

Name :

Hafiz Shoaib Mehmood

Batch :

CIS(22-26)

Subject :

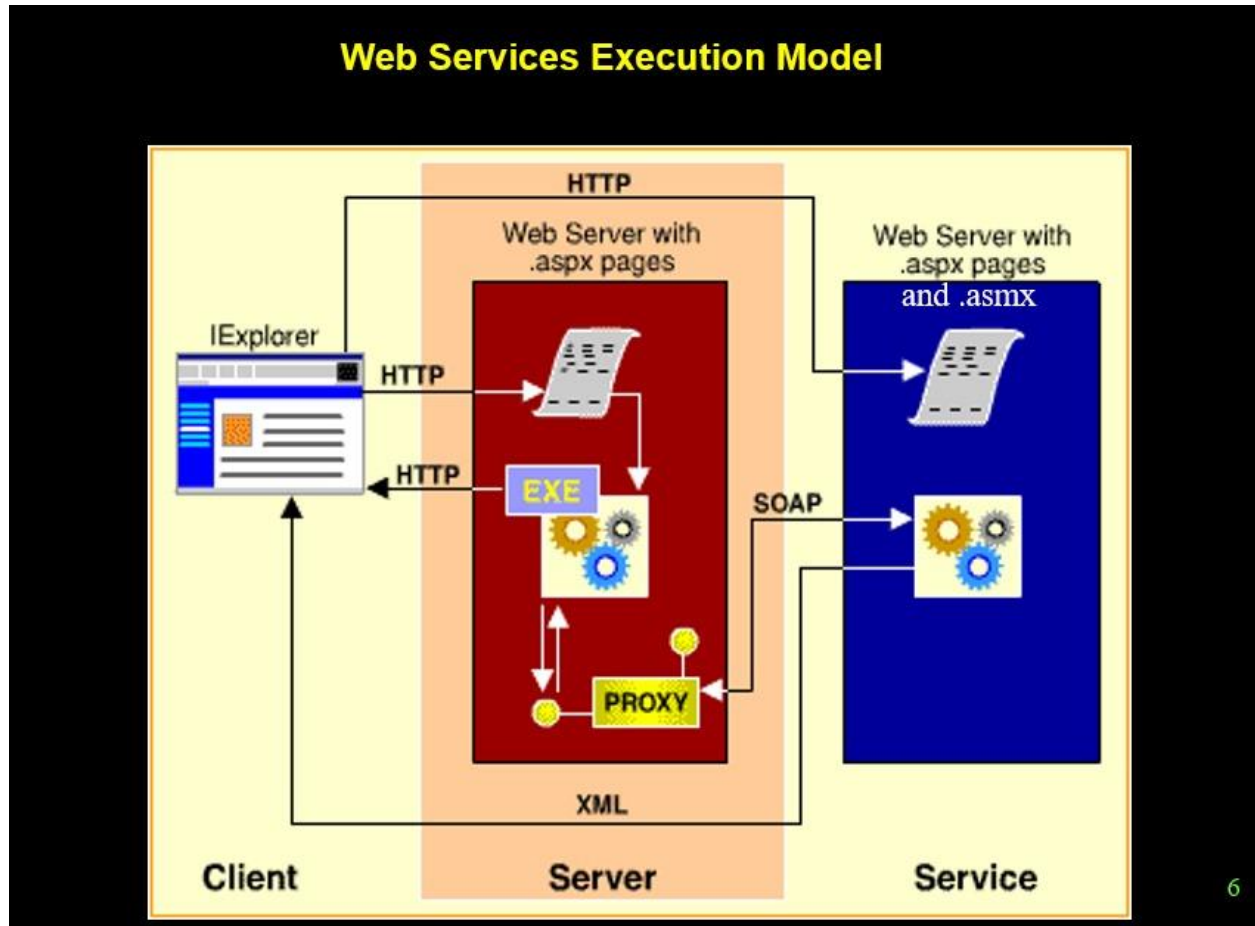
IAD

Assignment :

Web services

Web Services

Q1) Draw web services execution model?

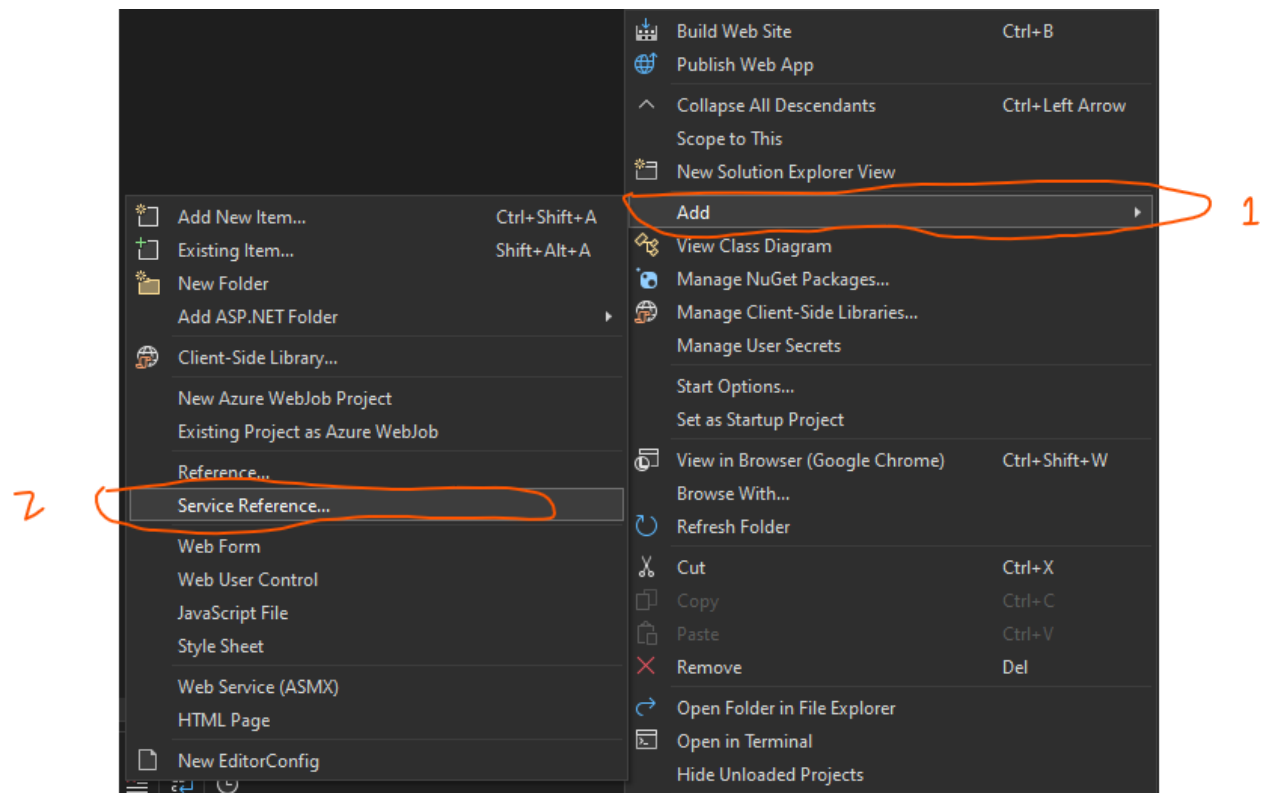


Q2) Develop a web service with four web methods as follows:

- (a) Add
- (b) Subtract
- (c) Multiply
- (d) Divide

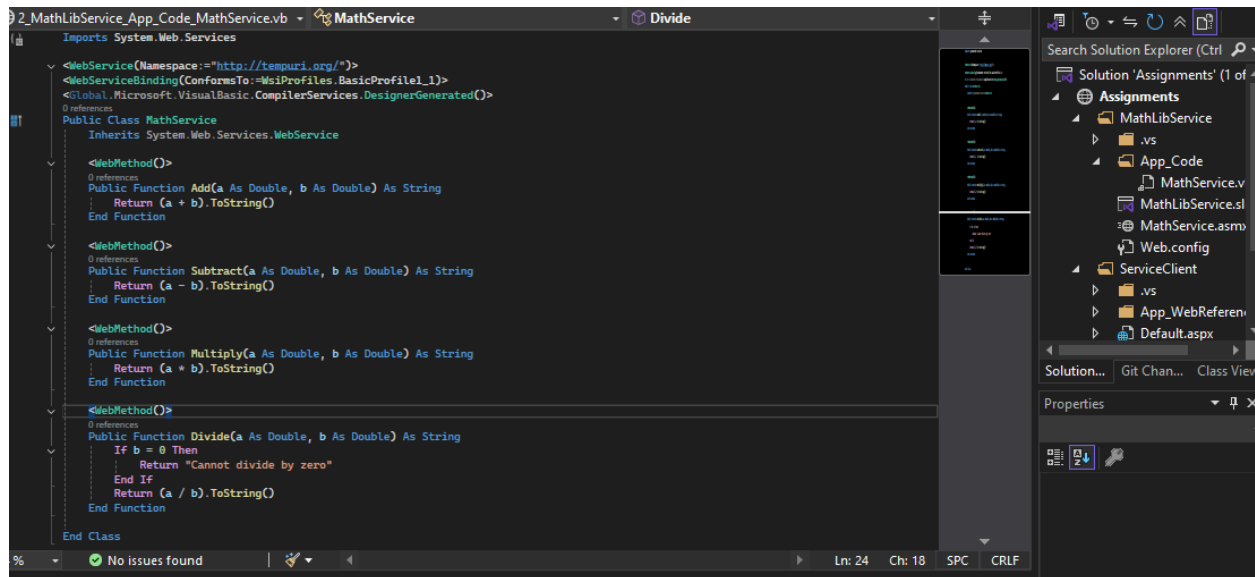
Answer :

MathLibService.asmx :



```
<%@ WebService Language="VB" CodeBehind="~/App_Code/MathService.vb"  
Class="MathService" %>
```

MathLibService_App_Code_MathService.vb :



```
Imports System.Web.Services

<WebService(Namespace:="http://tempuri.org/")>
<WebServiceBinding(ConformsTo:=WsdlProfiles.BasicProfile1_1)>
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated()>
Public Class MathService
    Inherits System.Web.Services.WebService

    <WebMethod()>
    0 references
    Public Function Add(a As Double, b As Double) As String
        Return (a + b).ToString()
    End Function

    <WebMethod()>
    0 references
    Public Function Subtract(a As Double, b As Double) As String
        Return (a - b).ToString()
    End Function

    <WebMethod()>
    0 references
    Public Function Multiply(a As Double, b As Double) As String
        Return (a * b).ToString()
    End Function

    <WebMethod()>
    0 references
    Public Function Divide(a As Double, b As Double) As String
        If b = 0 Then
            Return "Cannot divide by zero"
        End If
        Return (a / b).ToString()
    End Function
End Class
```

Run :

MathService

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.)

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
};
```

MathService

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

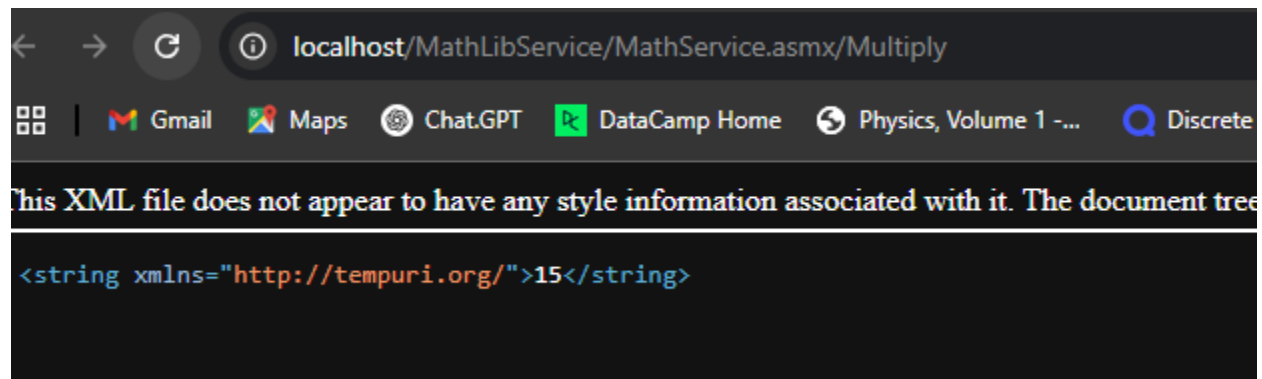
Multiply

Test

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

Parameter	Value
a:	5
b:	3

Output :



Q3) 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:

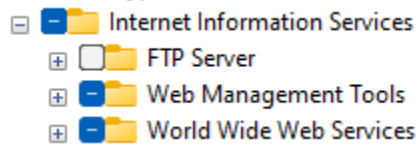
(a) MathLibService

(b) ServiceClient

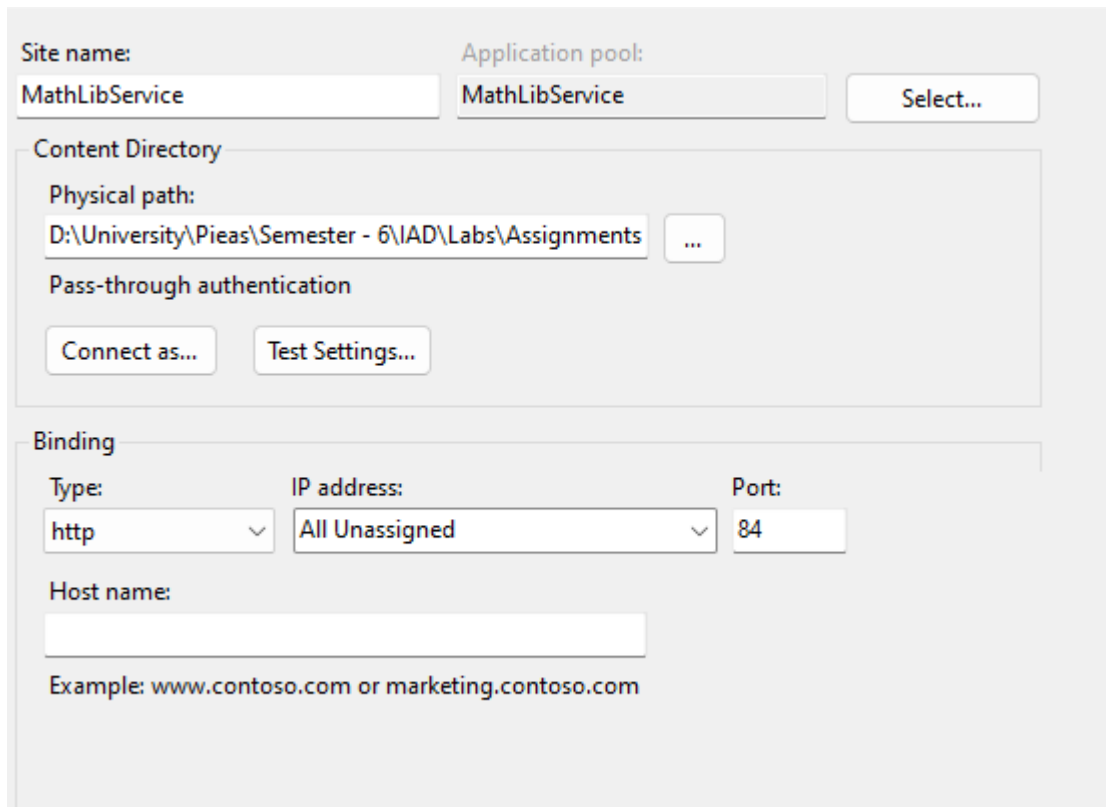
Answer :

- Install **IIS** via Control Panel > Turn Windows features on/off > Enable:

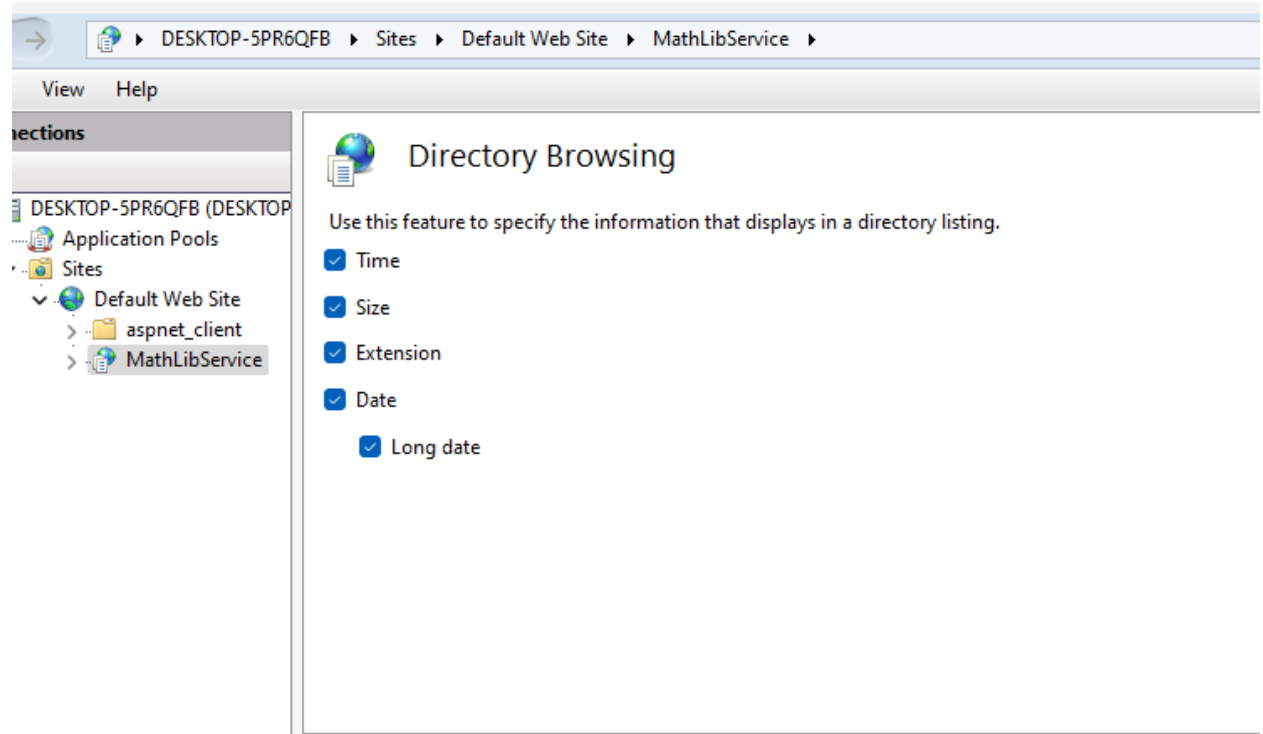
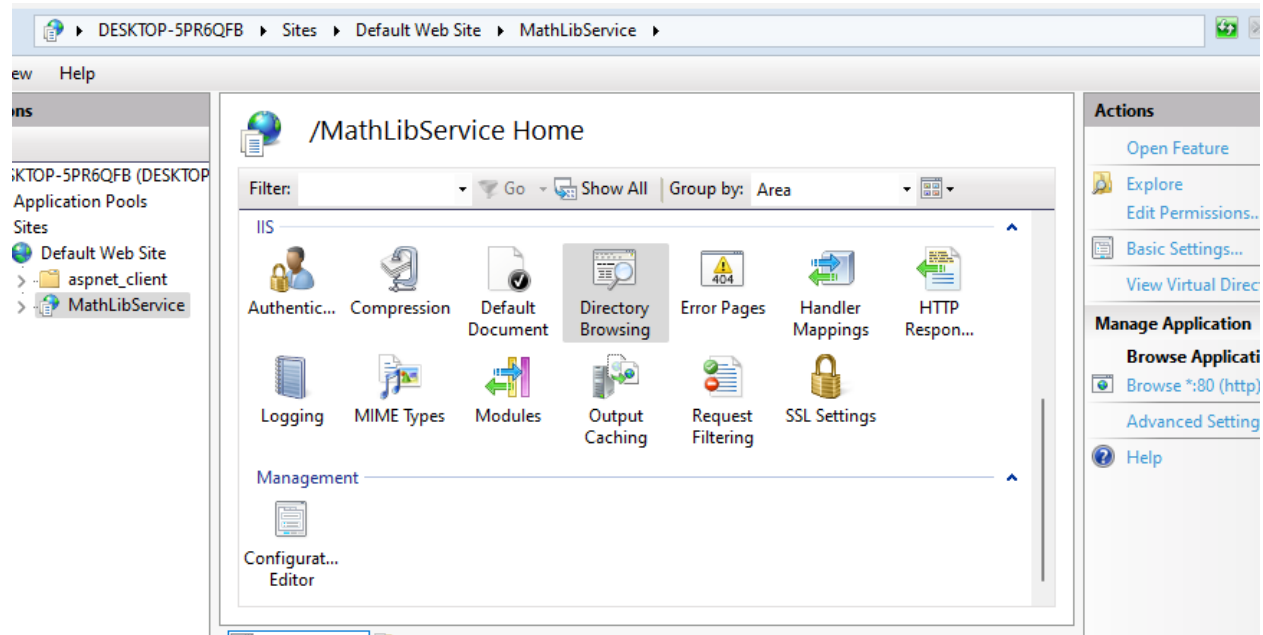
- Internet Information Services
- ASP.NET 4.8



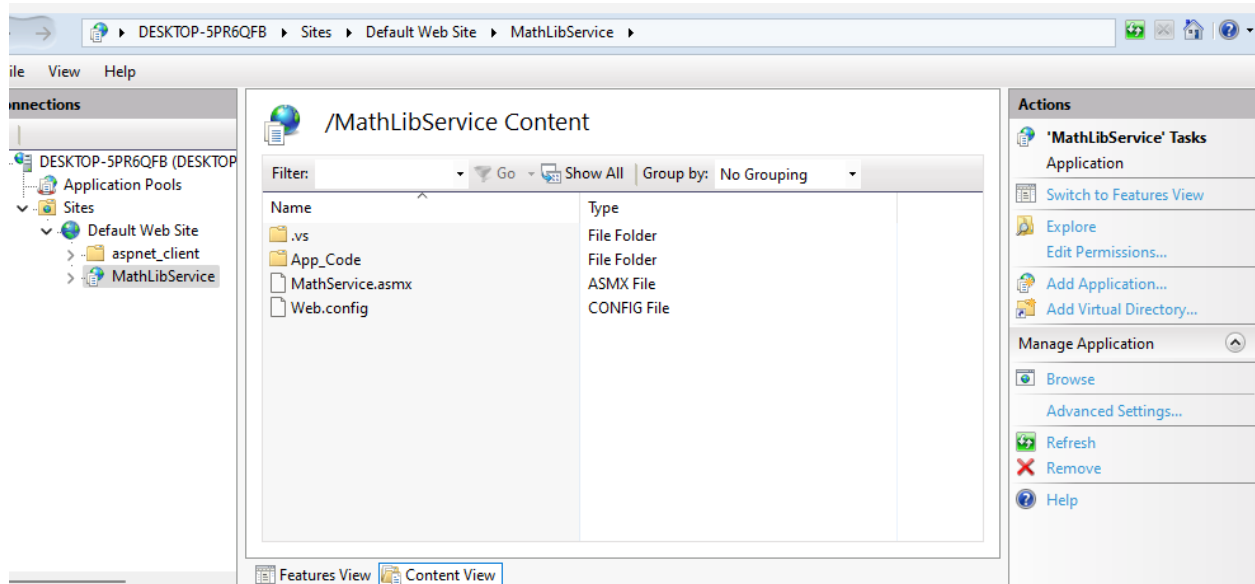
- Open **IIS Manager**
- Right-click Default Web Site → Add Application

A screenshot of the 'Add Application' dialog box in IIS Manager. The 'Site name' field is 'MathLibService'. The 'Application pool' is 'MathLibService'. The 'Content Directory' section shows the 'Physical path' as 'D:\University\Pieas\Semester - 6\IAD\Labs\Assignments'. The 'Binding' section shows 'Type' as 'http', 'IP address' as 'All Unassigned', and 'Port' as '84'. There is a 'Host name' field at the bottom with an example: 'www.contoso.com or marketing.contoso.com'.

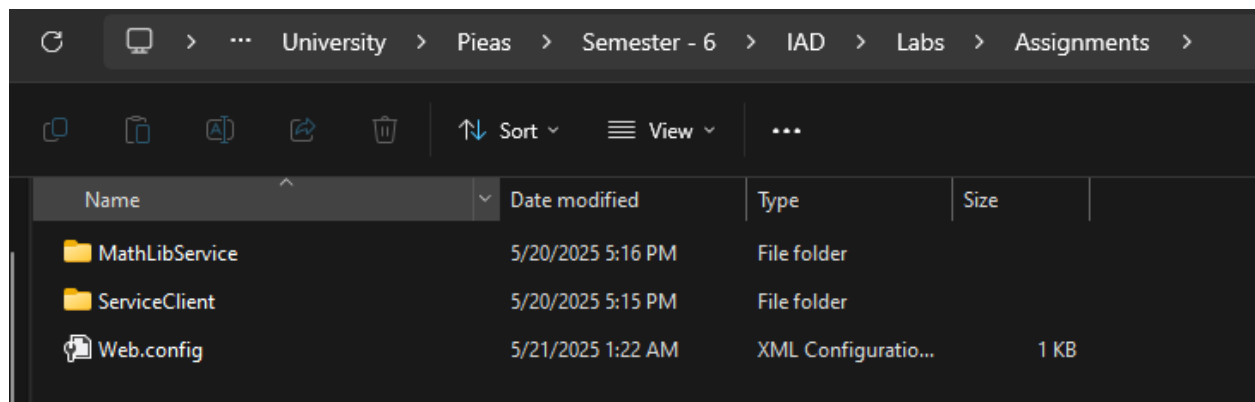
Directory Browsing Option :



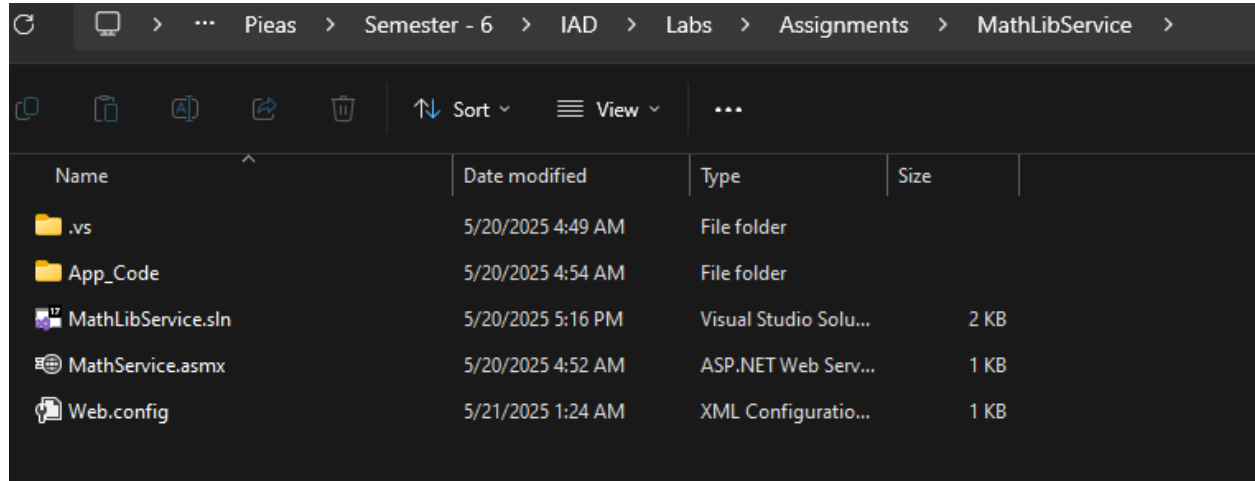
After All setup :



Assignment Folder Looks Like :



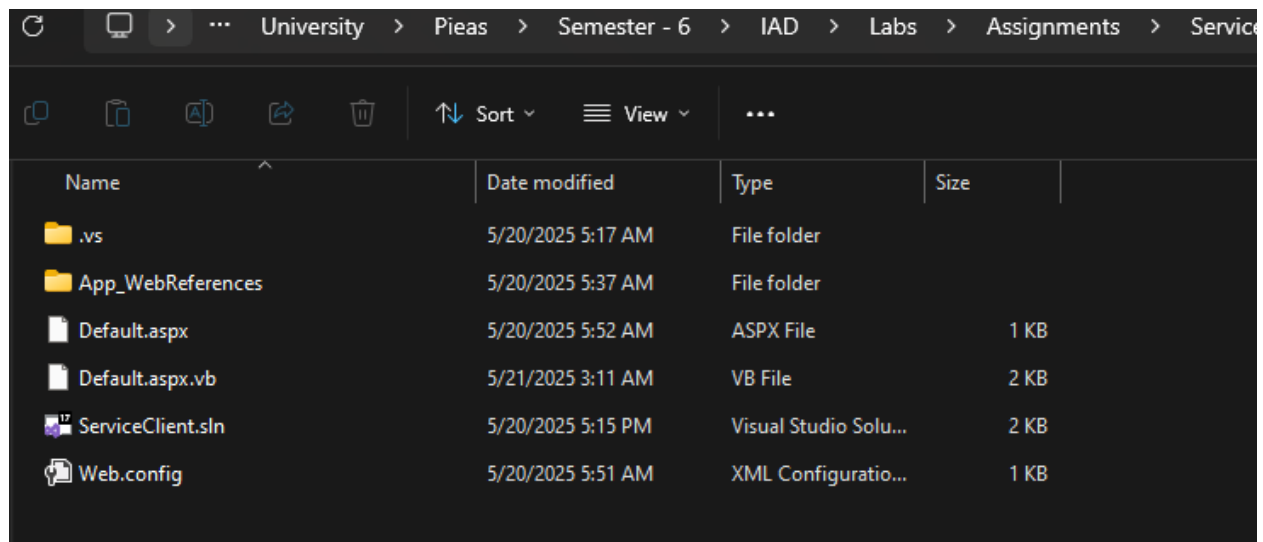
MathlibService folder :



The screenshot shows a Windows File Explorer window with the address bar path: Pieas > Semester - 6 > IAD > Labs > Assignments > MathLibService. The toolbar includes icons for copy, paste, delete, and a menu with Sort, View, and a three-dot menu. The file list is as follows:

Name	Date modified	Type	Size
.vs	5/20/2025 4:49 AM	File folder	
App_Code	5/20/2025 4:54 AM	File folder	
MathLibService.sln	5/20/2025 5:16 PM	Visual Studio Solu...	2 KB
MathService.asmx	5/20/2025 4:52 AM	ASP.NET Web Serv...	1 KB
Web.config	5/21/2025 1:24 AM	XML Configuratio...	1 KB

ServiceClient folder :



The screenshot shows a Windows File Explorer window with the address bar path: University > Pieas > Semester - 6 > IAD > Labs > Assignments > ServiceClient. The toolbar includes icons for copy, paste, delete, and a menu with Sort, View, and a three-dot menu. The file list is as follows:

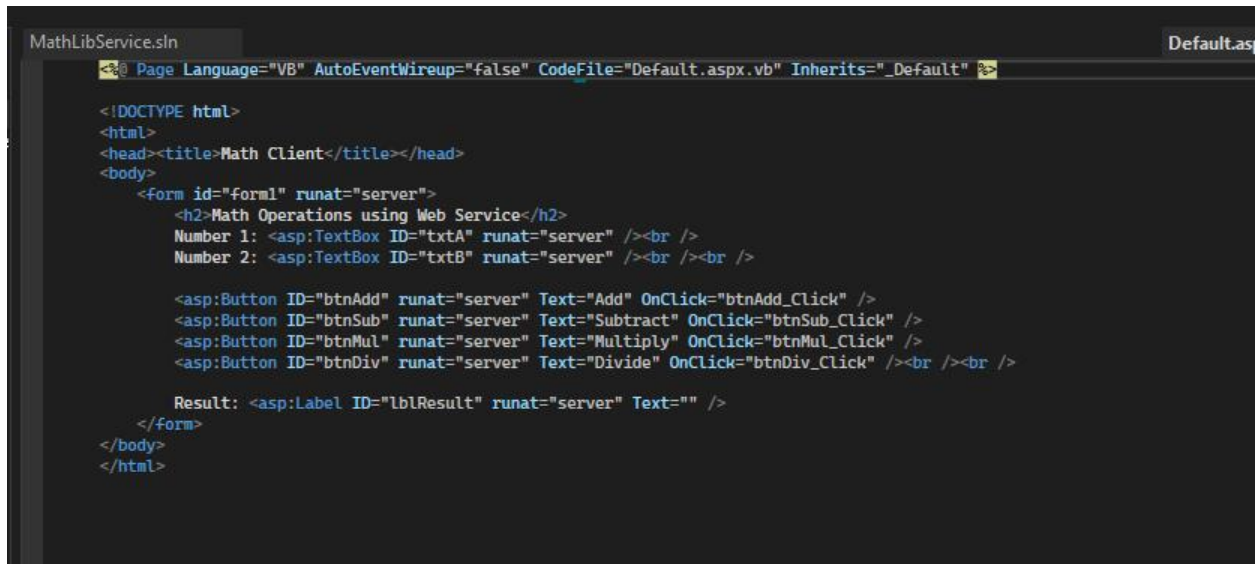
Name	Date modified	Type	Size
.vs	5/20/2025 5:17 AM	File folder	
App_WebReferences	5/20/2025 5:37 AM	File folder	
Default.aspx	5/20/2025 5:52 AM	ASPX File	1 KB
Default.aspx.vb	5/21/2025 3:11 AM	VB File	2 KB
ServiceClient.sln	5/20/2025 5:15 PM	Visual Studio Solu...	2 KB
Web.config	5/20/2025 5:51 AM	XML Configuratio...	1 KB

Q4) 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.

Answer :

In ServiceClient, I Created a file :

Default.aspx



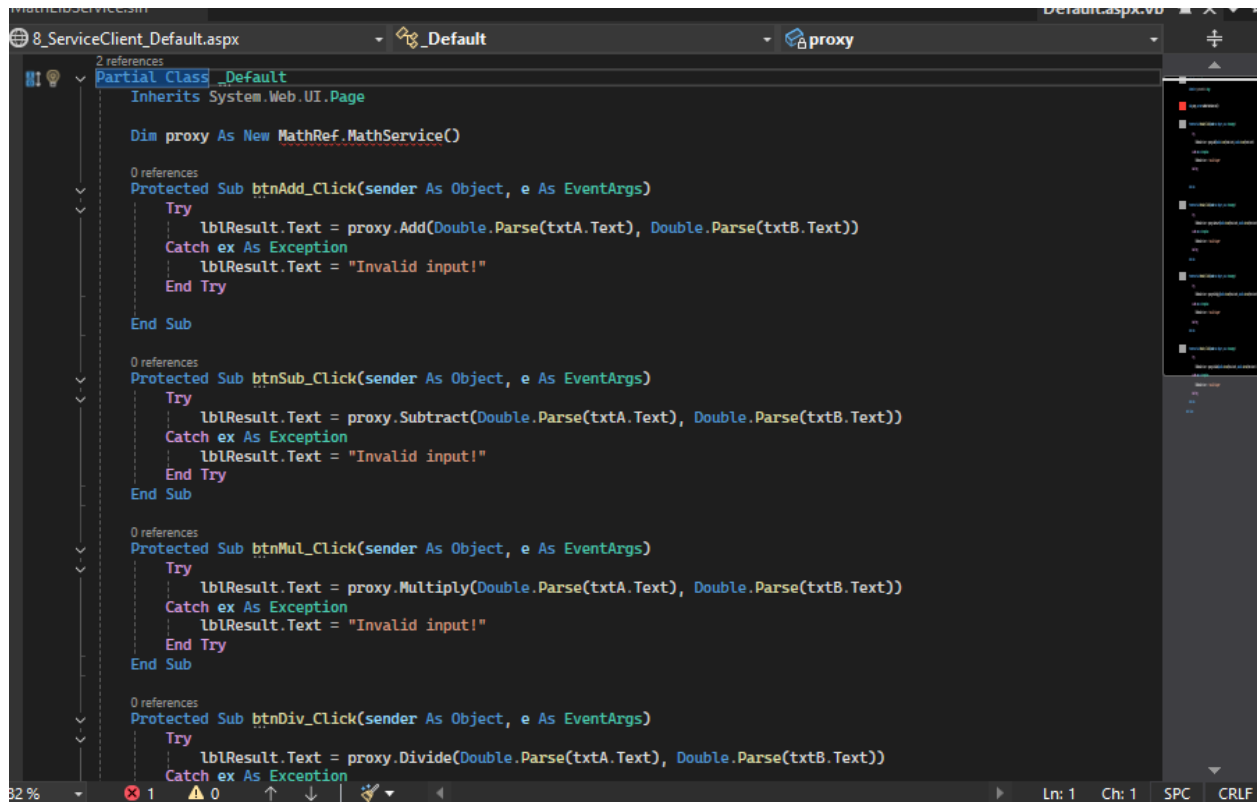
```
MathLibService.sln                                                                    Default.aspx
<@ Page Language="VB" AutoEventWireup="false" CodeFile="Default.aspx.vb" Inherits="_Default" >

<!DOCTYPE html>
<html>
<head><title>Math Client</title></head>
<body>
    <form id="form1" runat="server">
        <h2>Math Operations using Web Service</h2>
        Number 1: <asp:TextBox ID="txtA" runat="server" /><br />
        Number 2: <asp:TextBox ID="txtB" runat="server" /><br /><br />

        <asp:Button ID="btnAdd" runat="server" Text="Add" OnClick="btnAdd_Click" />
        <asp:Button ID="btnSub" runat="server" Text="Subtract" OnClick="btnSub_Click" />
        <asp:Button ID="btnMul" runat="server" Text="Multiply" OnClick="btnMul_Click" />
        <asp:Button ID="btnDiv" runat="server" Text="Divide" OnClick="btnDiv_Click" /><br /><br />

        Result: <asp:Label ID="lblResult" runat="server" Text="" />
    </form>
</body>
</html>
```

Default.aspx.vb :



```
Partial Class _Default
    Inherits System.Web.UI.Page

    Dim proxy As New MathRef.MathService()

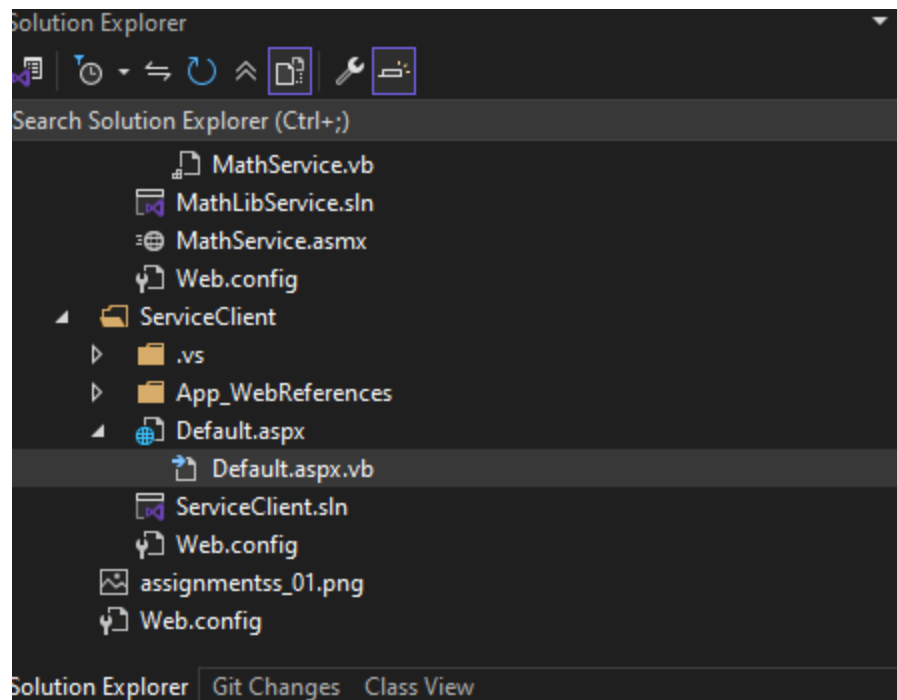
    0 references
    Protected Sub btnAdd_Click(sender As Object, e As EventArgs)
        Try
            lblResult.Text = proxy.Add(Double.Parse(txtA.Text), Double.Parse(txtB.Text))
        Catch ex As Exception
            lblResult.Text = "Invalid input!"
        End Try
    End Sub

    0 references
    Protected Sub btnSub_Click(sender As Object, e As EventArgs)
        Try
            lblResult.Text = proxy.Subtract(Double.Parse(txtA.Text), Double.Parse(txtB.Text))
        Catch ex As Exception
            lblResult.Text = "Invalid input!"
        End Try
    End Sub

    0 references
    Protected Sub btnMul_Click(sender As Object, e As EventArgs)
        Try
            lblResult.Text = proxy.Multiply(Double.Parse(txtA.Text), Double.Parse(txtB.Text))
        Catch ex As Exception
            lblResult.Text = "Invalid input!"
        End Try
    End Sub

    0 references
    Protected Sub btnDiv_Click(sender As Object, e As EventArgs)
        Try
            lblResult.Text = proxy.Divide(Double.Parse(txtA.Text), Double.Parse(txtB.Text))
        Catch ex As Exception
        End Try
    End Sub
End Class
```

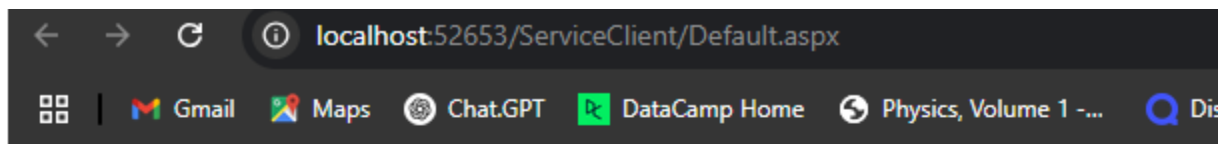
The Directory Structure looks like :



Output :

Functionalites:

Add :



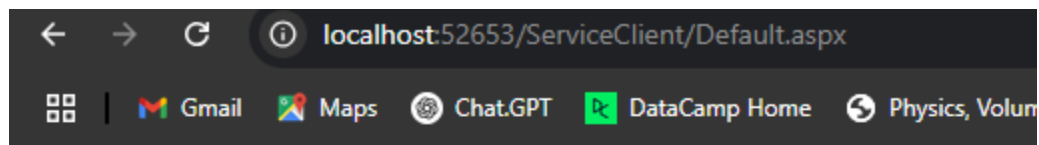
Math Operations using Web Service

Number 1:

Number 2:



Result:

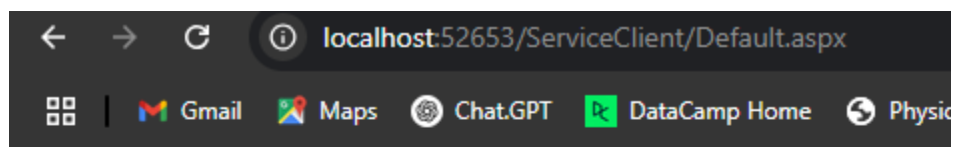


Math Operations using Web Service

Number 1:
Number 2:

Result: 15

Divide:



Math Operations using Web Service

Number 1:
Number 2:

Result: 15

Math Operations using Web Service

Number 1:

Number 2:

Result: 3
