Day 21 Assignment

By

VARUN SAI KUMAR CHEGONI

NB Healthcare and Technology

Date: 21 Feb 2022

Topics

Web Service

Content

1. Create a web service for Mathematical Operations.

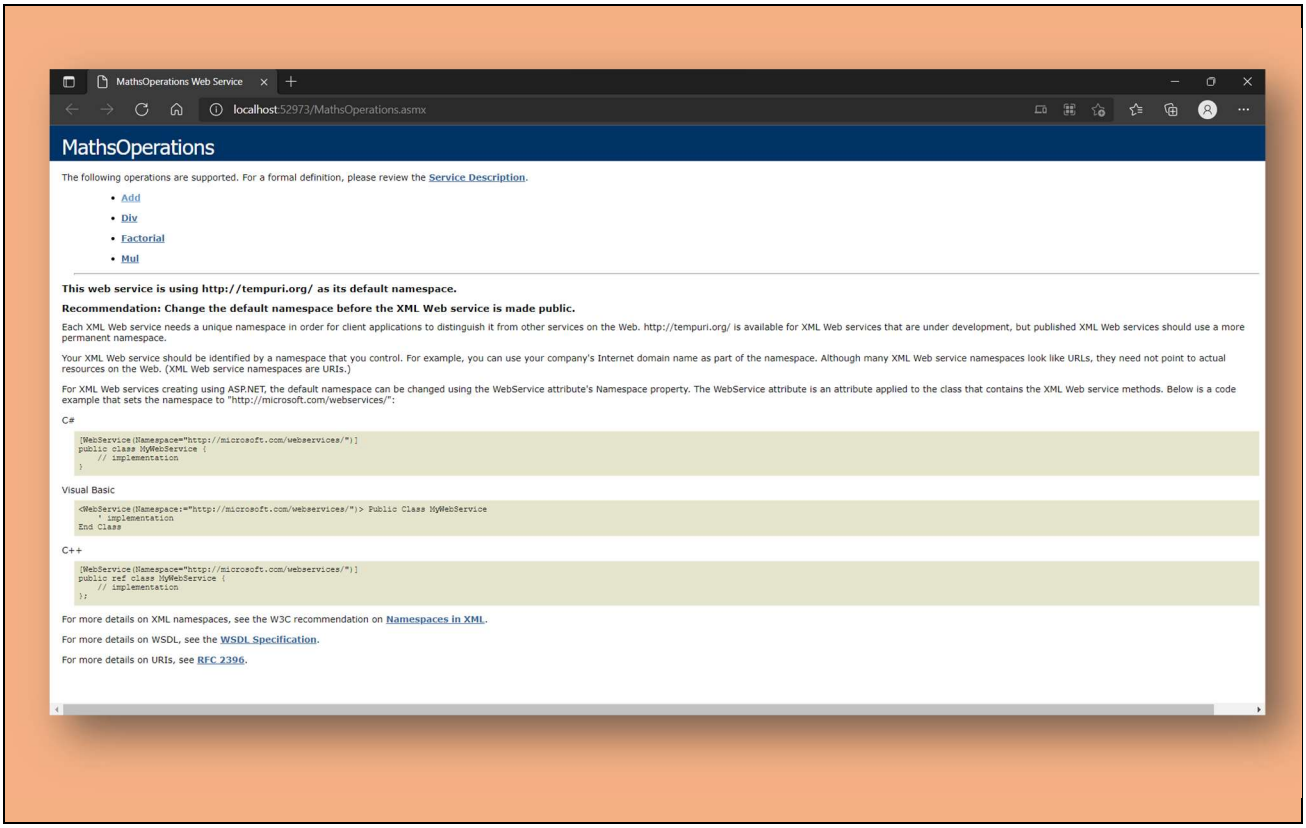
Steps to Create Web Service:

- Add .NET Framework project and item Template to Visual Studio in VS Installer
- After modifying open VS and select for dropdown: C# language, Windows platform, Web type.
- Find and select ASP.NET Web Application (.NET Framework)
- Create the web project.

Code :

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

namespace MyMathematicsWebLibrary
{
    /// <summary>
    /// Summary description for MathsOperations
    /// </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 line.
    // [System.Web.Script.Services.ScriptService]
    public class MathsOperations : System.Web.Services.WebService
    {
        [WebMethod]
        public int Factorial(int n)
        {
            int fact = 1;
            for (int i = 1; i < n; i++)
                fact *= i;
            return fact;
        }
        [WebMethod]
        public int Add(int a, int b)
        {
            return a + b;
        }
        [WebMethod]
        public int Mul(int a, int b)
        {
            return a * b;
        }
        [WebMethod]
        public int Div(int a, int b)
        {
            return a / b;
        }
    }
}
```

Output :

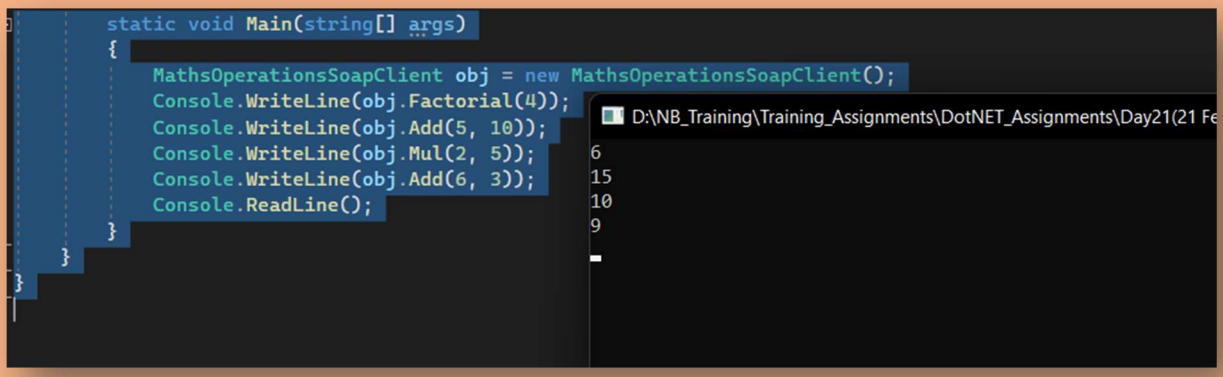

MathsOperations Web Service
localhost:52973/MathsOperations.asmx
MathsOperations
The following operations are supported. For a formal definition, please review the Service Description.
Add
Div
Factorial
Mul
This web service is using http://tempuri.org/ as its default namespace.
Recommendation: Change 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
<WebService(Namespace:="http://microsoft.com/webservices/")> Public Class MyWebService
' implementation
End Class
C++
[WebService(Namespace="http://microsoft.com/webservices/")]
public ref class MyWebService {
// implementation
};
For more details on XML namespaces, see the W3C recommendation on Namespaces in XML.
For more details on WSDL, see the WSDL Specification.
For more details on URIs, see RFC 2396.

2. Create a Console Application and consume the webservice

Code :

```
using MyTestClientApp.ServiceReference1;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace MyTestClientApp
{
    internal class Program
    {
        static void Main(string[] args)
        {
            MathsOperationsSoapClient obj = new MathsOperationsSoapClient();
            Console.WriteLine(obj.Factorial(4));
            Console.WriteLine(obj.Add(5, 10));
            Console.WriteLine(obj.Mul(2, 5));
            Console.WriteLine(obj.Add(6, 3));
            Console.ReadLine();
        }
    }
}
```

Output :

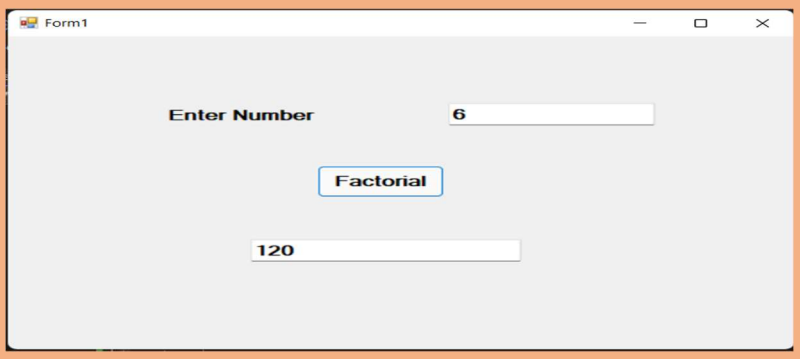


3. Create a Windows Forms application and consume the webservice (Factorial).

Code :

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using WindowsFormsApp1.ServiceReference1;

namespace WindowsFormsApp1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void textBox2_TextChanged(object sender, EventArgs e)
        {

        }

        private void button1_Click(object sender, EventArgs e)
        {
            int num;
            num = Convert.ToInt32(textBox1.Text);
            MathsOperationsSoapClient ans = new MathsOperationsSoapClient();
            textBox2.Text = ans.Factorial(num).ToString();
        }

        private void textBox1_TextChanged(object sender, EventArgs e)
        {

        }

        private void label1_Click(object sender, EventArgs e)
        {

        }
    }
}
```

Output :



4. Screen shots of webservice running

Screenshot: