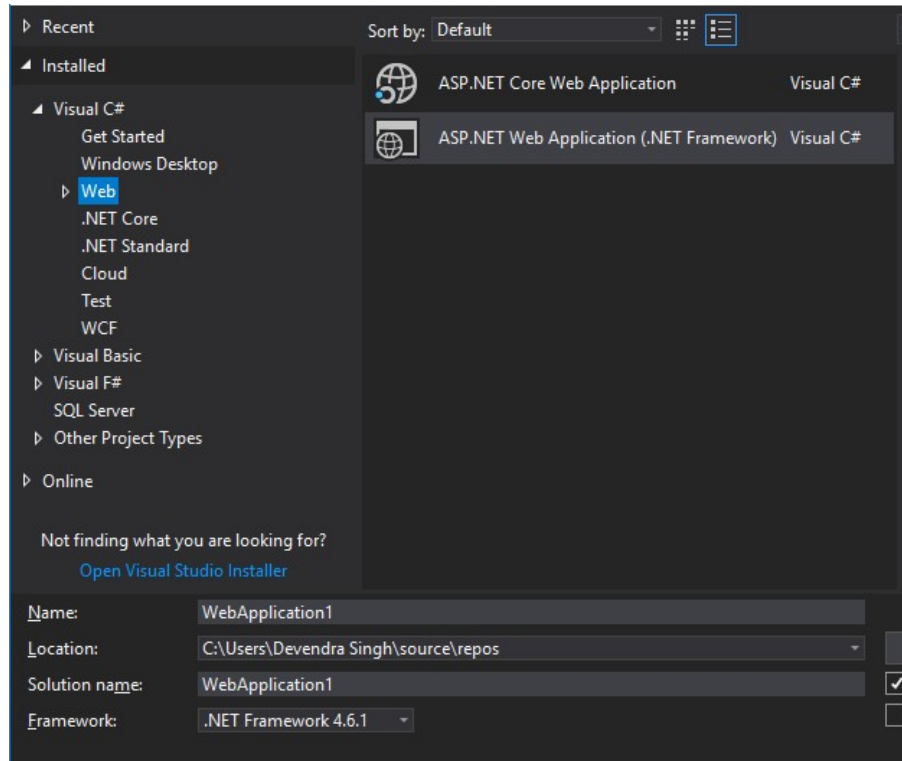
In this practical we are creating Web Service in Visual Studio and then we will consume it in NetBeans.

1. Open Visual Studio IDE and click on File.

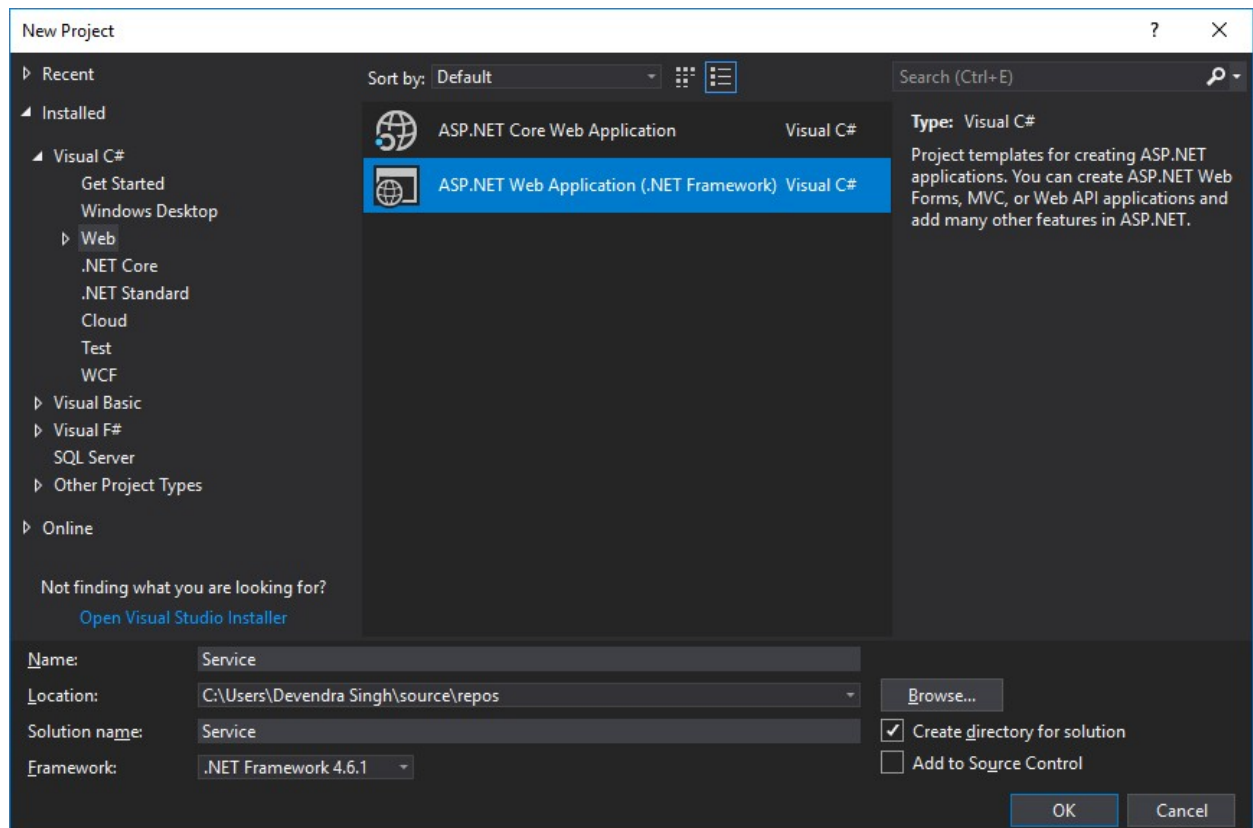
File -> New -> Project



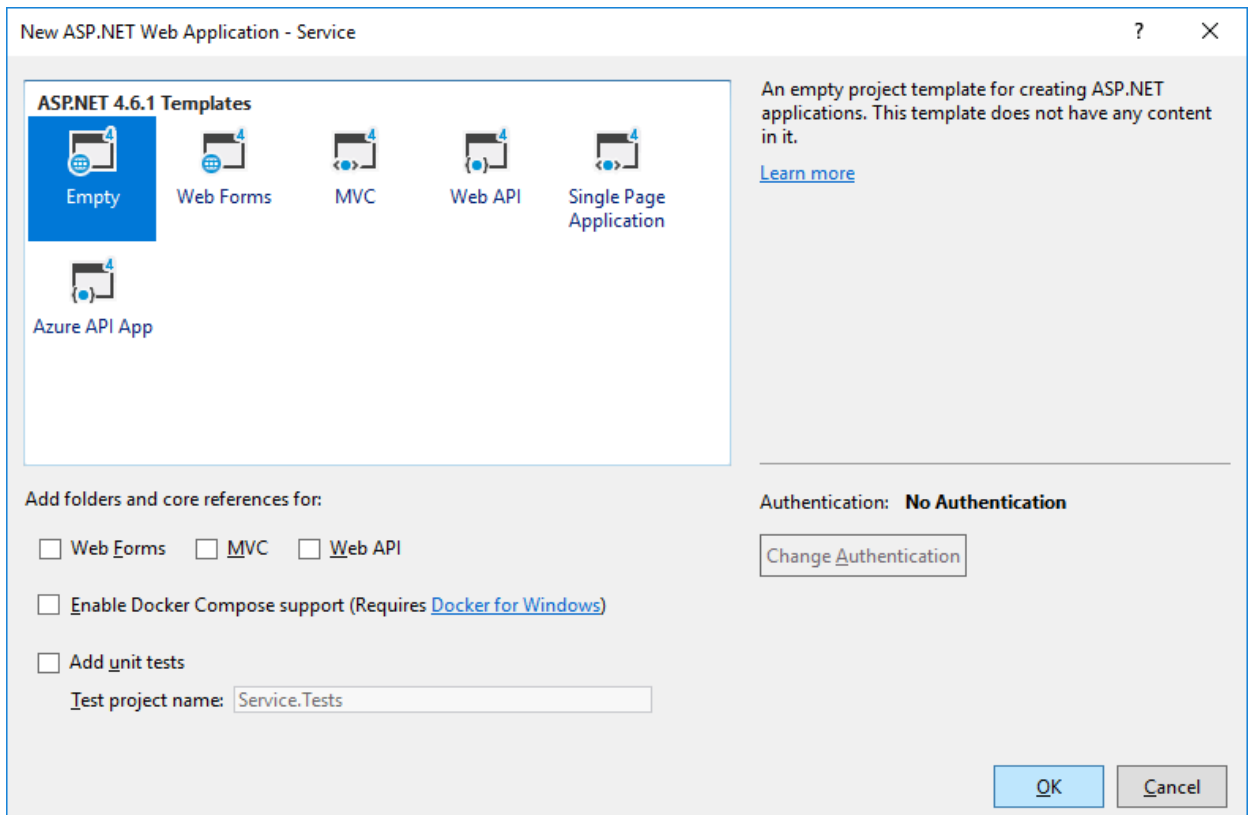
2. Click on Web.



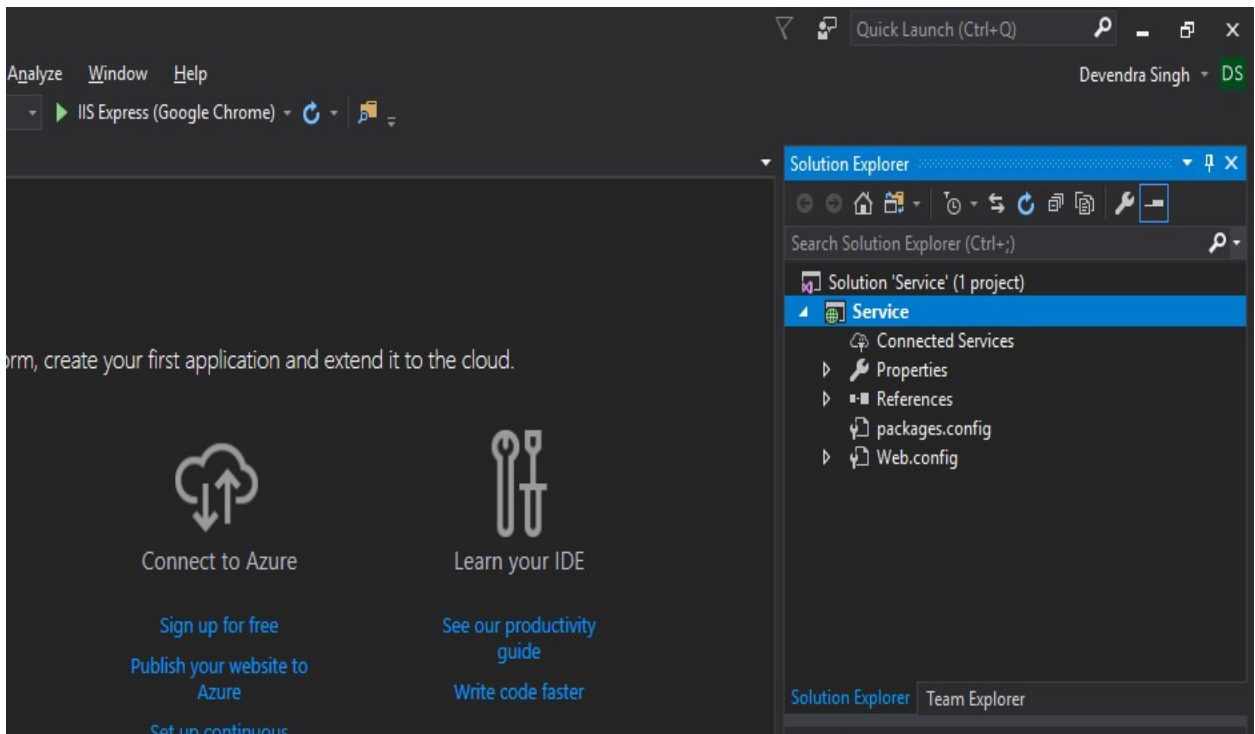
3. Select **ASP.NET Web Application** and give **Name** as **Service**. After that click on **OK** button.



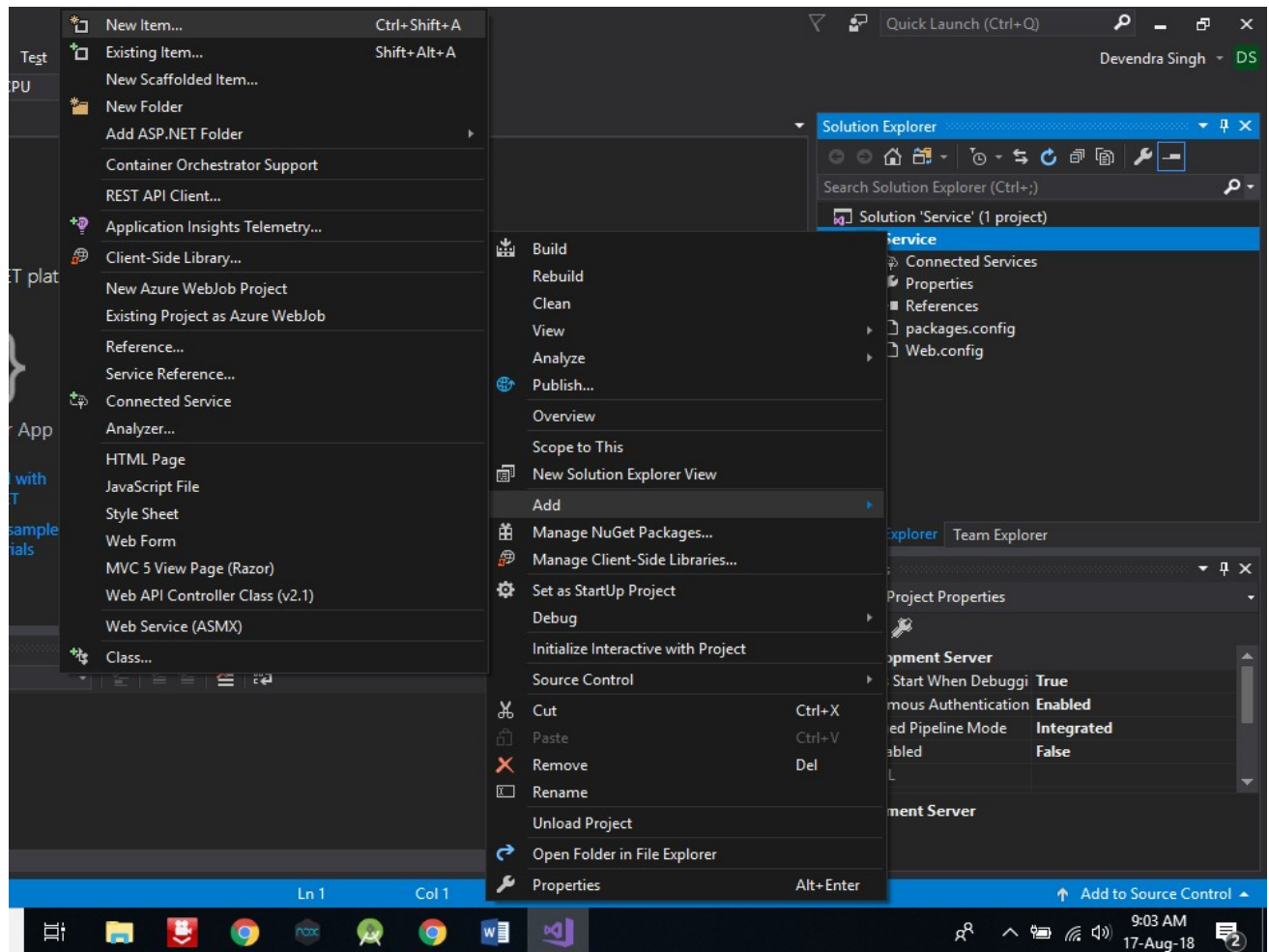
4. Select Empty and click on OK button.



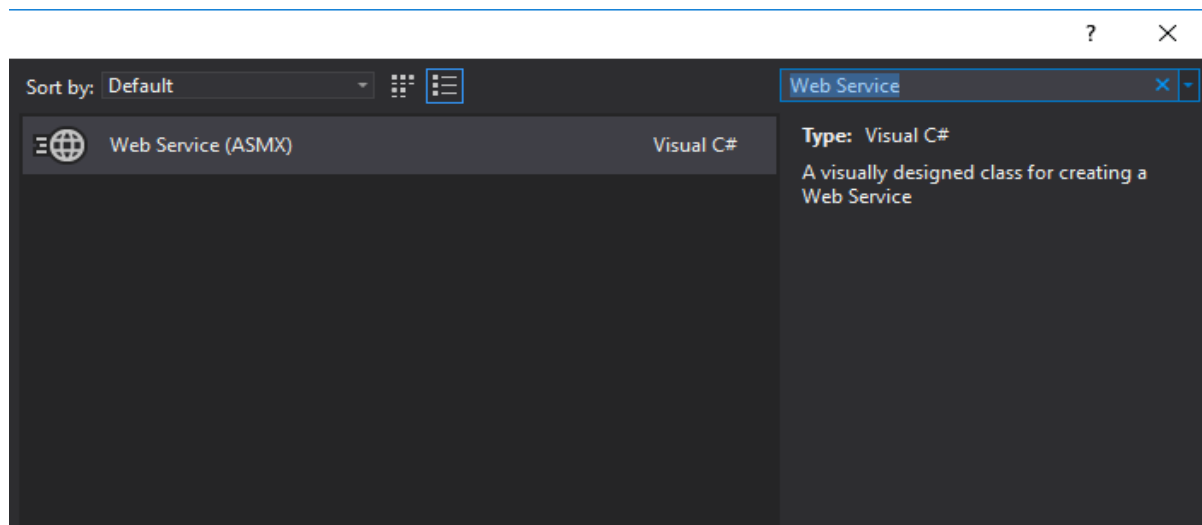
5. Now you can see, on right side in Solution Explorer Service project is created.



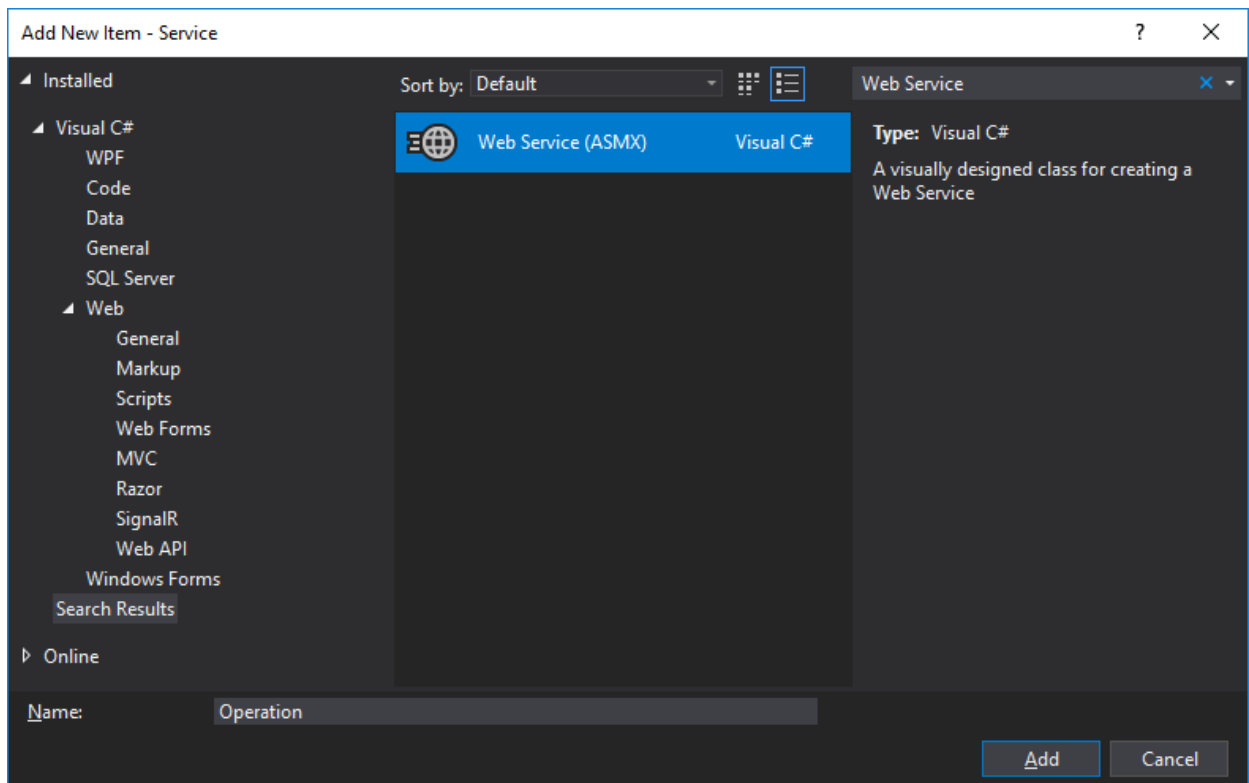
6. Right click on Service -> Add -> New Item.



7. Search for Web Service.



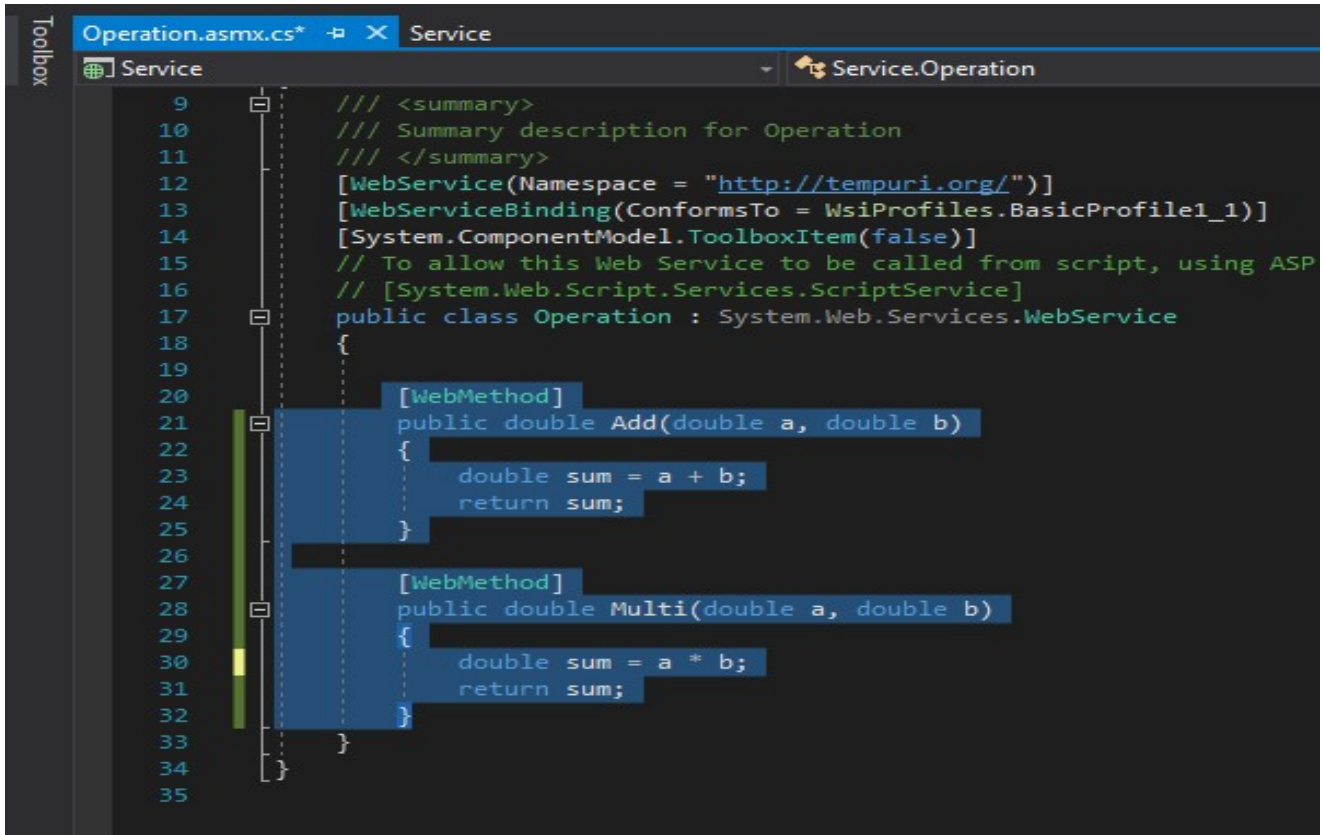
8. Select **Web Service** and give **Name** as **Operation**. After then click on **Add** button.



9. After click on Add button, Operation.asmx.cs file will be automatically open otherwise open it from Solution Explorer and **Add the following code into Class Operation**.

```
[WebMethod]
public double Add(double a, double b)
{
    double sum = a + b;
    return sum;
}
```

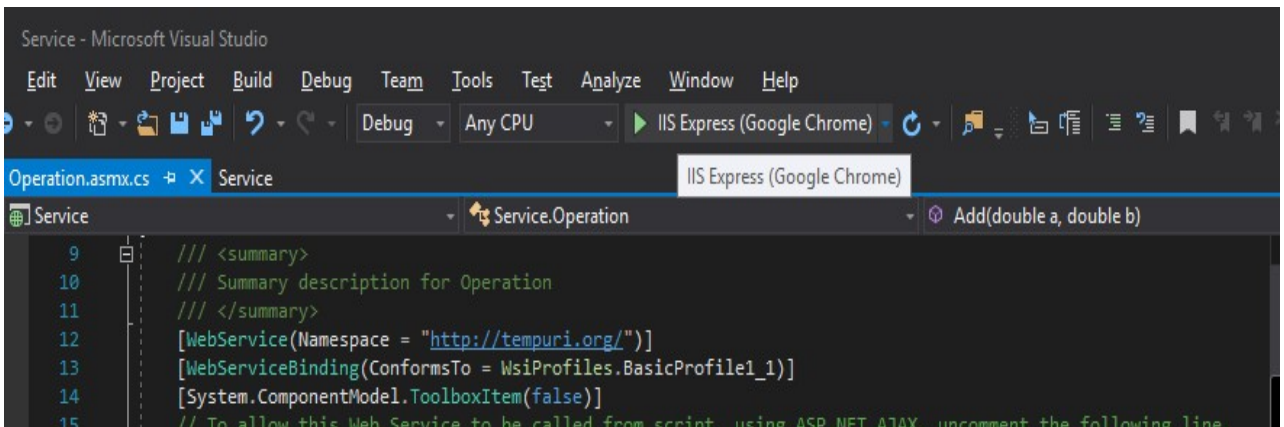
```
[WebMethod]
public double Multi(double a, double b)
{
    double sum = a * b;
    return s
}
```



```
9  /// <summary>
10  /// Summary description for Operation
11  /// </summary>
12  [WebService(Namespace = "http://tempuri.org/")]
13  [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
14  [System.ComponentModel.ToolboxItem(false)]
15  // To allow this Web Service to be called from script, using ASP
16  // [System.Web.Script.Services.ScriptService]
17  public class Operation : System.Web.Services.WebService
18  {
19
20      [WebMethod]
21      public double Add(double a, double b)
22      {
23          double sum = a + b;
24          return sum;
25      }
26
27      [WebMethod]
28      public double Multi(double a, double b)
29      {
30          double sum = a * b;
31          return sum;
32      }
33  }
34
35
```

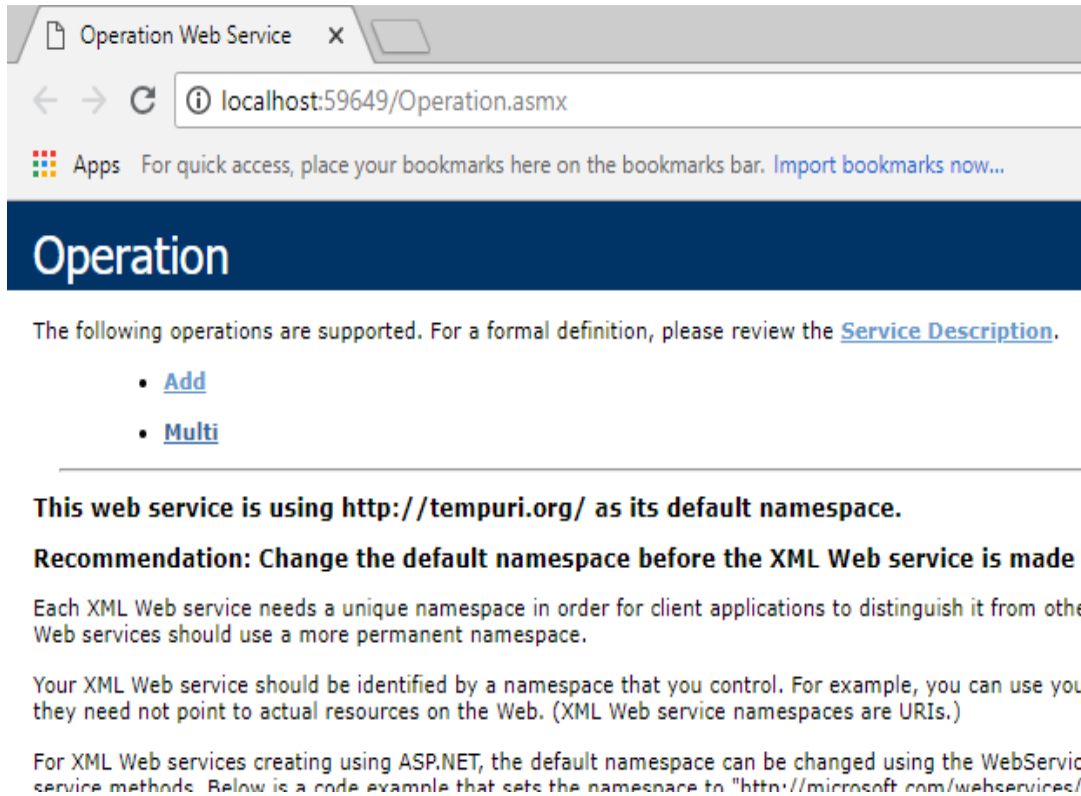
After that press **Ctrl+S** to save the methods. Actually we are creating two methods for Web Service. One is for addition of two numbers and second one is for multiplication of two numbers.

10. Now run the project by click on **Green arrow button** below the **Window** menu.



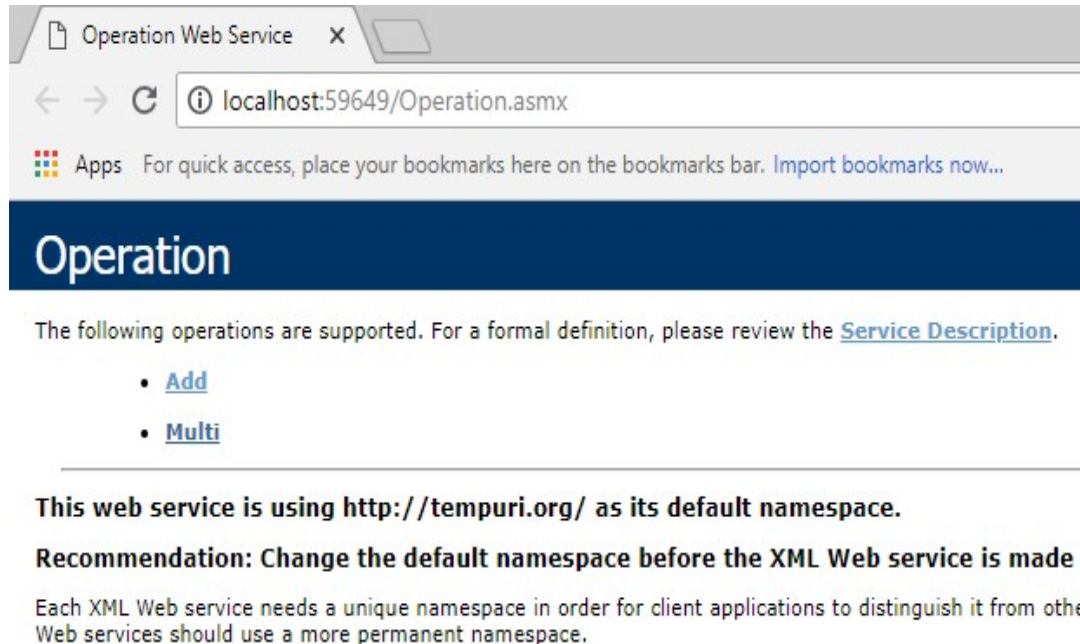
```
Service - Microsoft Visual Studio
Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express (Google Chrome)
Operation.asmx.cs Service IIS Express (Google Chrome)
Service Service.Operation Add(double a, double b)
9  /// <summary>
10  /// Summary description for Operation
11  /// </summary>
12  [WebService(Namespace = "http://tempuri.org/")]
13  [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
14  [System.ComponentModel.ToolboxItem(false)]
15  // To allow this Web Service to be called from script, using ASP.NET AJAX, uncomment the following line.
```

11. A window will open in browser and that is our web service.

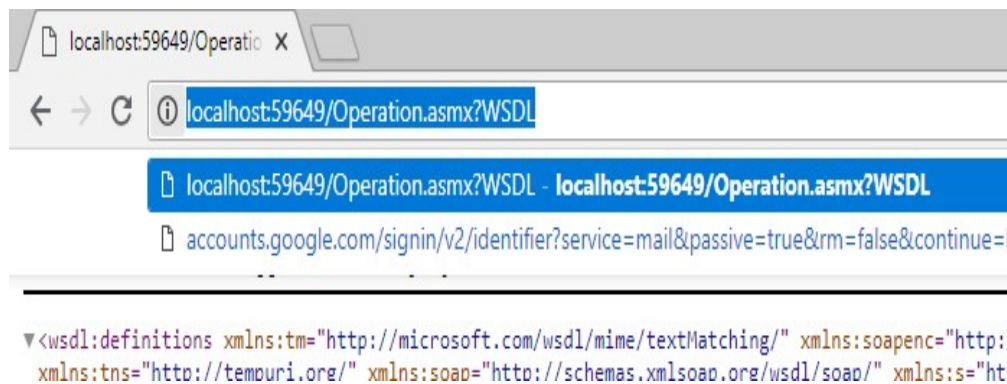


12. You can check services by click on Add or Multi option. But we don't need this.

13. Now click on Service Description option.



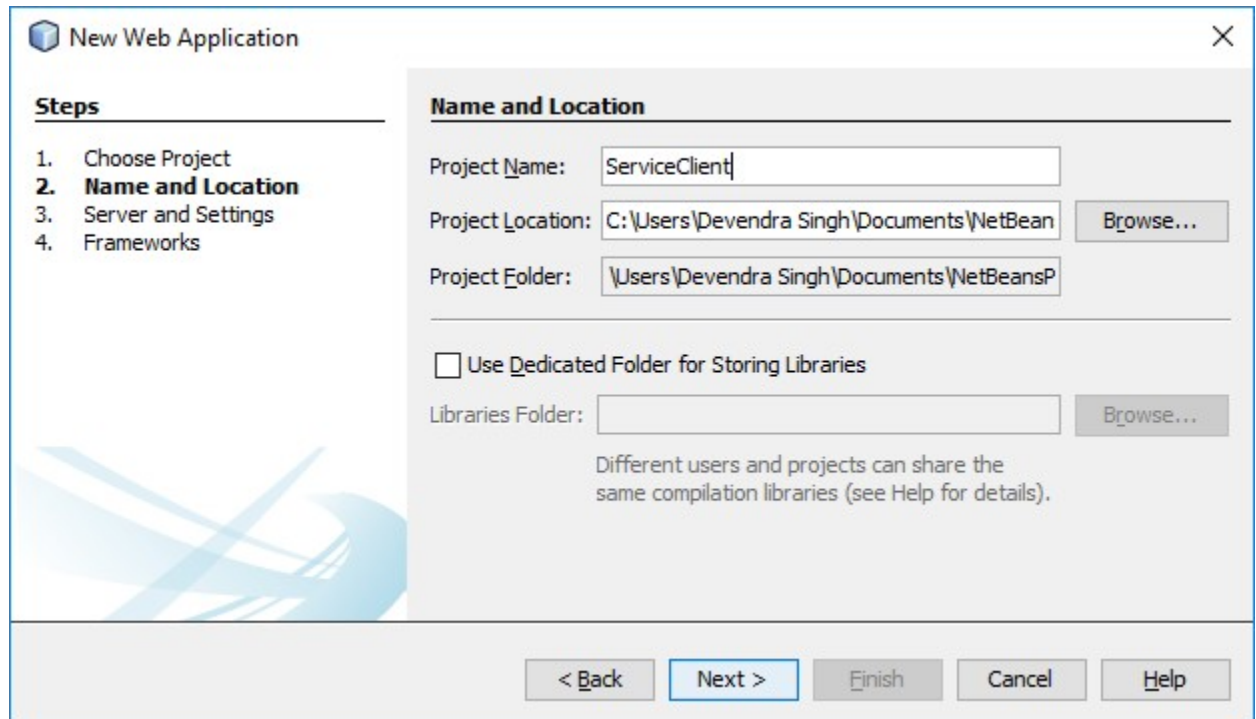
14. Now a new window will open. Select the link and copy it. We will use this link in NetBeans to consume these services.



15. Don't close Visual Studio and browser, just minimize it otherwise server will stop. But save the link anywhere, so that we can use it later.

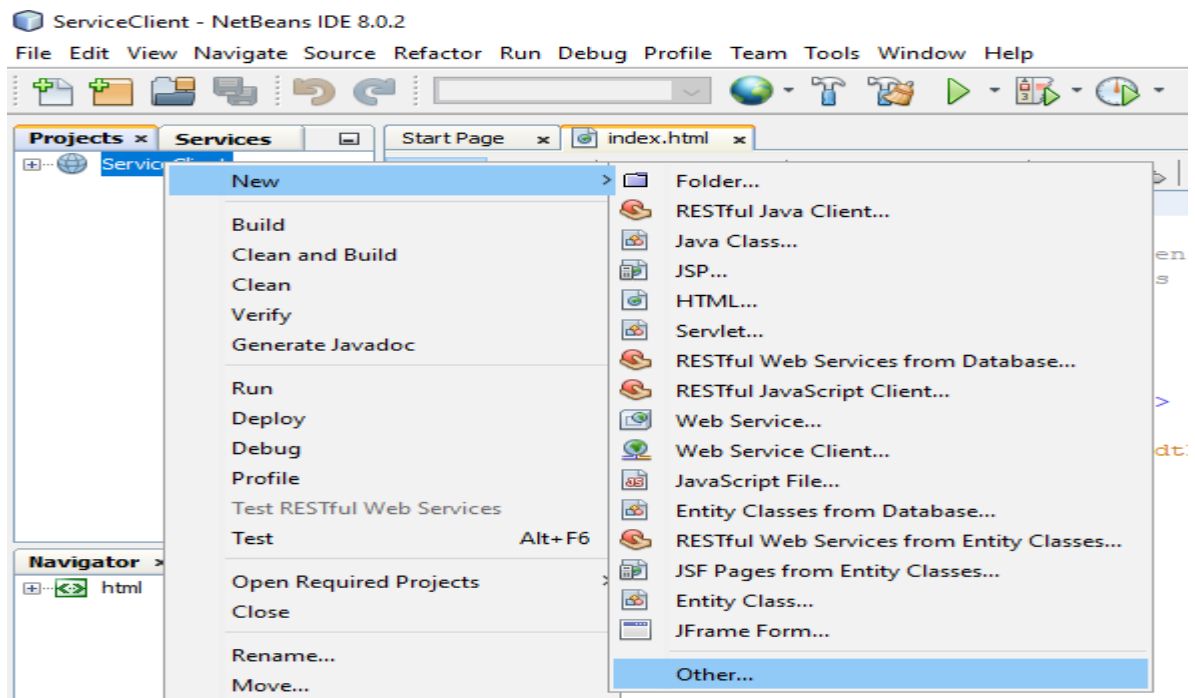
16. Now open NetBeans.

17. Create a Web Application with name ServiceClient. Next -> Finish.

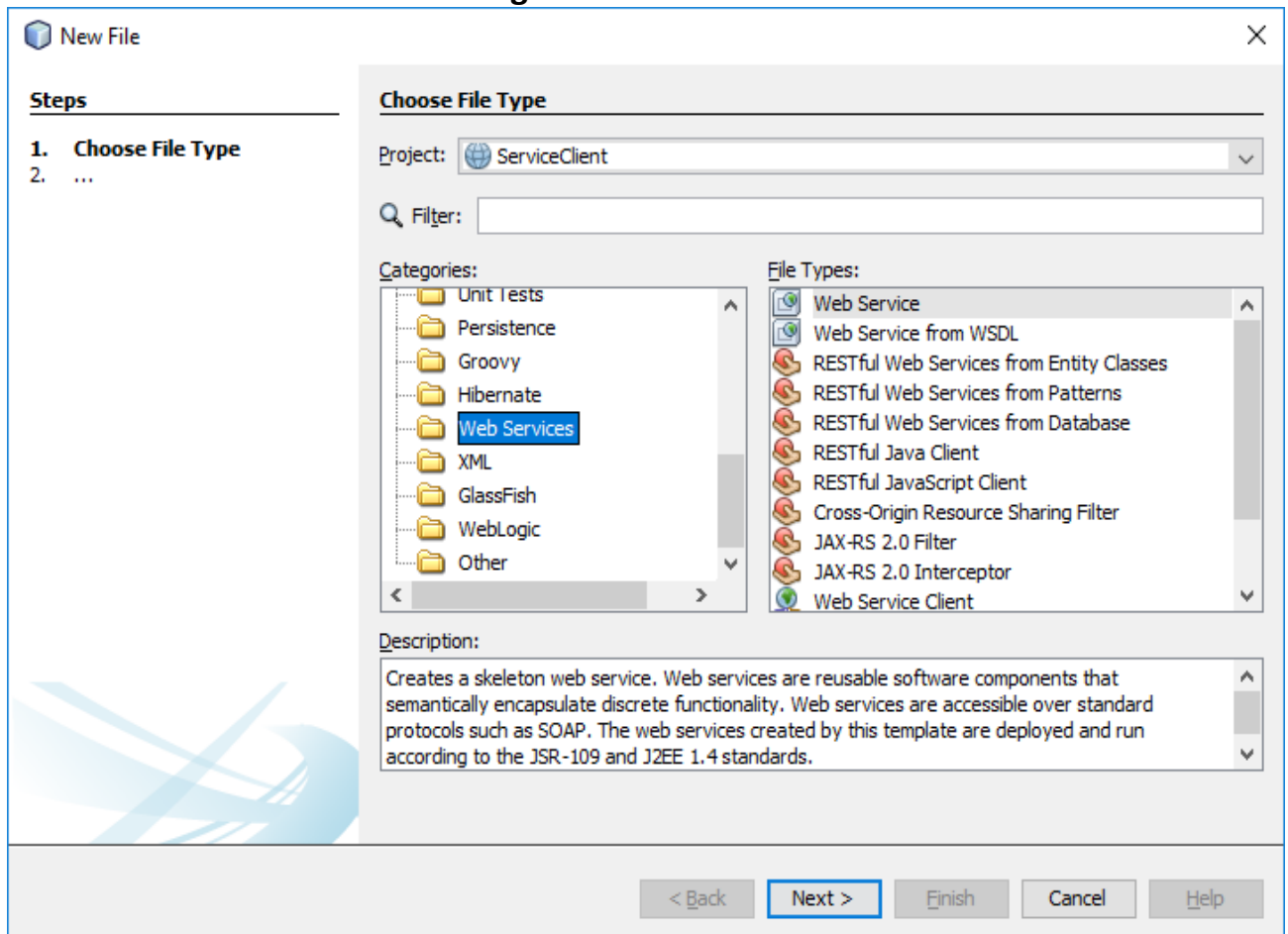


18. Now create a Web Service Client.

Right click on ServiceClient -> New -> Other.

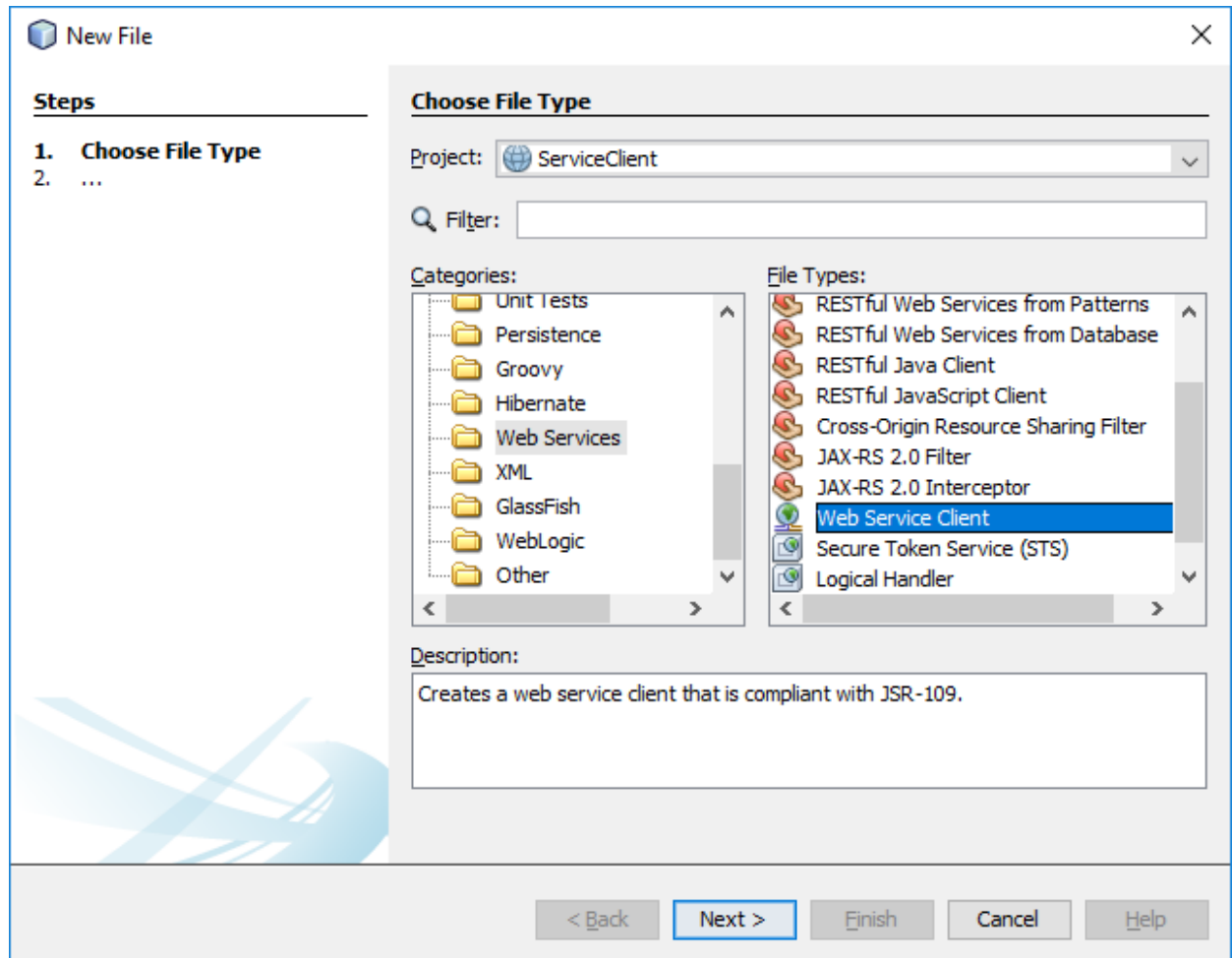


19. Now select **Web Services** in **Categories** section.

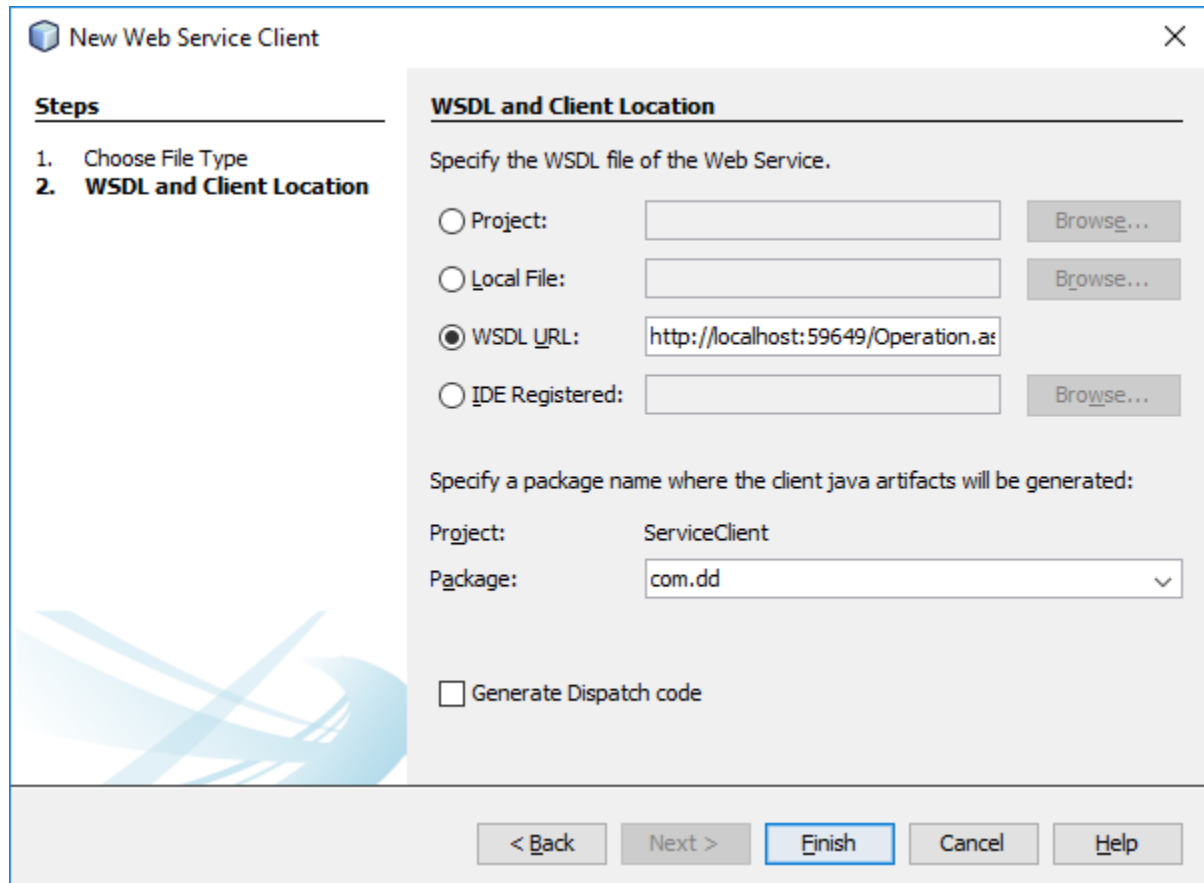


20. Select **Web Service Client** in **File Types** and Click on **Next** button.

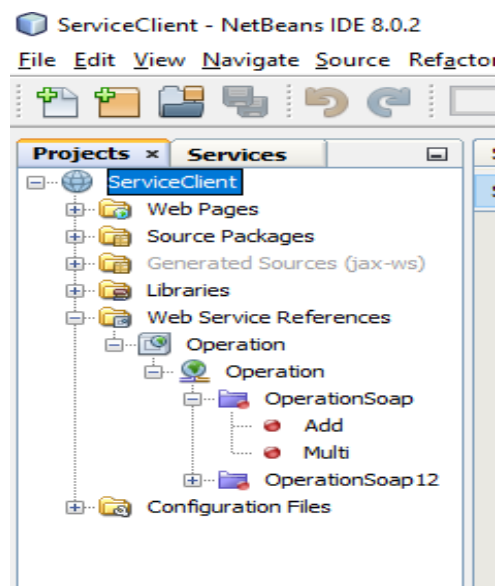
GAURI



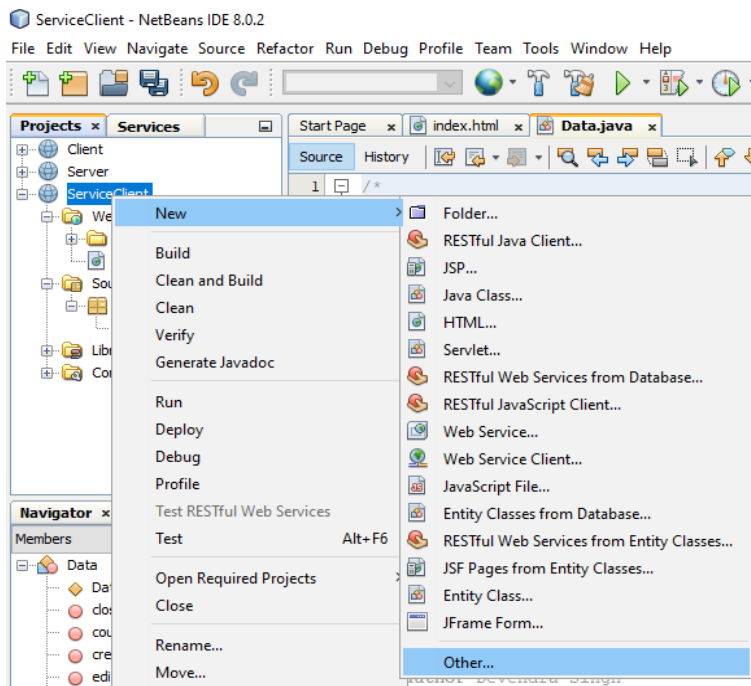
21. Select WSDL URL and paste the link that you have copied from browser on run of Visual Studio enter package name com.dd. After that click on Finish button.



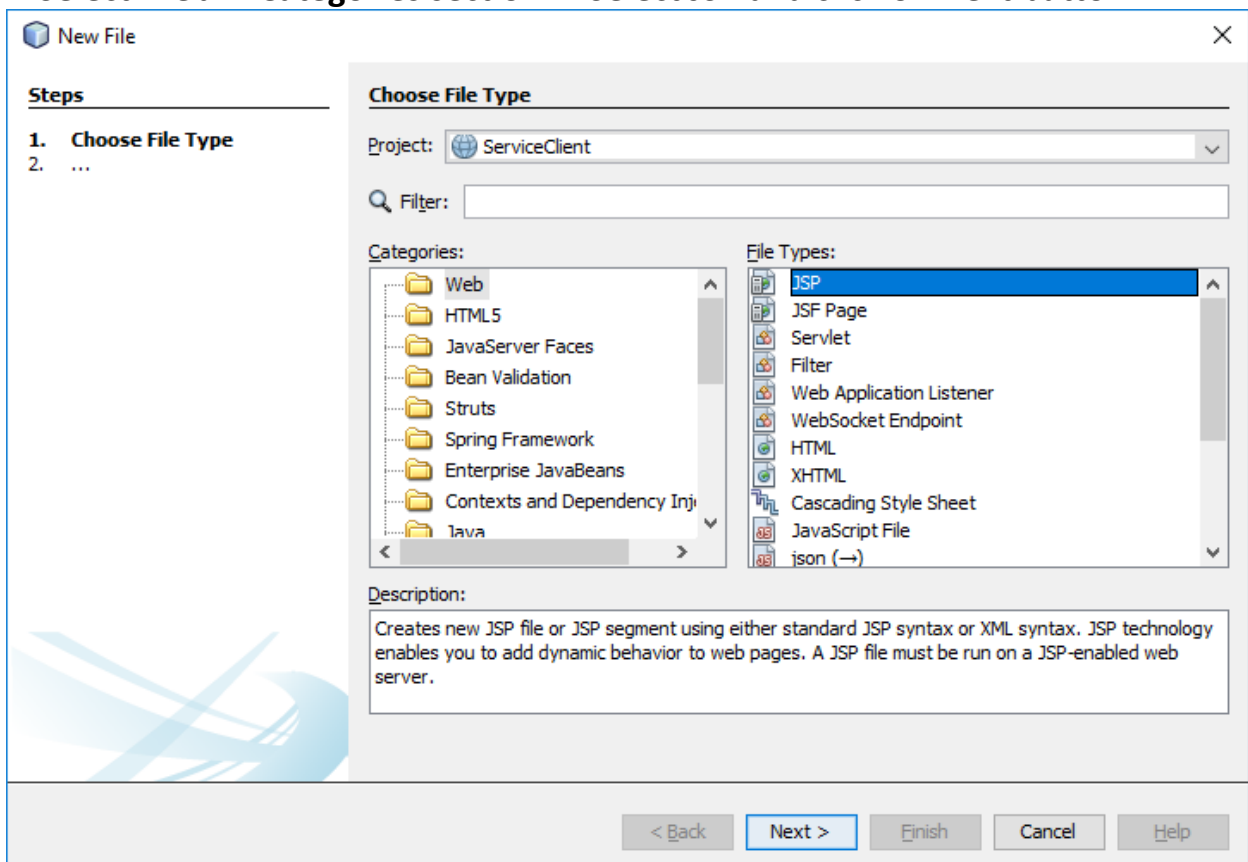
22.As you can see, we got both the service methods i.e. Add & Multi. But in this practical I'm going to use only one service method. You can do for both also.



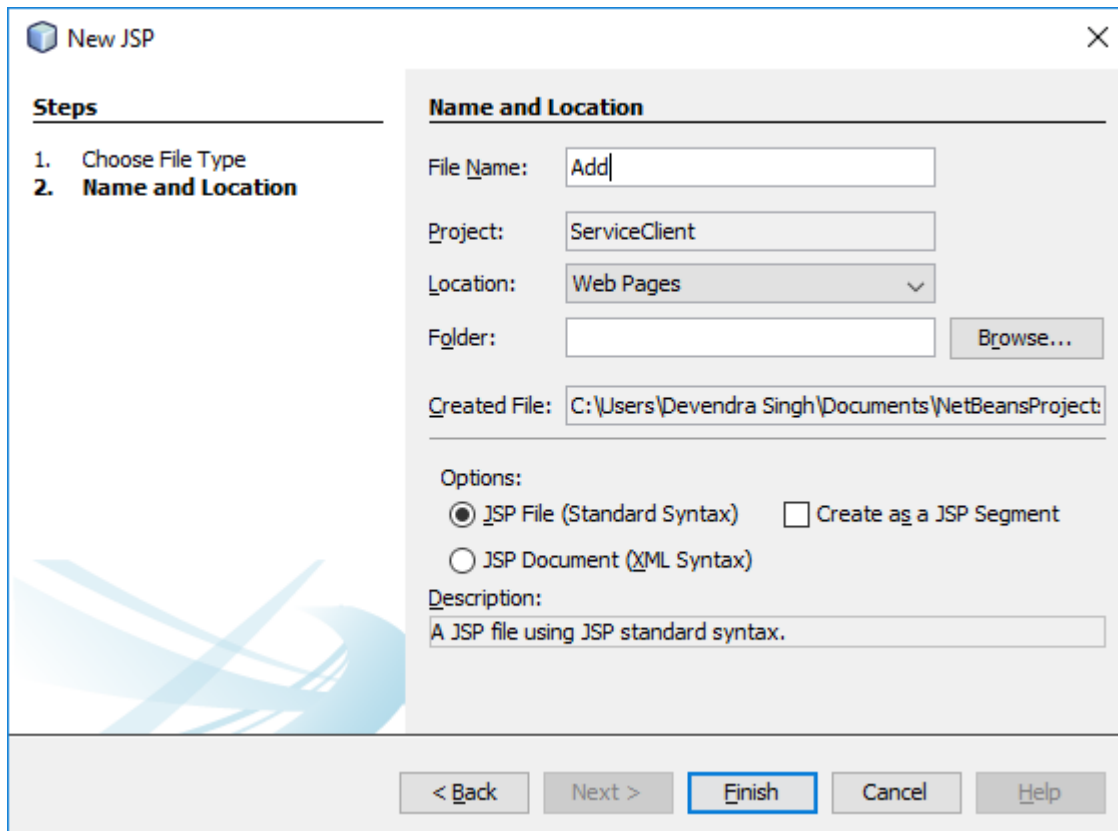
23.Now create a JSP page. Right click on ServiceClient -> New -> Other



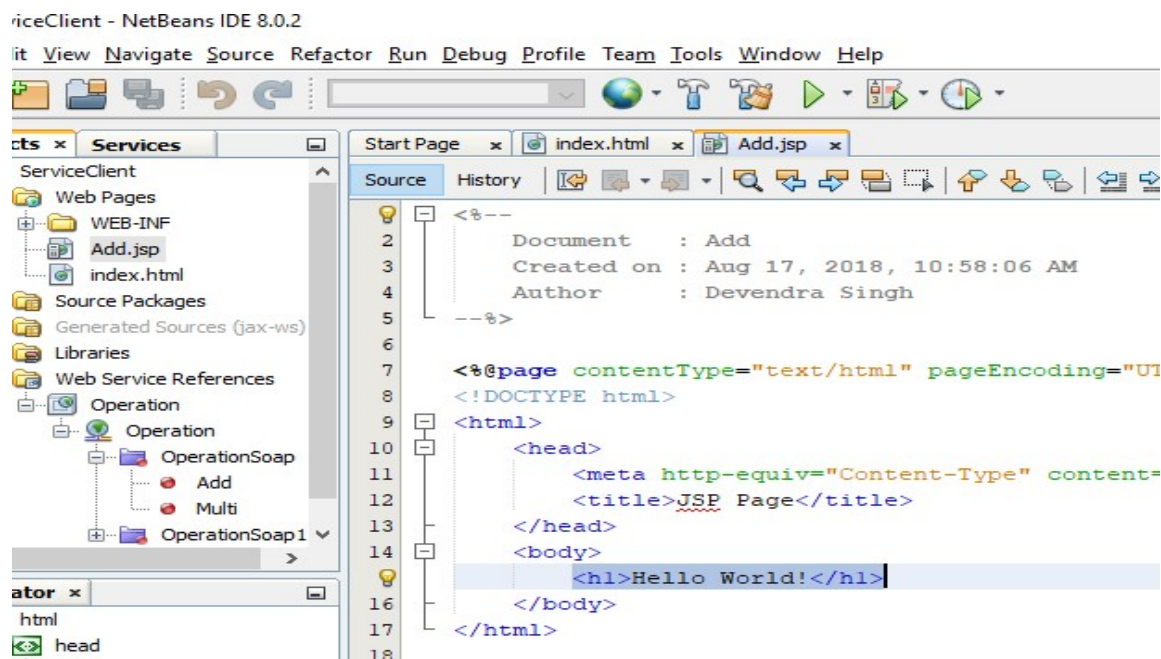
24. Select Web in Categories section -> Select JSP and click on Next button.



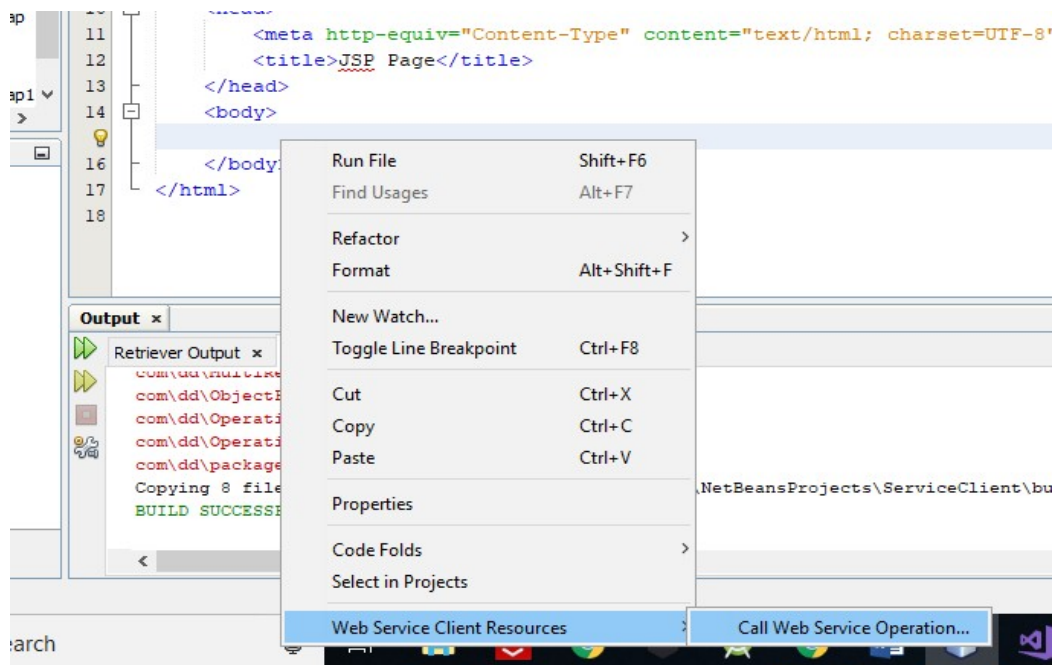
25. Enter File Name **Add** and click on **Finish** button.



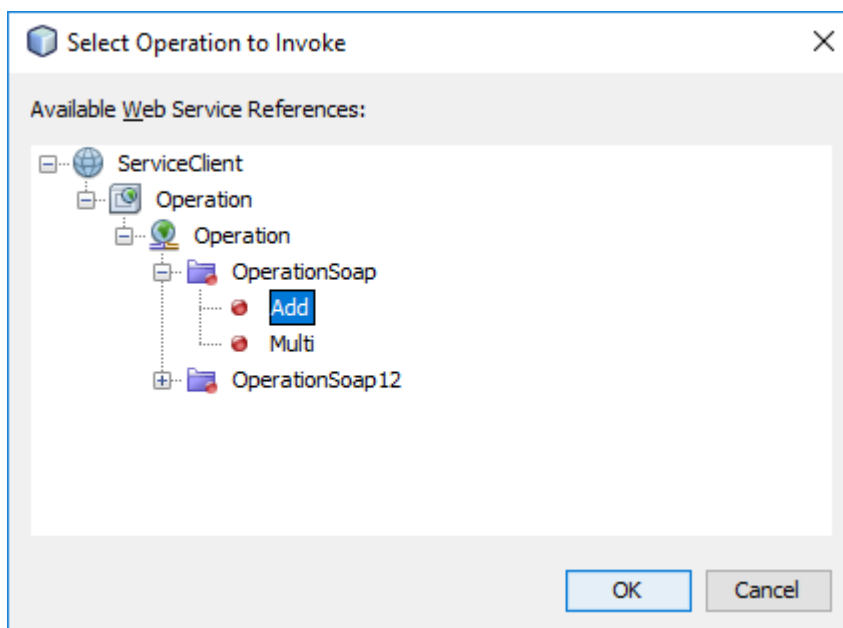
26. In **Add.jsp** file delete the selected part in body tag, because we don't need this.



27. Right click between body tag and select **Call Web Service Operation**.

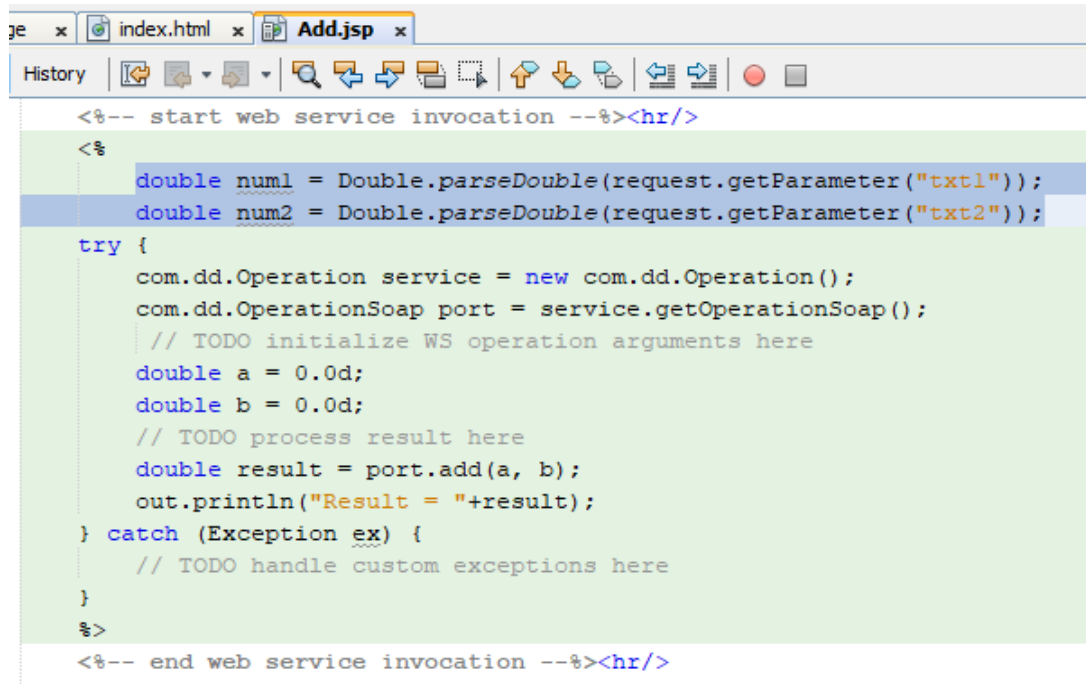


28. Expand and select **Add**. After select, click on **OK** button.



29. Now add the following code outside of try block.

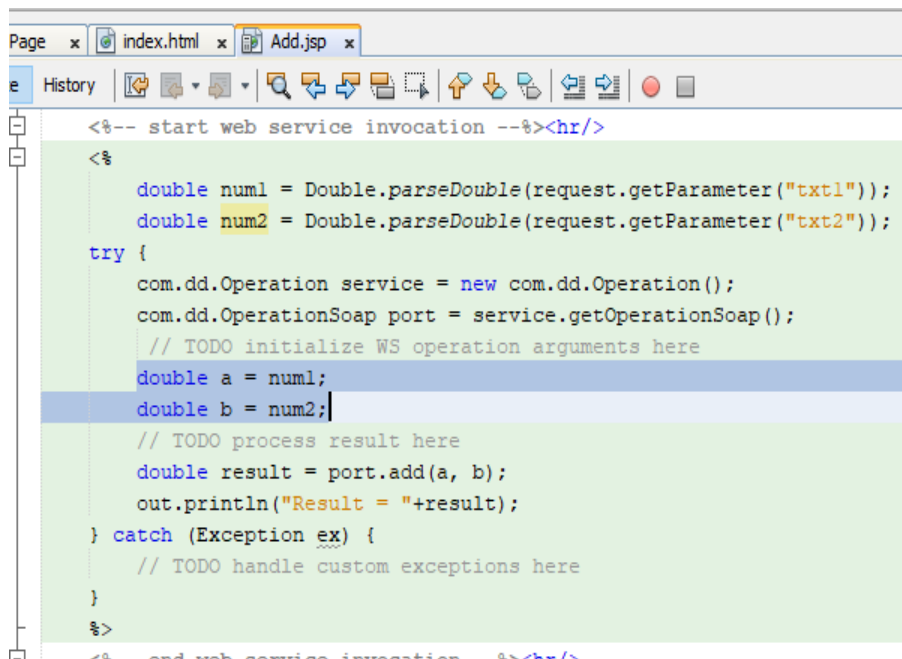
```
double num1 = Double.parseDouble(request.getParameter("txt1"));
double num2 = Double.parseDouble(request.getParameter("txt2"));
```



```
<!-- start web service invocation --><hr/>
<%
    double num1 = Double.parseDouble(request.getParameter("txt1"));
    double num2 = Double.parseDouble(request.getParameter("txt2"));

    try {
        com.dd.Operation service = new com.dd.Operation();
        com.dd.OperationSoap port = service.getOperationSoap();
        // TODO initialize WS operation arguments here
        double a = 0.0d;
        double b = 0.0d;
        // TODO process result here
        double result = port.add(a, b);
        out.println("Result = "+result);
    } catch (Exception ex) {
        // TODO handle custom exceptions here
    }
%>
<!-- end web service invocation --><hr/>
```

30. Now pass num1 & num2 to a & b variable respectively. After that press Ctrl+S to save this code.



```
<!-- start web service invocation --><hr/>
<%
    double num1 = Double.parseDouble(request.getParameter("txt1"));
    double num2 = Double.parseDouble(request.getParameter("txt2"));

    try {
        com.dd.Operation service = new com.dd.Operation();
        com.dd.OperationSoap port = service.getOperationSoap();
        // TODO initialize WS operation arguments here
        double a = num1;
        double b = num2;

        // TODO process result here
        double result = port.add(a, b);
        out.println("Result = "+result);
    } catch (Exception ex) {
        // TODO handle custom exceptions here
    }
%>
<!-- end web service invocation --><hr/>
```

31. Now open index.html file of ServiceClient project and replace the contents of body tag with following code. After that press Ctrl+S to save it.

<form>

**<input type="text" name="txt1" placeholder="Enter First Number">

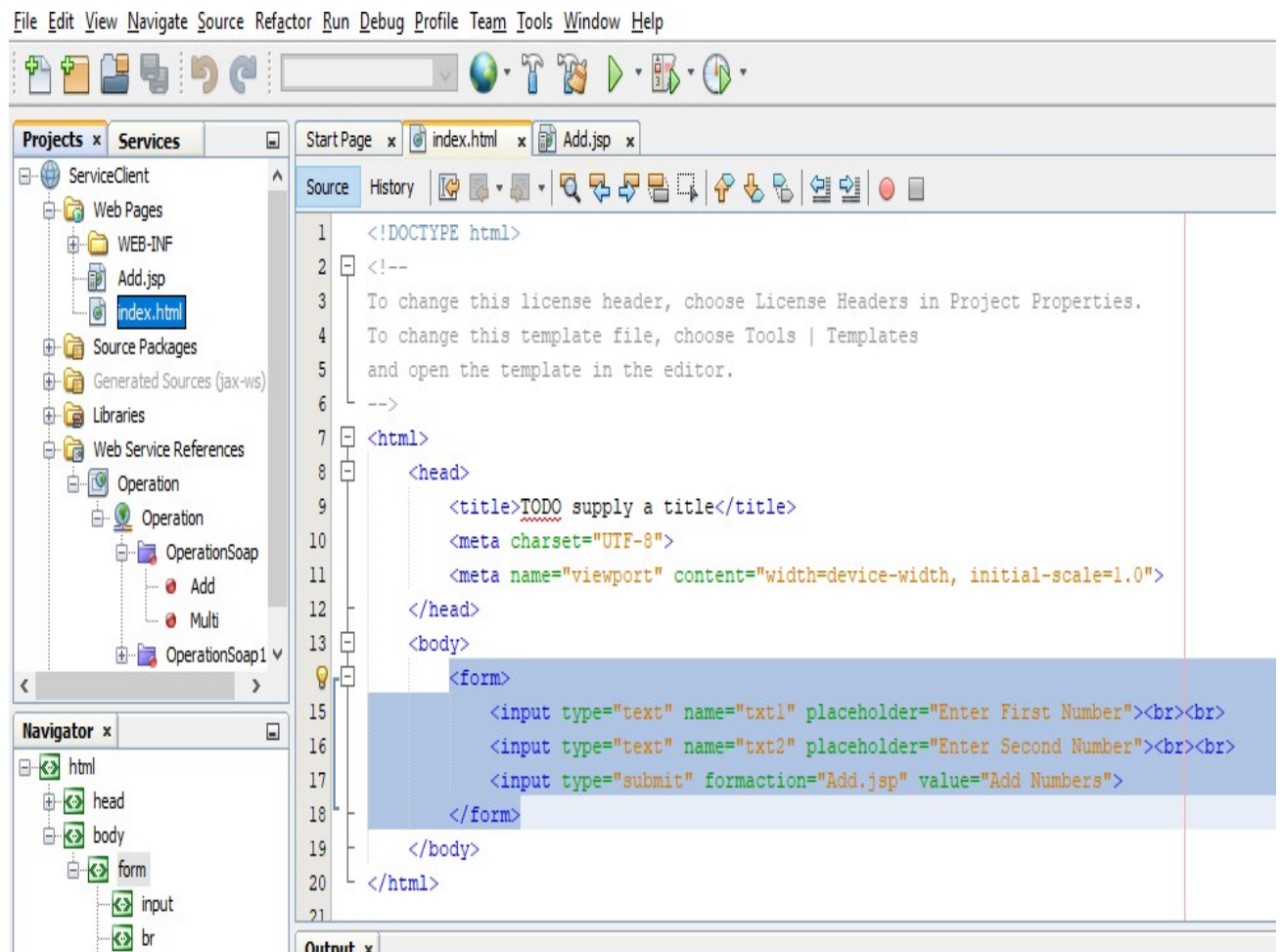
**

**<input type="text" name="txt2" placeholder="Enter Second Number">

**

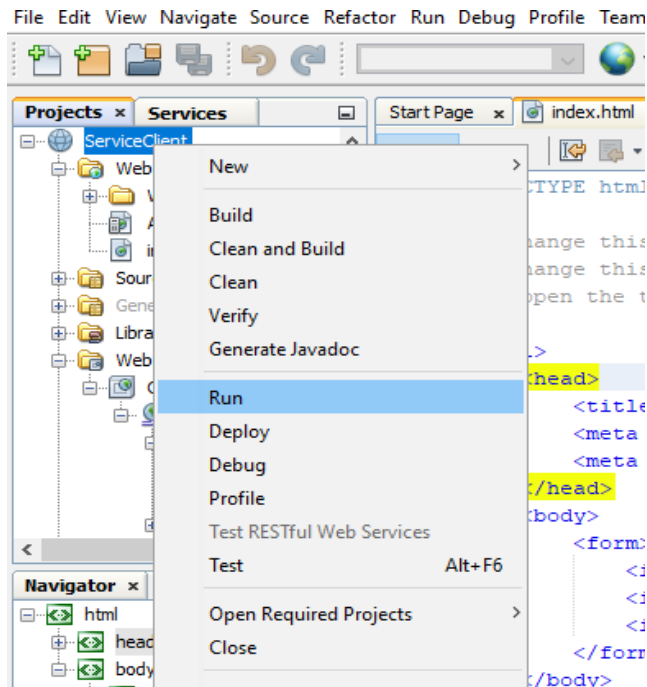
<input type="submit" formaction="Add.jsp" value="Add Numbers">

</form>

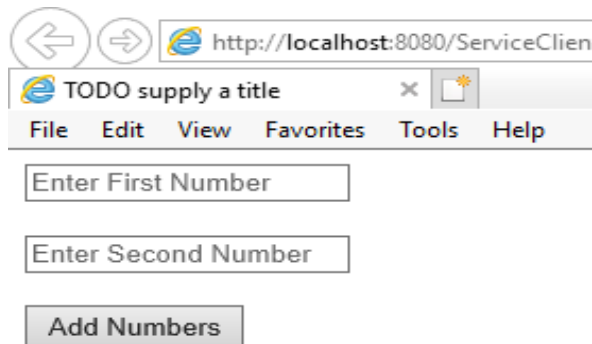


32. Now run the ServerClient web application.

GAURI



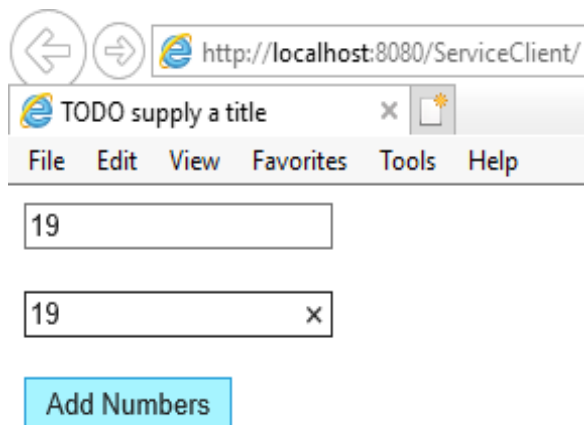
33. A window will open in browser as below.



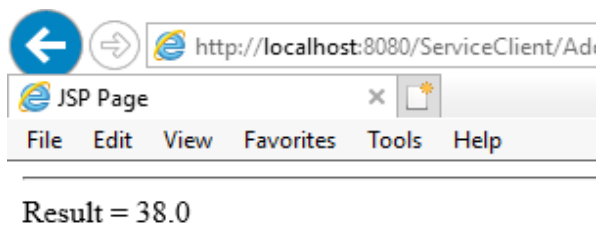
34. Now enter two numbers and click on Add Numbers button. Wait to get result.....

You will get result on a new page.

Mahesh



A screenshot of a web browser window. The address bar shows the URL `http://localhost:8080/ServiceClient/`. The browser has a single tab titled "TODO supply a title". The menu bar includes "File", "Edit", "View", "Favorites", "Tools", and "Help". The page content consists of two text input fields, each containing the number "19", and a light blue button labeled "Add Numbers".



A screenshot of a web browser window. The address bar shows the URL `http://localhost:8080/ServiceClient/AddNumbers.jsp`. The browser has a single tab titled "JSP Page". The menu bar includes "File", "Edit", "View", "Favorites", "Tools", and "Help". The page content displays the text "Result = 38.0".