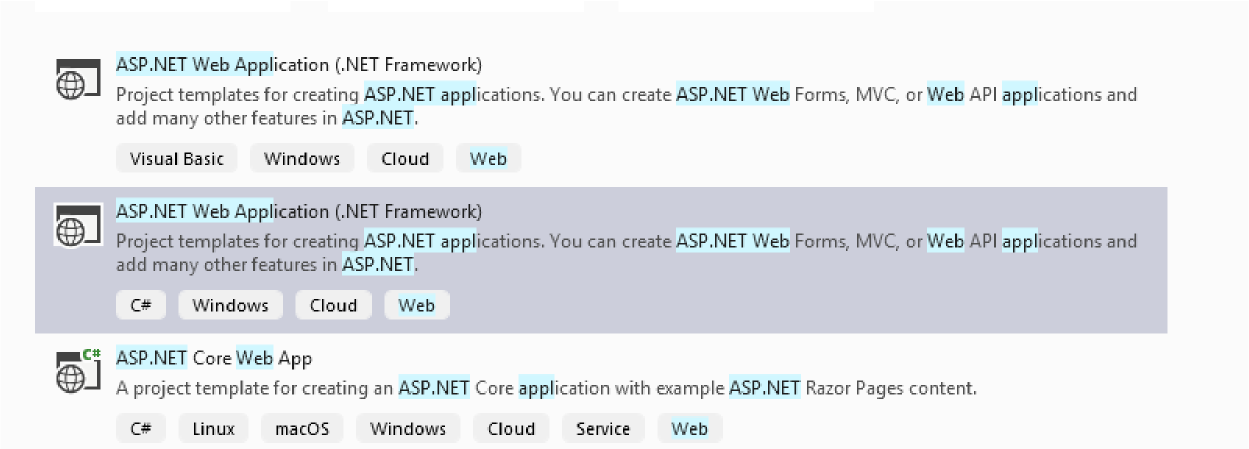
Practical:2

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

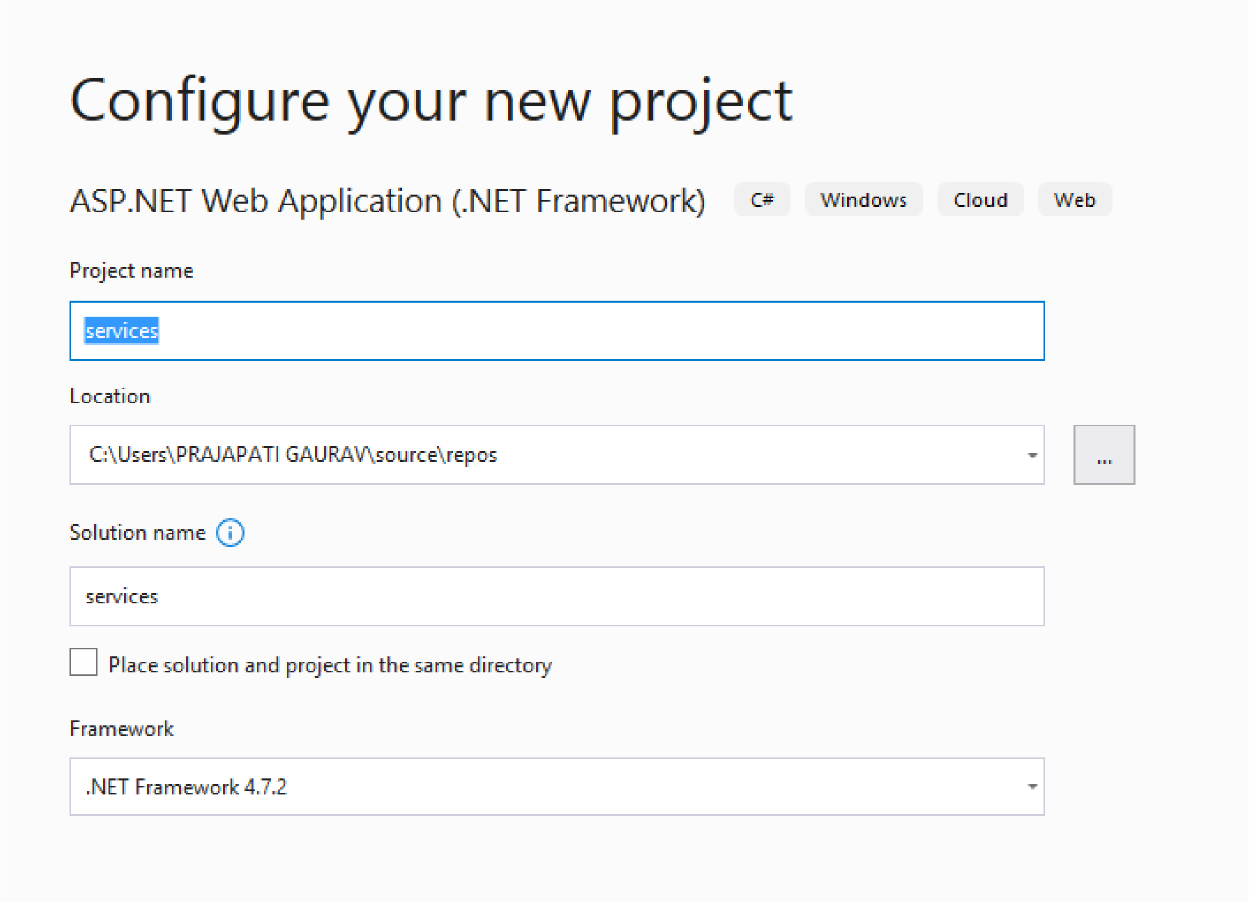
Create project in visual studio and named its services



ASP.NET Web Application (.NET Framework)
Project templates for creating ASP.NET applications. You can create ASP.NET Web Forms, MVC, or Web API applications and add many other features in ASP.NET.

ASP.NET Web Application (.NET Framework)
Project templates for creating ASP.NET applications. You can create ASP.NET Web Forms, MVC, or Web API applications and add many other features in ASP.NET.

ASP.NET Core Web App
A project template for creating an ASP.NET Core application with example ASP.NET Razor Pages content.



Configure your new project

ASP.NET Web Application (.NET Framework)

Project name: services

Location: C:\Users\PRAJAPATI GAURAV\source\repos

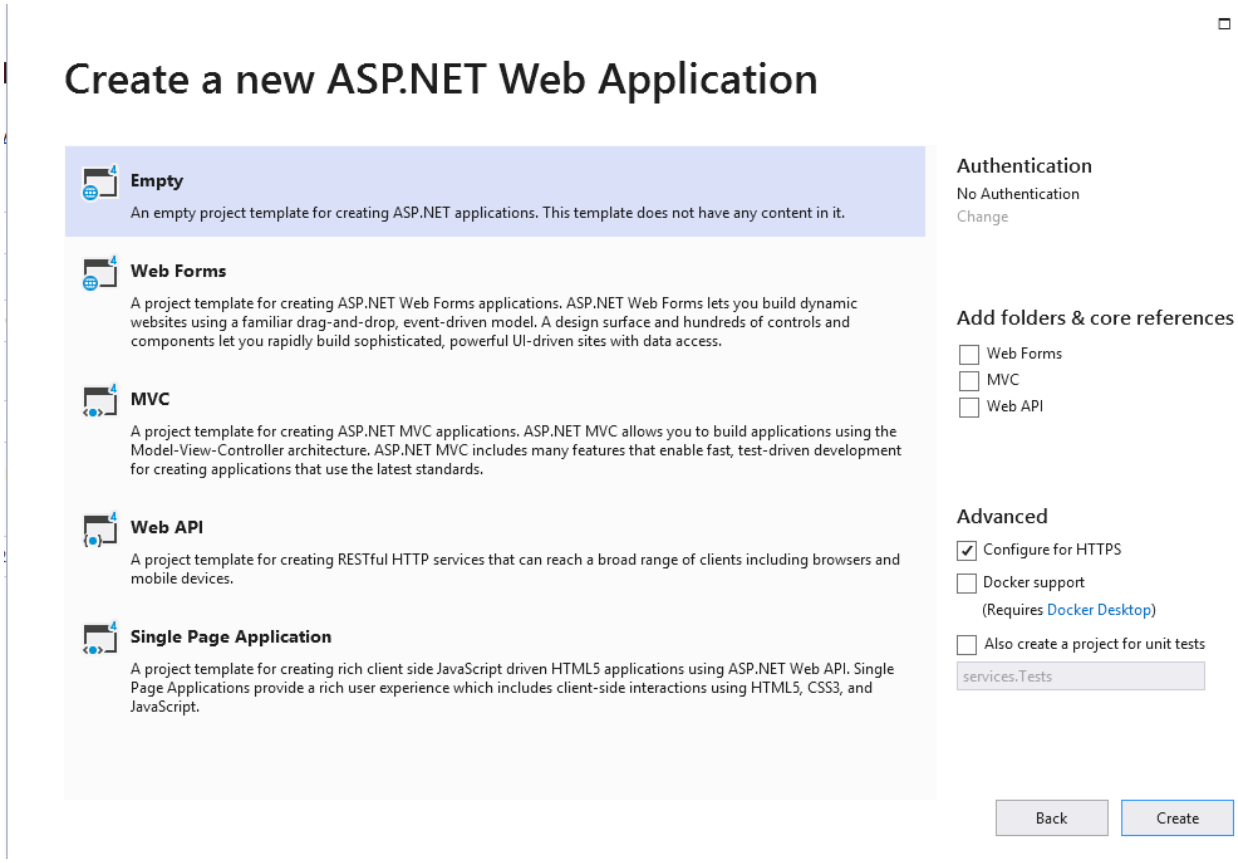
Solution name: services

□ Place solution and project in the same directory

Framework: .NET Framework 4.7.2

Nikhil Singh TYCS47s

select empty and click on create



Create a new ASP.NET Web Application

Empty
An empty project template for creating ASP.NET applications. This template does not have any content in it.

Web Forms
A project template for creating ASP.NET Web Forms applications. ASP.NET Web Forms lets you build dynamic websites using a familiar drag-and-drop, event-driven model. A design surface and hundreds of controls and components let you rapidly build sophisticated, powerful UI-driven sites with data access.

MVC
A project template for creating ASP.NET MVC applications. ASP.NET MVC allows you to build applications using the Model-View-Controller architecture. ASP.NET MVC includes many features that enable fast, test-driven development for creating applications that use the latest standards.

Web API
A project template for creating RESTful HTTP services that can reach a broad range of clients including browsers and mobile devices.

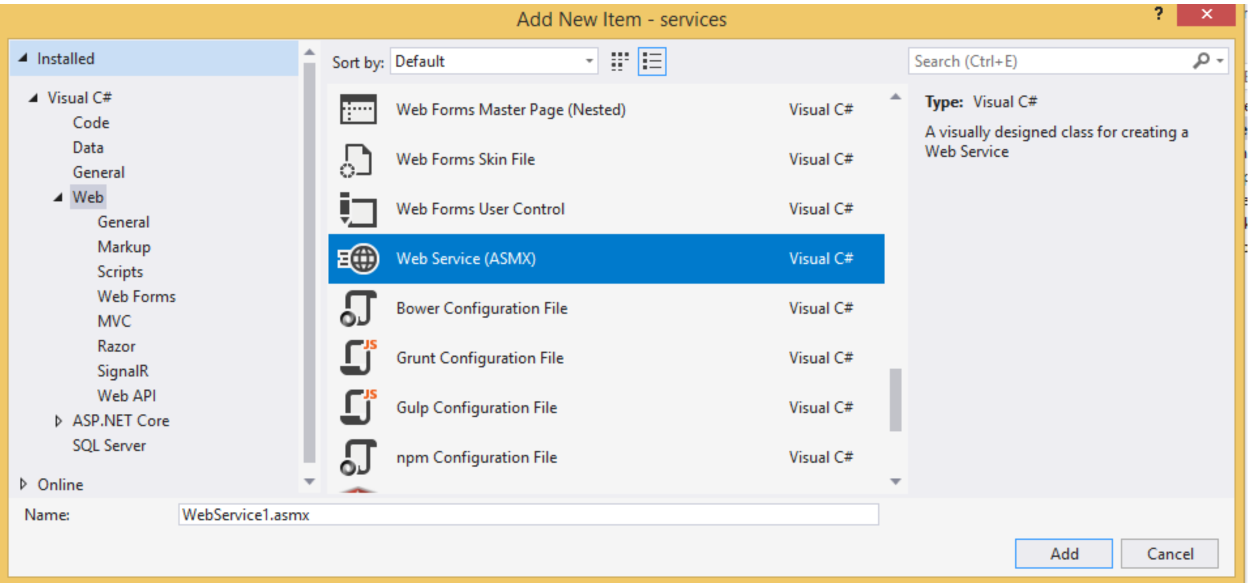
Single Page Application
A project template for creating rich client side JavaScript driven HTML5 applications using ASP.NET Web API. Single Page Applications provide a rich user experience which includes client-side interactions using HTML5, CSS3, and JavaScript.

Authentication
No Authentication
Change

Add folders & core references
□ Web Forms
□ MVC
□ Web API

Advanced
☑ Configure for HTTPS
□ Docker support
(Requires Docker Desktop)
□ Also create a project for unit tests
services.Tests

Add web service in a project



Add New Item - services

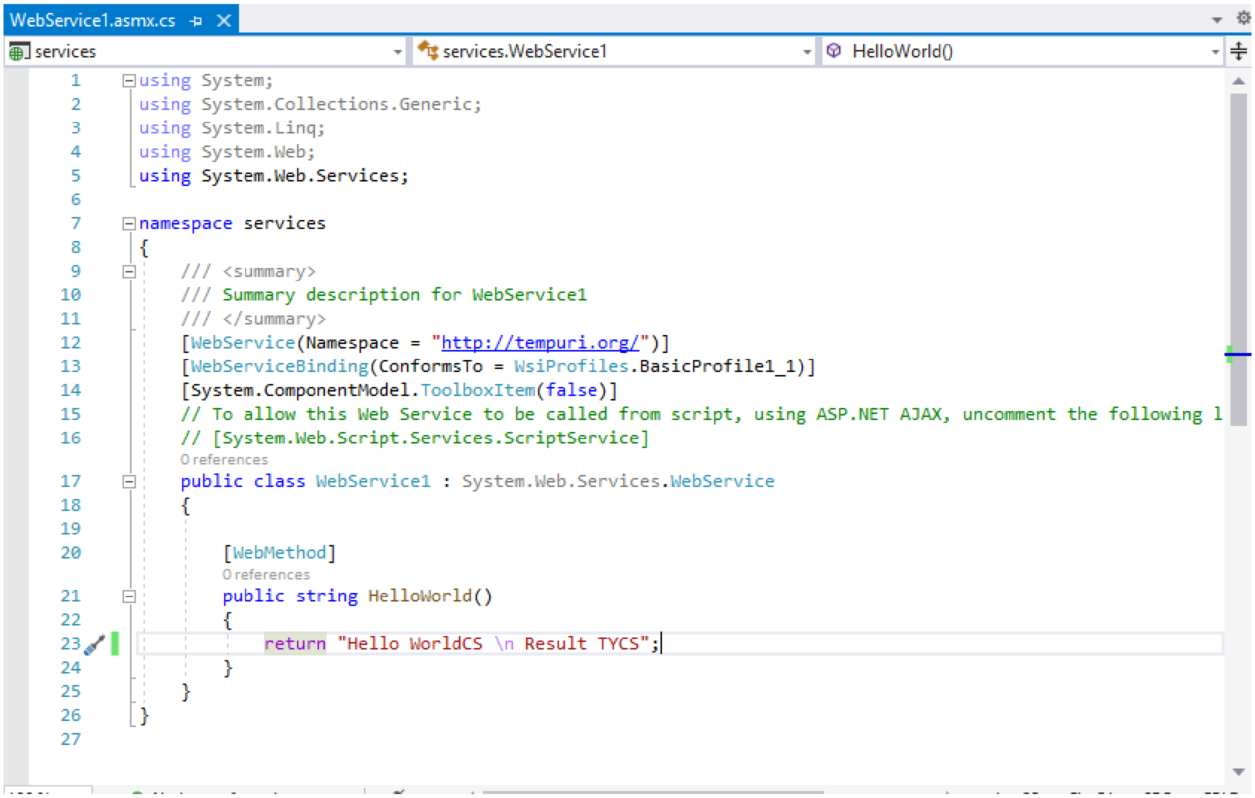
Sort by: Default

Search (Ctrl+E)

Web Forms Master Page (Nested)	Visual C#
Web Forms Skin File	Visual C#
Web Forms User Control	Visual C#
Web Service (ASMX)	Visual C#
Bower Configuration File	Visual C#
Grunt Configuration File	Visual C#
Gulp Configuration File	Visual C#
npm Configuration File	Visual C#

Type: Visual C#
A visually designed class for creating a Web Service

Name: WebService1.asmx

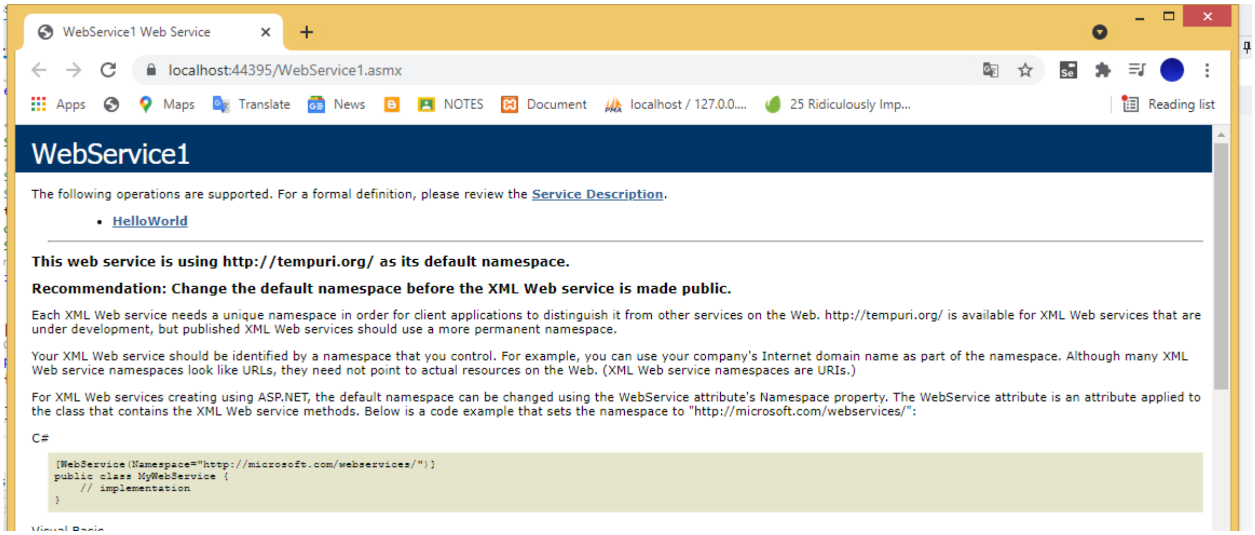


```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Services;

namespace services
{
    /// <summary>
    /// Summary description for WebService1
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [System.ComponentModel.ToolboxItem(false)]
    // To allow this Web Service to be called from script, using ASP.NET AJAX, uncomment the following l
    // [System.Web.Script.Services.ScriptService]
    0 references
    public class WebService1 : System.Web.Services.WebService
    {

        [WebMethod]
        0 references
        public string HelloWorld()
        {
            return "Hello WorldCS \n Result TYCS";
        }
    }
}
```

Save and run the project



WebService1

The following operations are supported. For a formal definition, please review the Service Description.

- HelloWorld

This web service is using http://tempuri.org/ as its default namespace.

Recommendation: Change the default namespace before the XML Web service is made public.

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. http://tempuri.org/ is available for XML Web services that are under development, but published XML Web services should use a more permanent namespace.

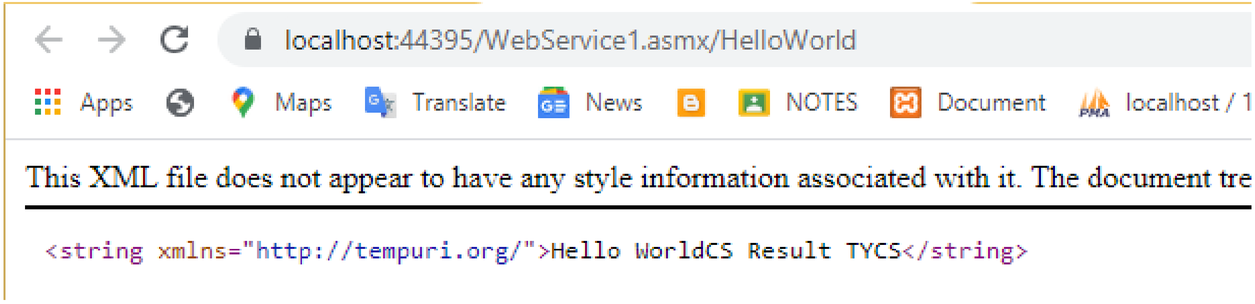
Your XML Web service should be identified by a namespace that you control. For example, you can use your company's Internet domain name as part of the namespace. Although many XML Web service namespaces look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces are URIs.)

For XML Web services creating using ASP.NET, the default namespace can be changed using the WebService attribute's Namespace property. The WebService attribute is an attribute applied to the class that contains the XML Web service methods. Below is a code example that sets the namespace to "http://microsoft.com/webservices/":

C#

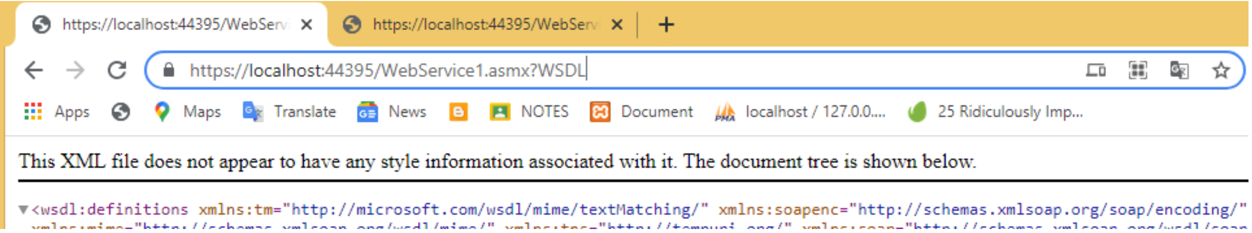
```
[WebService(Namespace="http://microsoft.com/webservices/")]
public class MyWebService {
    // implementation
}
```

Visual Basic



This XML file does not appear to have any style information associated with it. The document tre

```
<string xmlns="http://tempuri.org/">Hello WorldCS Result TYCS</string>
```

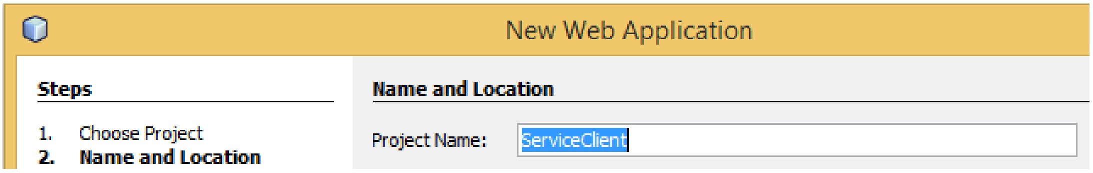

https://localhost:44395/WebServ
https://localhost:44395/WebServ
https://localhost:44395/WebService1.asmx?WSDL
Apps
Maps
Translate
News
NOTES
Document
localhost / 127.0.0....
25 Ridiculously Imp...

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<wsdl:definitions xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
```

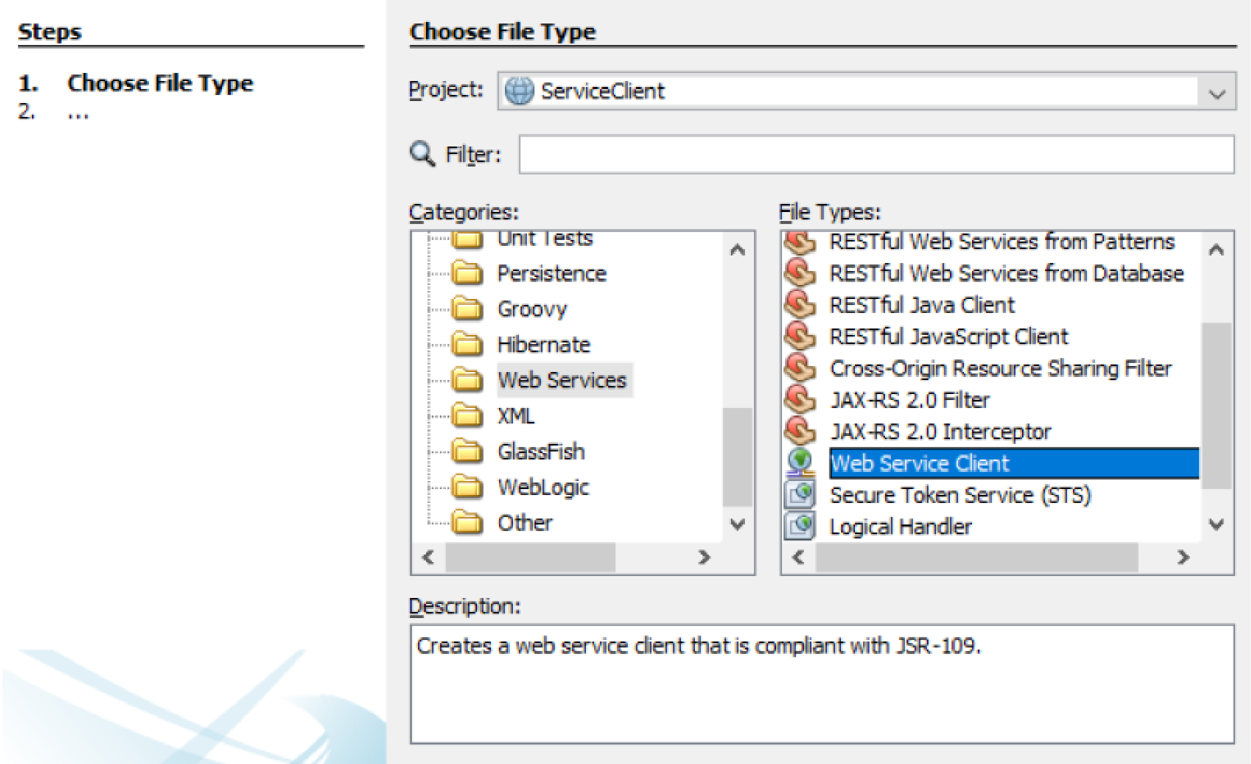
Don’t close Visual Studio and browser, just minimize it otherwise server will stop. But save the link anywhere, so that we can use it later

Now open NetBeans and Create a Web Application with name ServiceClient



Project Name: ServiceClient

Add webservice client



Steps

1. **Choose File Type**
2. ...

Choose File Type

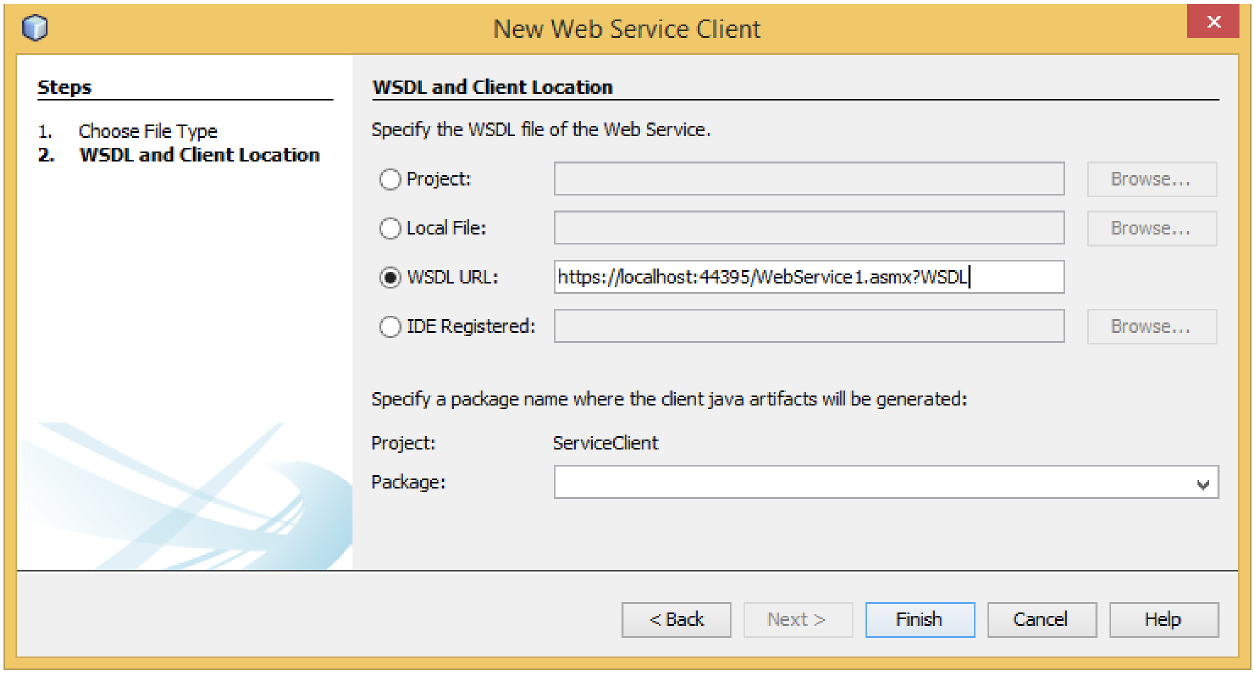
Project: ServiceClient

Filter:

Description:

Creates a web service client that is compliant with JSR-109.

Add WSDL URL and click on finsh



New Web Service Client

Steps

1. Choose File Type
2. **WSDL and Client Location**

WSDL and Client Location

Specify the WSDL file of the Web Service.

○ Project:

○ Local File:

● WSDL URL: https://localhost:44395/WebService1.asmx?WSDL

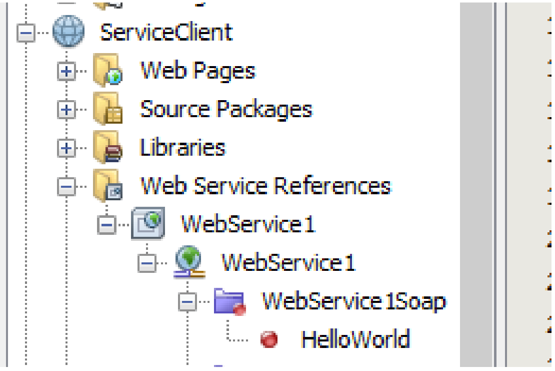
○ IDE Registered:

Specify a package name where the client java artifacts will be generated:

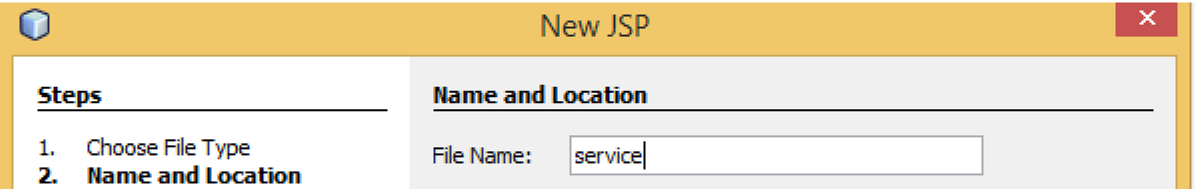
Project: ServiceClient

Package:

Now we get hello world service add this service into jsp page

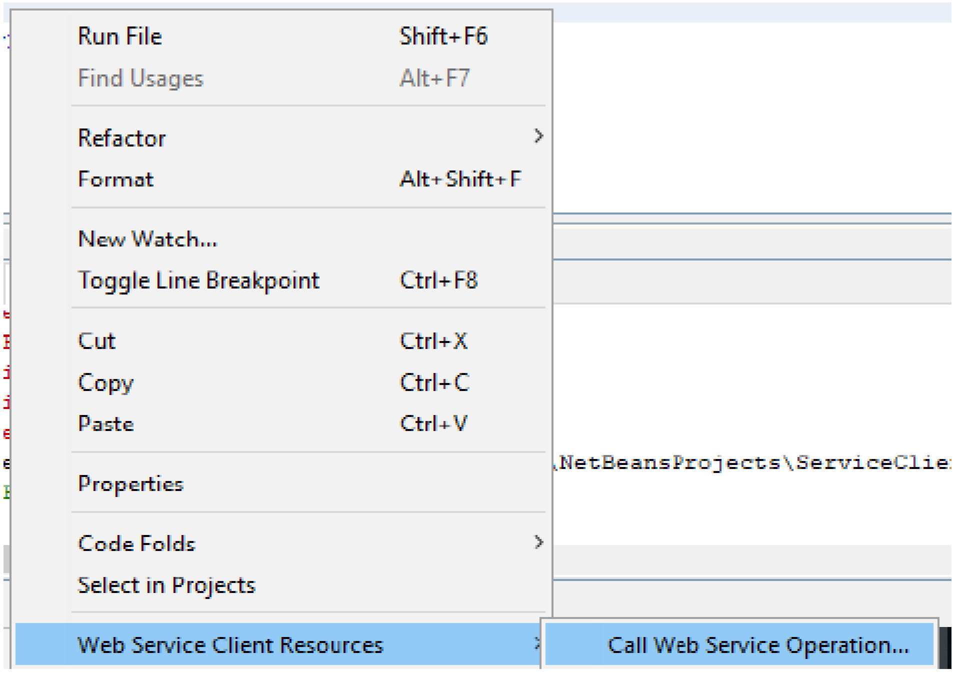


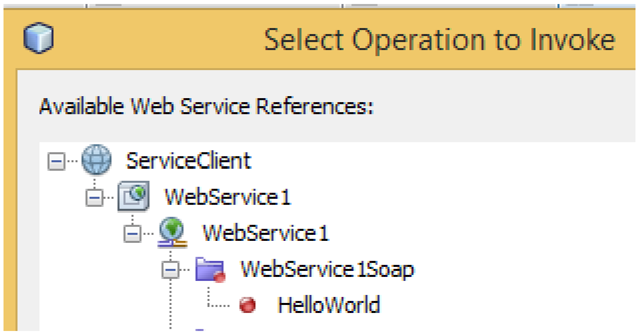
Create jsp page named its service



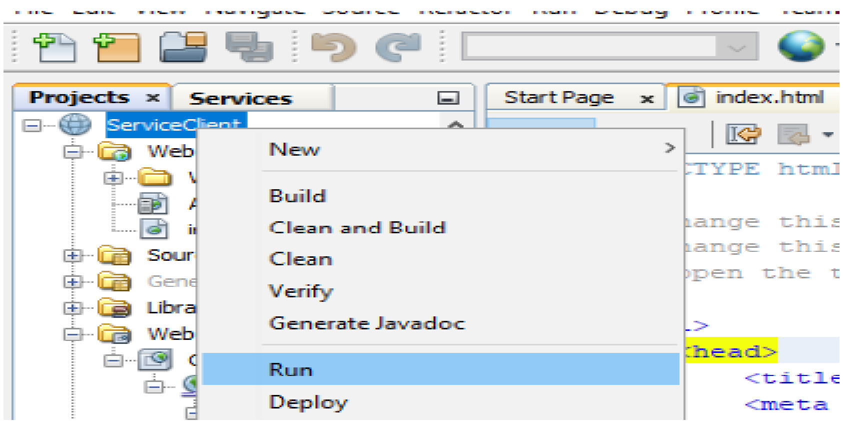
File Name: service

Add the webservice into jsp page

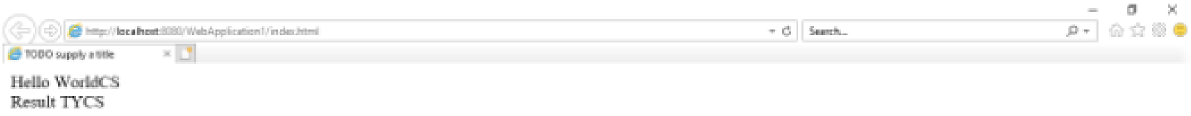




Run the serviceclient



Output:



```
Hello WorldCS
Result TYCS
```