

Day 21 Assignment

By

Vinay Kudali

21-02-2022

2. Create a web service for Mathematical Operations. Example: Factorial, add, mul, div

Code:

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

namespace WebApplication
{
    /// <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
    line.
    // [System.Web.Script.Services.ScriptService]
    public class WebService1 : System.Web.Services.WebService
    {

        [WebMethod]
        public int Add(int m, int n)
        {
            return m+n;
        }
        [WebMethod]
        public int Mul(int m, int n)
        {
            return m*n;
        }
        [WebMethod]
        public int Div(int m, int n)
        {
            return m/n;
        }
        [WebMethod]
```

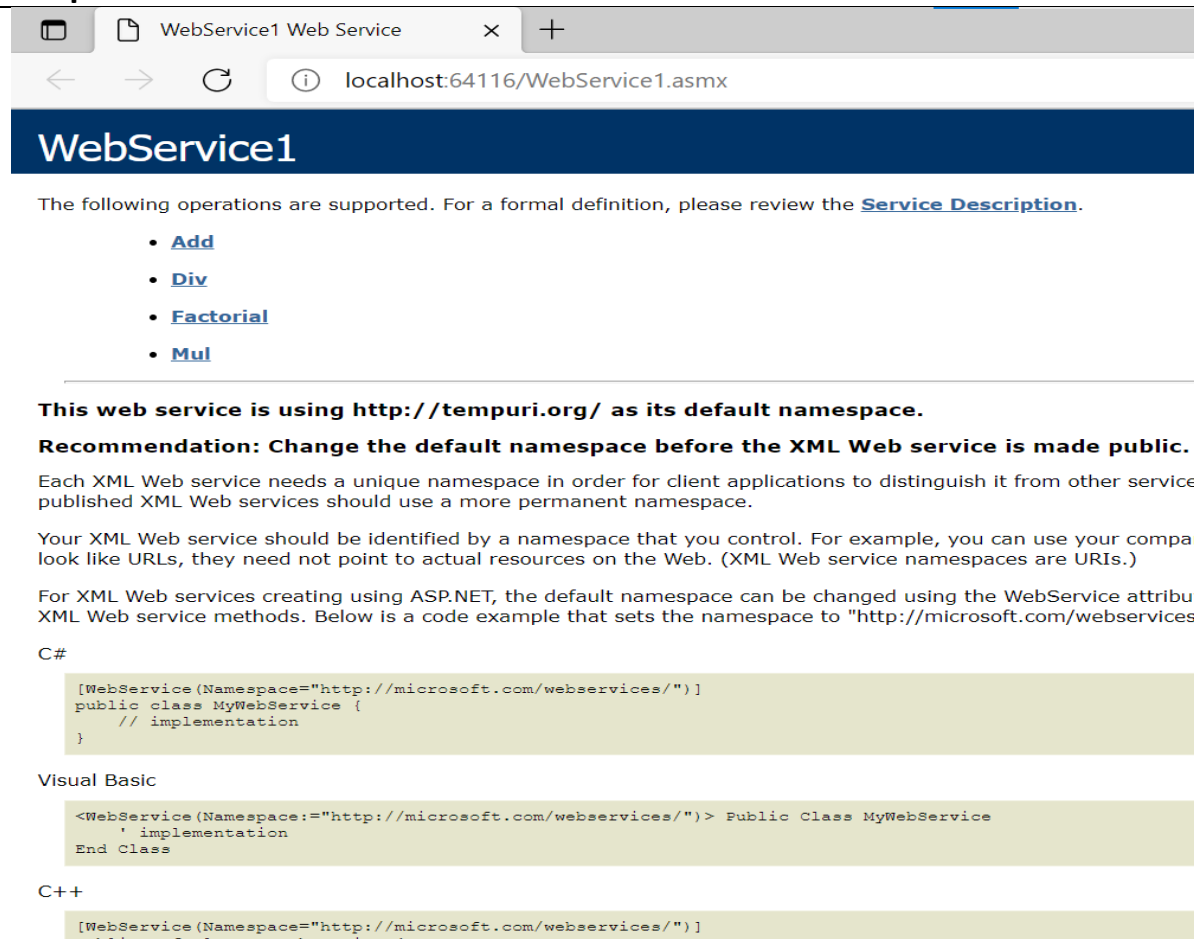
```

public int Factorial(int n)
{
    int fact = 1;
    for (int i = 1; i <=n; i++)
    {
        fact*=i;
    }
    return fact;
}

}
}

```

Output:



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 service: 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 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 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 {

```

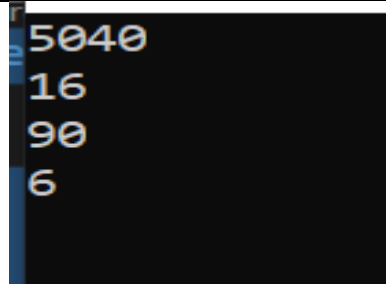
3. Create a Console Application and consume the webservice

Code:

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

namespace Algebra
{
    internal class Program
    {
        static void Main(string[] args)
        {
            WebService1SoapClient wssc = new WebService1SoapClient();
            Console.WriteLine(wssc.Factorial(7));
            Console.WriteLine(wssc.Add(7, 9));
            Console.WriteLine(wssc.Mul(9, 10));
            Console.WriteLine(wssc.Div(12, 2));
            Console.ReadLine();
        }
    }
}
```

Output:

A screenshot of a console window with a black background and yellow text. It displays the output of a C# console application that consumes a web service. The output consists of five lines of numbers: 5040, 16, 90, and 6, each on a new line. The first line is slightly indented compared to the others.

```
5040
16
90
6
```

4. Create a Windows Forms application and consume the webservice [for finding factorial of the number]

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 button1_Click(object sender, EventArgs e)
        {
            int n=Convert.ToInt32(textBox1.Text);
            WebService1SoapClient obj = new WebService1SoapClient();
            textBox2.Text=obj.Factorial(n).ToString();
        }
    }
}
```

Output:

Form1

5

Factorial

120