Day 21 Assignment

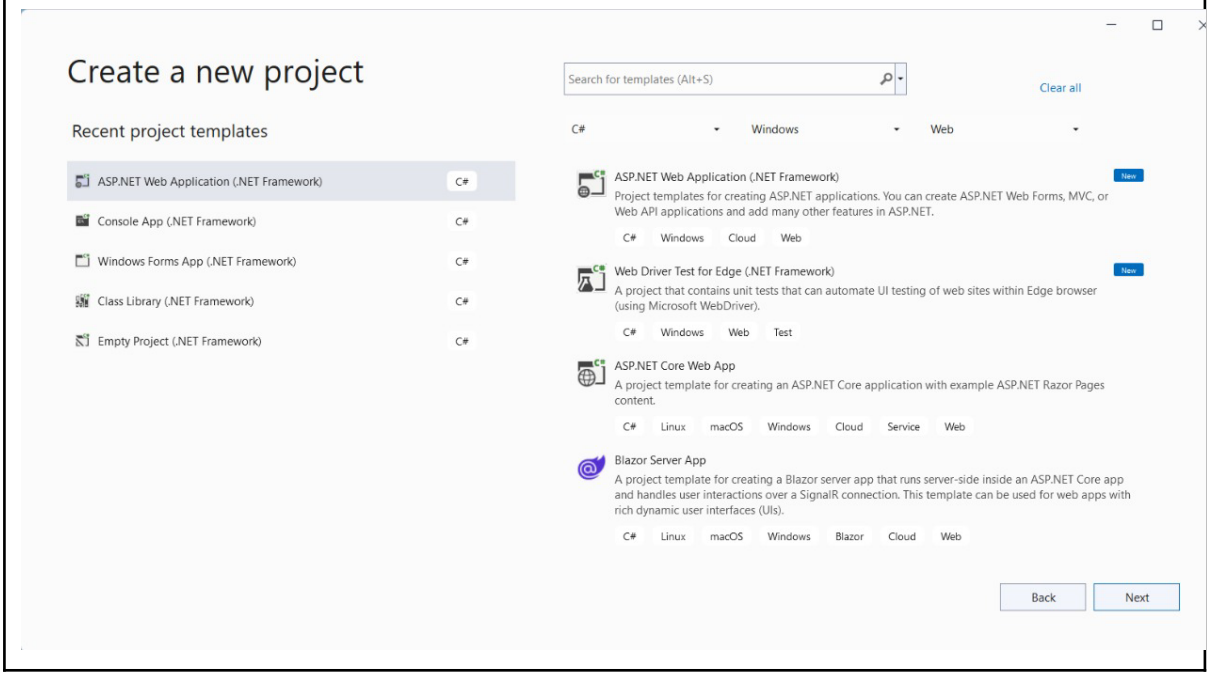
By

Praveen Chakravarthi

21-02-2022

NB Health Care

1. Update your Visual Studio with .Net Framework Templates add on(as discussed in the class)



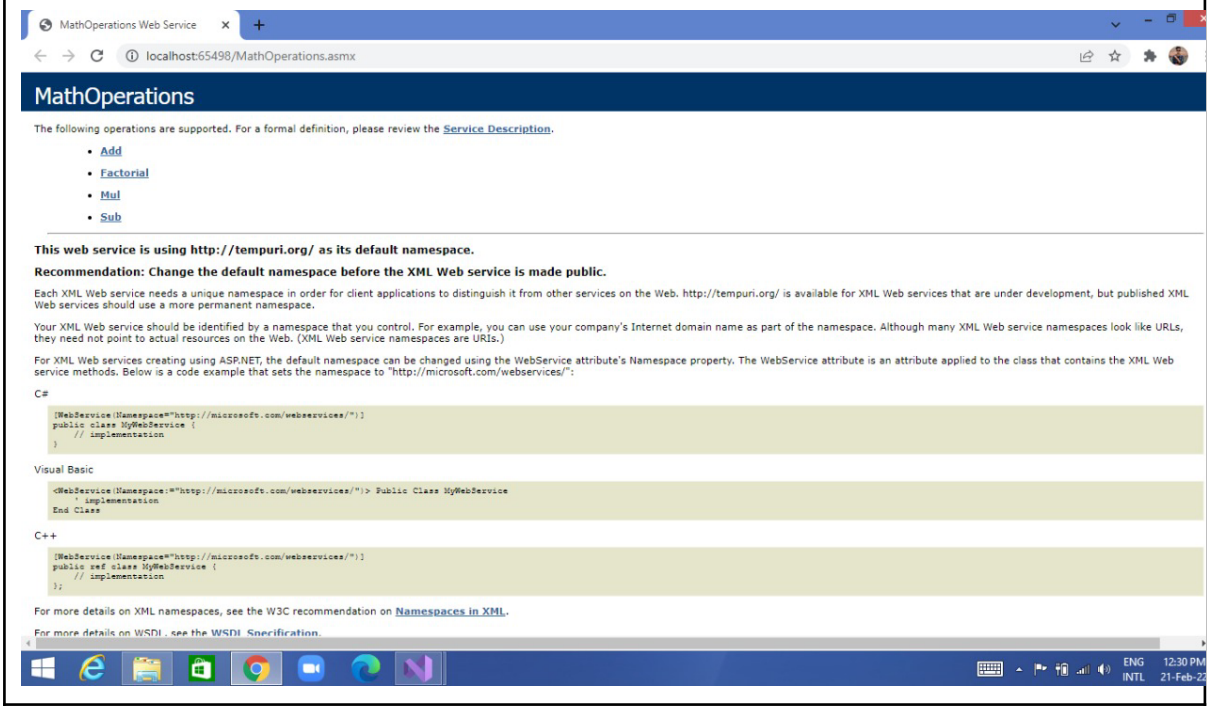
2. Create a web service for Mathematical Operations.

Code :

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

namespace PraveenWebService
{
    /// <summary>
    /// Summary description for MathOperations
    /// </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 MathOperations : 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 Mul(int a,  int b)
        {
            return a*b;
        }
        [WebMethod]
        public int Add(int a, int b)
        {
            return a+b ;
        }
        [WebMethod]
        public int Sub(int a, int b)
        {
            return a-b;
        }
    }
}
```

Output :

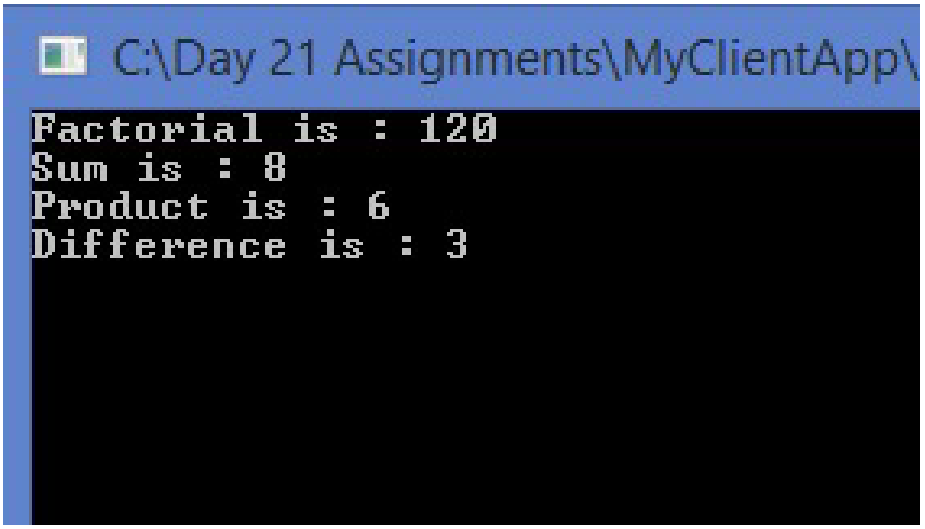


3. Create a Console Application and consume the webservice

Code :

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


namespace MyClientApp
{
    internal class Program
    {
        static void Main(string[] args)
        {
            MathOperationsSoapClient MO = new MathOperationsSoapClient();
            Console.WriteLine($"Factorial is : {MO.Factorial(5)}");
            Console.WriteLine($"Sum is : {MO.Add(3,5)}");
            Console.WriteLine($"Product is : {MO.Mul(2,3)}");
            Console.WriteLine($"Difference is : {MO.Sub(5,2)}");
            Console.ReadLine();
        }
    }
}
```

Output :

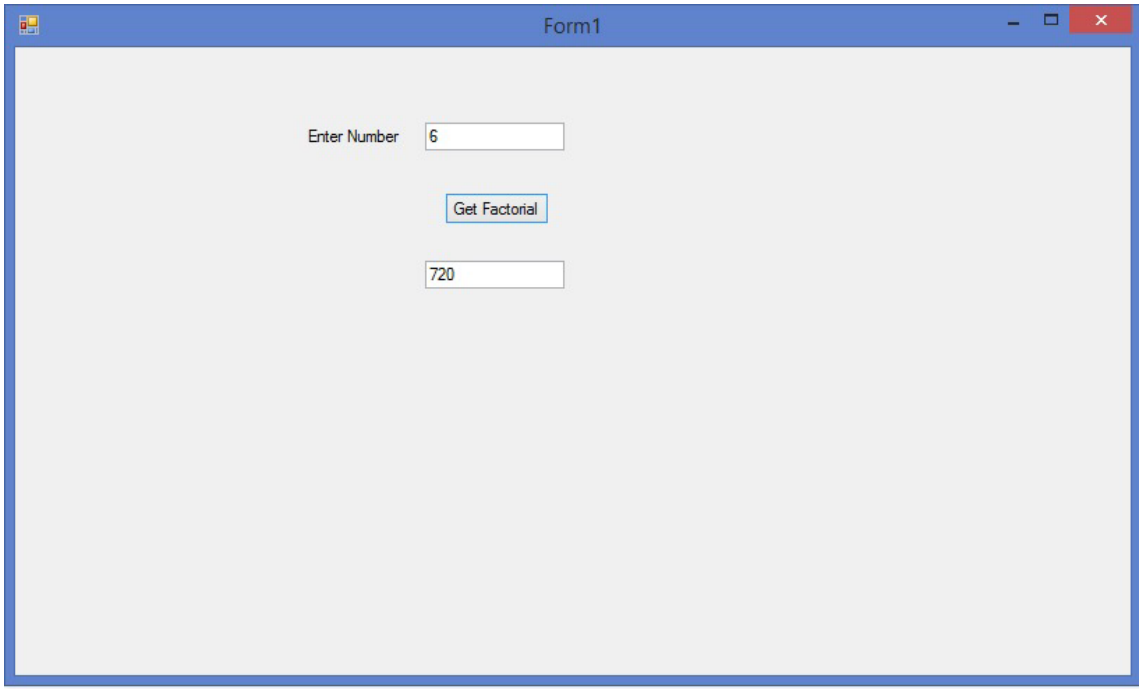


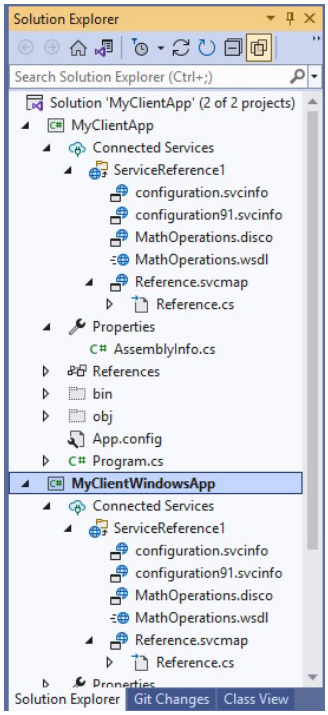
4. Create a Windows Forms application and consume the webservice [for finding factorial of the number]

Code :

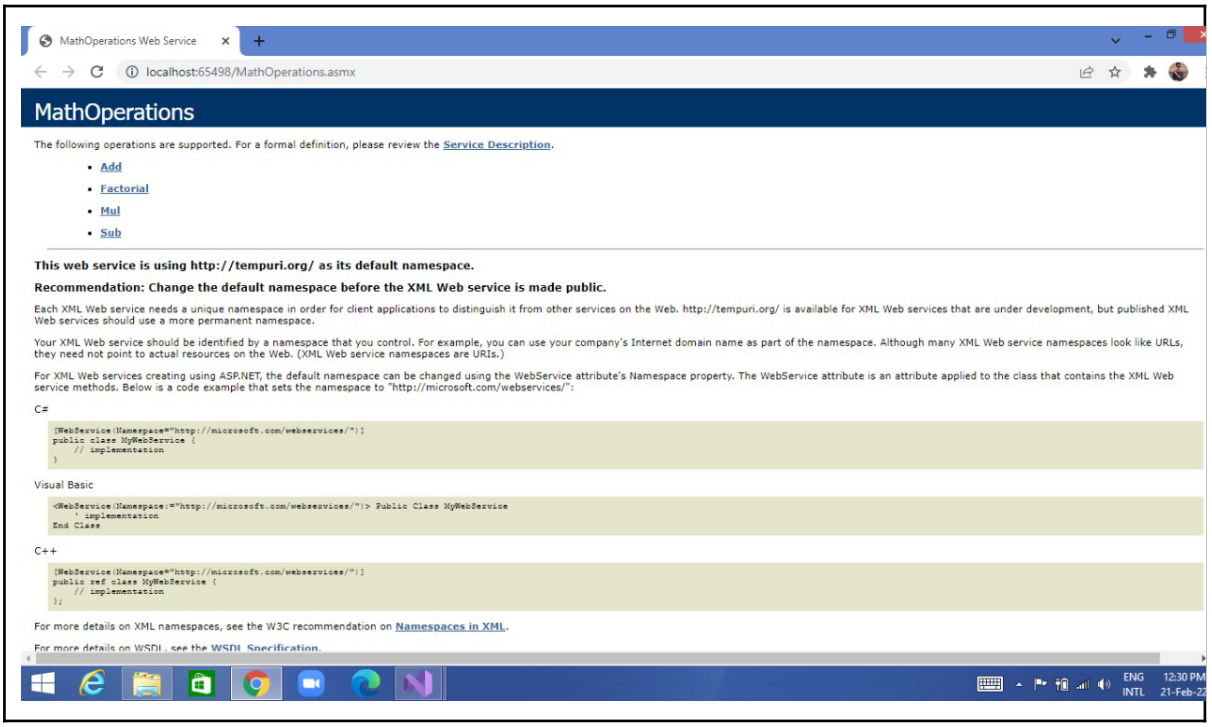
```
using MyClientWindowsApp.ServiceReference1;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

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

        private void button1_Click(object sender, EventArgs e)
        {
            int n = Convert.ToInt32(textBox1.Text);
            MathOperationsSoapClient MO = new MathOperationsSoapClient();
            textBox2.Text = MO.Factorial(n).ToString();

        }
    }
}
```

Output :





5. Put the screen shots of webservice running


MathOperations Web Service
localhost:65498/MathOperations.asmx
MathOperations
The following operations are supported. For a formal definition, please review the Service Description.
Add
Factorial
Mul
Sub
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.
ENG
INTL
12:30 PM
21-Feb-22