

TITLE:

Web Services - Assignment

Student Name:

Urwah Rasheed

Registration No:

03-3-1-055-2022

Subject:

Internet Application Development

Submitted to:

Mr. Irfan Hameed

Submitted Date:

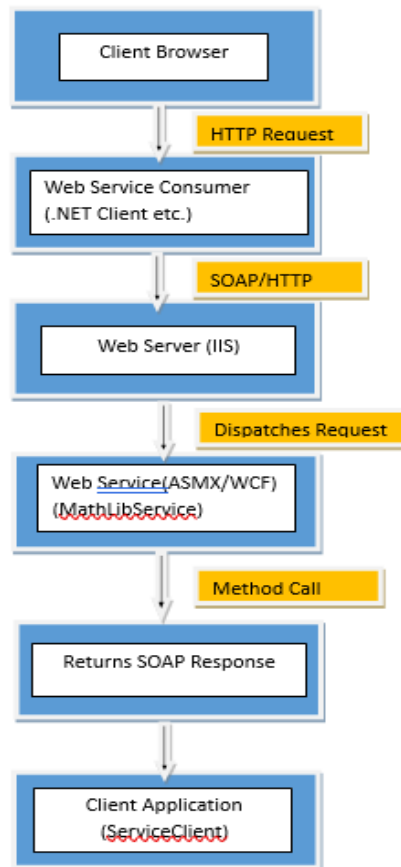
May 18, 2025



Pakistan Institute of Engineering and Applied Sciences

QUESTION NO: 01

Draw web services execution model.

**QUESTION NO: 02**

Develop a web service that includes the following four web methods:

1. Add – Accepts two input parameters and returns the sum as a string.
2. Subtract – Accepts two input parameters and returns the difference as a string.
3. Multiply – Accepts two input parameters and returns the product as a string.
4. Divide – Accepts two input parameters and returns the quotient as a string.

Each method should take two parameters and return a single string result.

Step-by-Step: Create a .asmx Web Service in ASP.NET Web Forms (VB.NET)**1. Create the .asmx File**

In Visual Studio, follow these steps:

1. Right-click on the project (e.g., WebApplication1) in Solution Explorer.
2. Select Add > New Item.
3. Choose Web Service (ASMX).
4. Name it: WebService1.asmx (or your preferred name).
5. Click Add.

CODE:

```
Imports System.Web.Services
```

```
Imports System.Web.Services.Protocols
```

```
Imports System.ComponentModel
```

```
<WebService(Namespace:="http://tempuri.org/")>
```

```
<WebServiceBinding(ConformsTo:=WsiProfiles.BasicProfile1_1)>
```

```
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated(>
```

```
Public Class MathLibService
```

```
    Inherits System.Web.Services.WebService
```

```
<WebMethod(>
```

```
    Public Function Add(ByVal a As Integer, ByVal b As Integer) As String
```

```
        Return (a + b).ToString()
```

```
    End Function
```

```
<WebMethod(>
```

```
    Public Function Subtract(ByVal a As Integer, ByVal b As Integer) As String
```

```
        Return (a - b).ToString()
```

```
    End Function
```

```
<WebMethod(>
```

```
Public Function Multiply(ByVal a As Integer, ByVal b As Integer) As String
```

```
    Return (a * b).ToString()
```

```
End Function
```

```
<WebMethod(>
```

```
Public Function Divide(ByVal a As Integer, ByVal b As Integer) As String
```

```
    If b = 0 Then
```

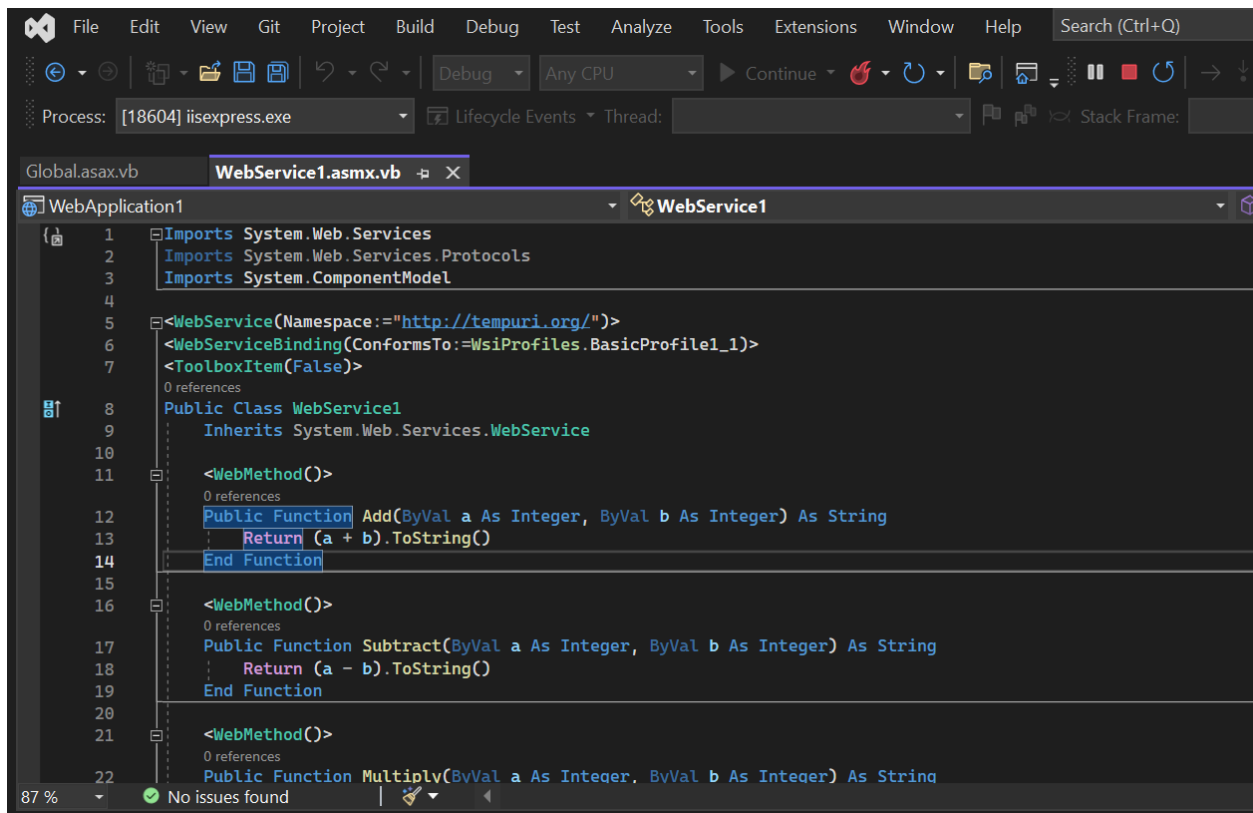
```
        Return "Cannot divide by zero"
```

```
    End If
```

```
    Return (a \ b).ToString() ' Integer division
```

```
End Function
```

```
End Class
```



QUESTION NO: 03

Test the web service using browser on local pc.

Install Internet Information Services on your laptop/pc (if not installed)

Using IIS manager create two separate applications as follows:

(a) MathLibService

(b) ServiceClient

Step 1: Install IIS

- Go to **Control Panel > Programs > Turn Windows features on/off**
- Enable:
 - Internet Information Services
 - .NET Extensibility
 - ASP.NET
 - IIS Management Console

Step 2: Publish Your Service

- Right-click project → **Publish**
- Choose: **Folder** → Publish to C:\inetpub\wwwroot\MathLibService

Step 3: Add Application in IIS

1. Open IIS Manager
2. Right-click Default Web Site → Add Application
3. Alias: MathLibService
4. Physical Path: C:\inetpub\wwwroot\MathLibService
5. Test with: <http://localhost/MathLibService/MathLibService.asmx>

INTERNET APPLICATION DEVELOPMENT

Add Web Reference

Navigate to a web service URL and click Add Reference to add all the available services.

URL:

→

WebService1

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [Add](#)
- [Divide](#)
- [Multiply](#)
- [Subtract](#)

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.)

Web services found at this URL:

1 Service Found:
- WebService1

Web reference name:

Add Reference

WebService1 Web Service

https://localhost:44344/WebService1.asmx

Gmail YouTube Maps Student Registratio... Jobs | LinkedIn ChatGPT Oracle SmartX Innovations... preply - Search 6th Semester

WebService1

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [Add](#)
- [Divide](#)
- [Multiply](#)
- [Subtract](#)

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 look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces are URIs.)
For XML Web services created using ASP.NET, the default namespace can be changed using the WebService attribute's Namespace property. The WebService attribute is an attribute on 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
```

WebService1 Web Service x +

← ↻ https://localhost:44344/WebService1.asmx?op=Add

Gmail YouTube Maps Student Registratio... Jobs | LinkedIn ChatGPT Oracle SmartX Innovations... prepl

WebService1

Click [here](#) for a complete list of operations.

Add

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
a:	5
b:	7

WebService1

Click [here](#) for a complete list of operations.

Divide

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
a:	8
b:	4

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

```
<string xmlns="http://tempuri.org/">2</string>
```

QUESTION NO: 04

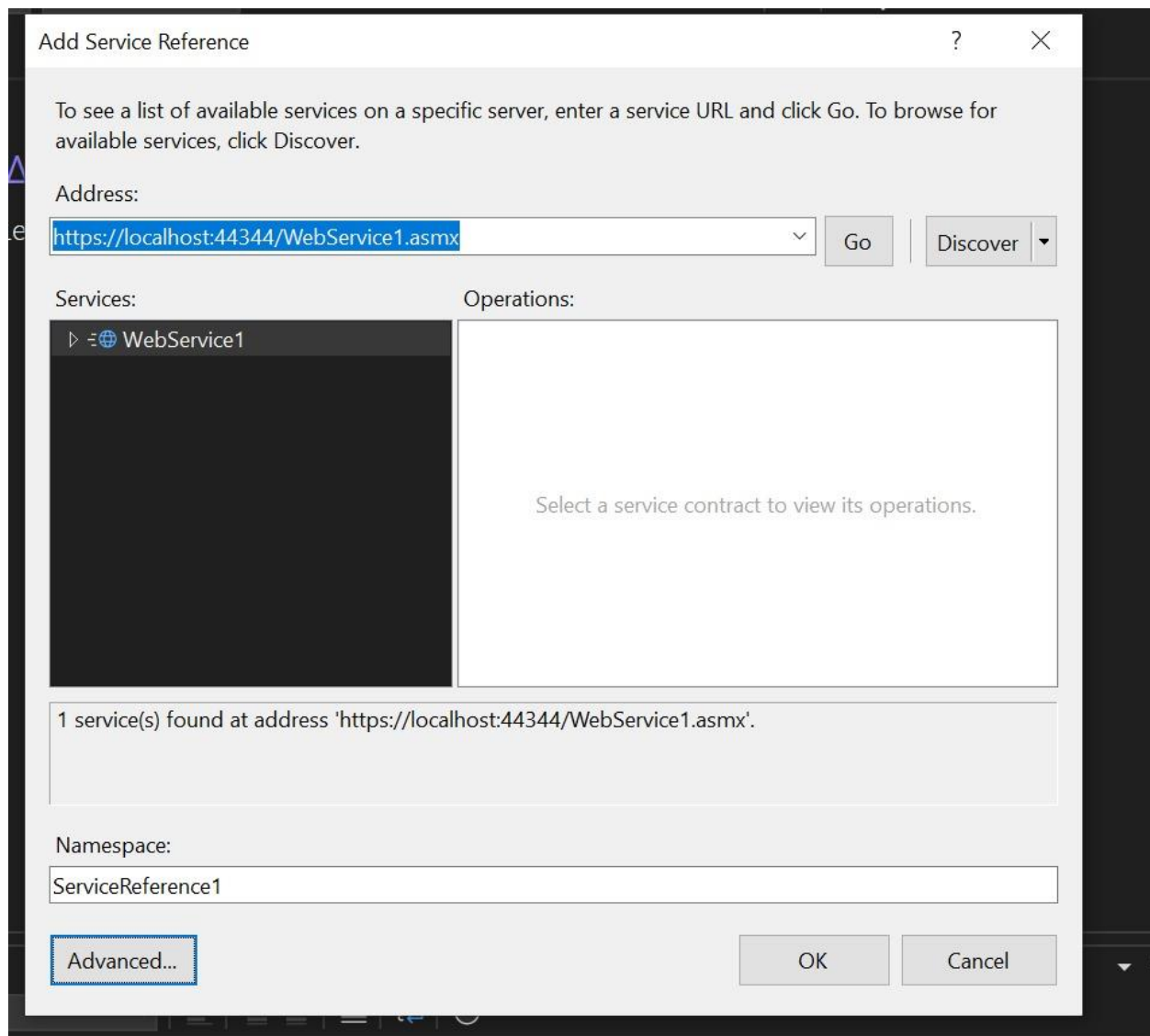
Implement a web application “ServiceClient” which will use the MathLibService as a web service. Develop a suitable web page for its demonstration. Use some text boxes to take input from user. Use proxy class to get answer (response from web service) and display it on page to user.

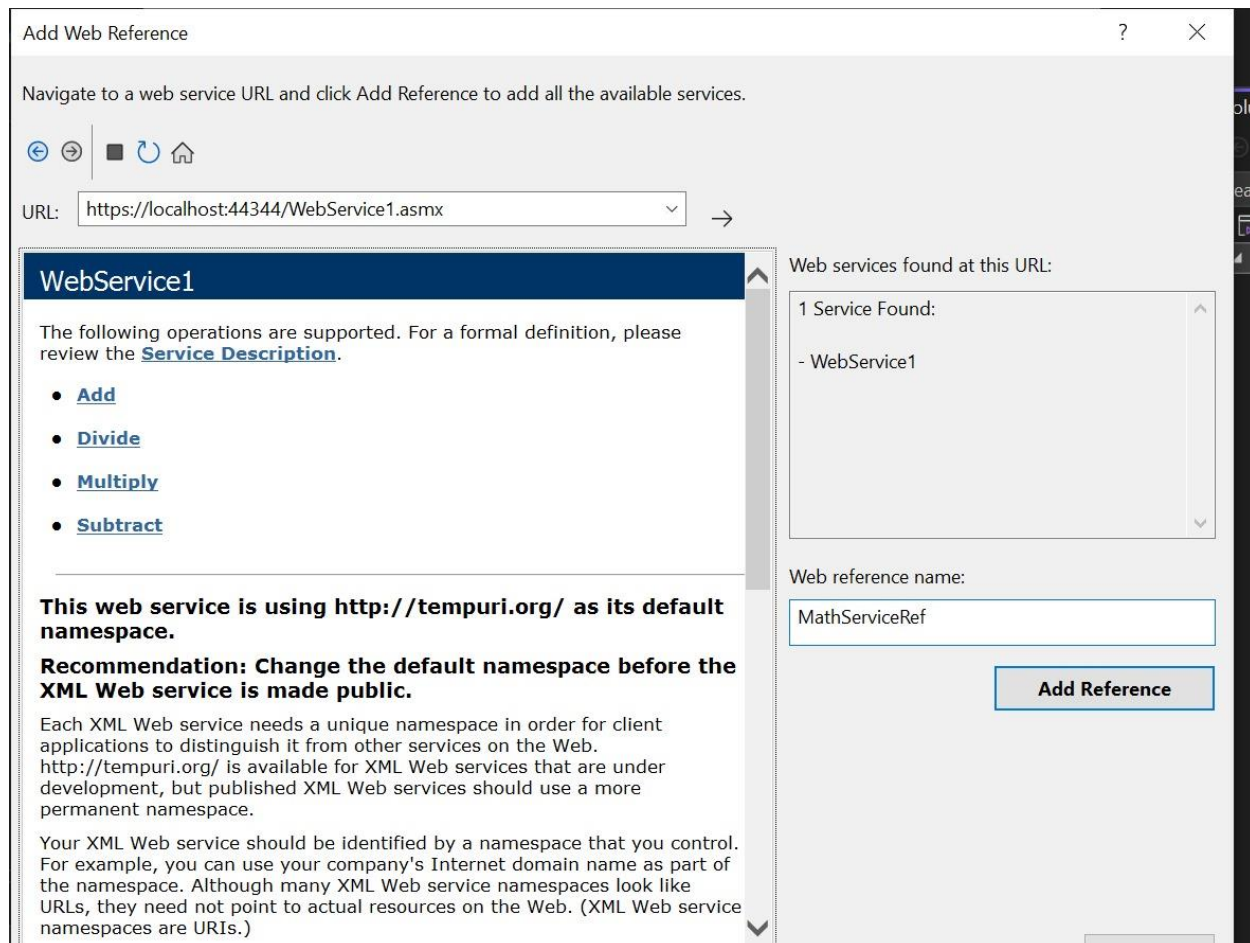
Step 1: Create a New Project

- File → New → Project → ASP.NET Web Application (.NET Framework)
- Name it: ServiceClient
- Choose: Empty → Enable **Web Forms**

► Step 2: Add Web Reference

1. Right-click project → Add Service Reference → Advanced → Add Web Reference
2. URL: `http://localhost/MathLibService/MathLibService.asmx`
3. Name it: MathServiceRef





Design Web Form

Default.aspx:

Imports System.Web.Services

Imports System.Web.Services.Protocols

Imports System.ComponentModel

```
<WebService(Namespace:="http://tempuri.org/")>
```

```
<WebServiceBinding(ConformsTo:=WsiProfiles.BasicProfile1_1)>
```

```
<ToolboxItem(False)>
```

Public Class WebService1

Inherits System.Web.Services.WebService

<WebMethod(>

Public Function Add(ByVal a As Integer, ByVal b As Integer) As String

Return (a + b).ToString()

End Function

<WebMethod(>

Public Function Subtract(ByVal a As Integer, ByVal b As Integer) As String

Return (a - b).ToString()

End Function

<WebMethod(>

Public Function Multiply(ByVal a As Integer, ByVal b As Integer) As String

Return (a * b).ToString()

End Function

<WebMethod(>

Public Function Divide(ByVal a As Integer, ByVal b As Integer) As String

If b = 0 Then

Return "Cannot divide by zero"

End If

Return (a \ b).ToString() ' Integer division

End Function

End Class

Default.aspx.vb:

Imports System.Runtime.Remoting

Imports WebService1

Partial Class _Default

Inherits System.Web.UI.Page

Protected Sub btnAdd_Click(sender As Object, e As EventArgs) Handles
btnAdd.Click

Dim service As New Service1()

lblResult.Text = "Result: " & service.Add(Convert.ToInt32(txtA.Text),
Convert.ToInt32(txtB.Text))

End Sub

Protected Sub btnSub_Click(sender As Object, e As EventArgs) Handles
btnSub.Click

Dim service As New Service1()

lblResult.Text = "Result: " & service.Subtract(Convert.ToInt32(txtA.Text),
Convert.ToInt32(txtB.Text))

End Sub

Protected Sub btnMul_Click(sender As Object, e As EventArgs) Handles
btnMul.Click

Dim service As New Service1()

lblResult.Text = "Result: " & service.Multiply(Convert.ToInt32(txtA.Text),
Convert.ToInt32(txtB.Text))

End Sub

Protected Sub btnDiv_Click(sender As Object, e As EventArgs) Handles
btnDiv.Click

Dim service As New Service1()

lblResult.Text = "Result: " & service.Divide(Convert.ToInt32(txtA.Text),
Convert.ToInt32(txtB.Text))

End Sub

End Class

Simple Calculator

Number A:

Number B:

Add

Subtract

Multiply

Divide

Simple Calculator

Number A:

Number B:

Result: 30

Simple Calculator

Number A:

Number B:

Please enter valid integers.