

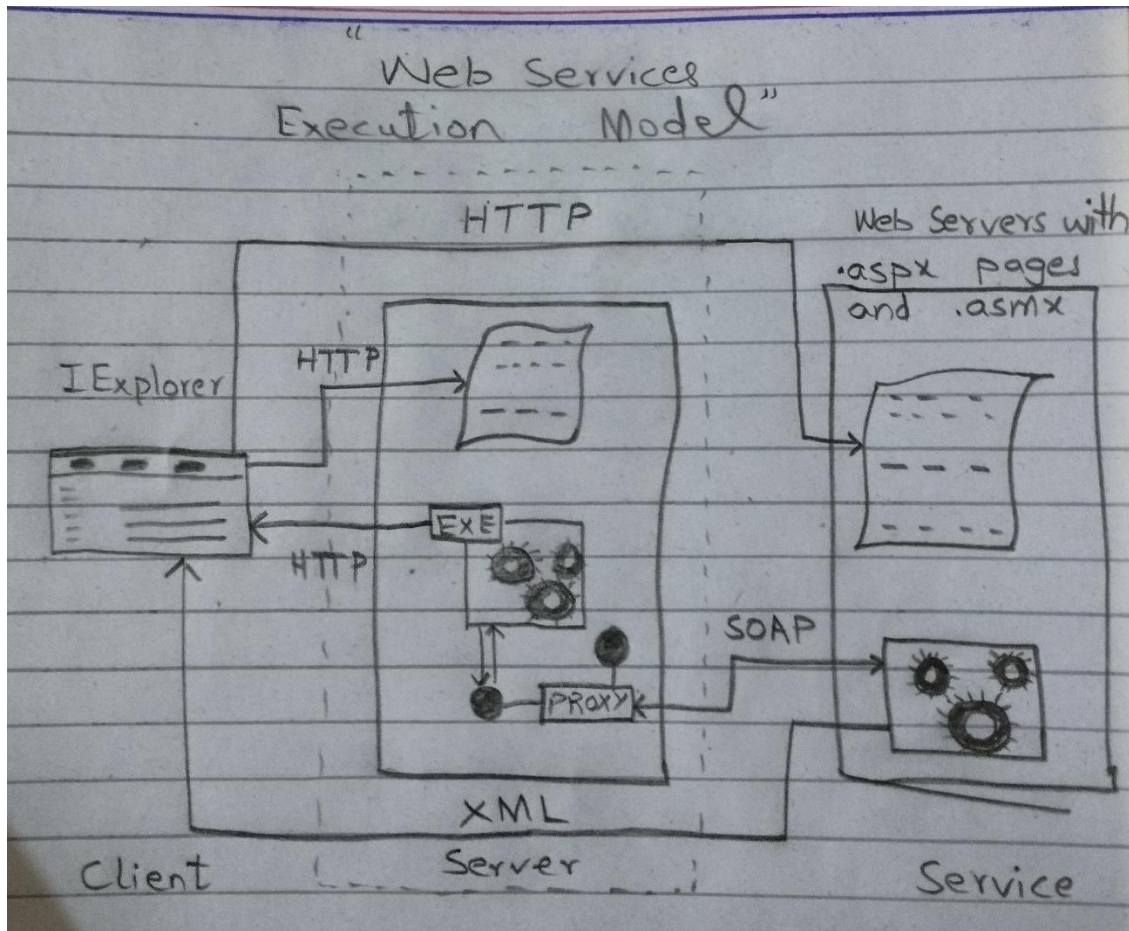


Name: Waleed Abbasi

Course: Internet Application Development

Assignment: WebServices

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**
- ?**

Lets create an application with a webform MathLibService.aspx and .vb.

MathLibService.asmx

```
<%@ WebService Language="VB" Class="MathLibService" %>
```

MathLibService.vb

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

```
<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(a As Double, b As Double) As String
```

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

```
End Function
```

<WebMethod>

Public Function Subtract(a As Double, b As Double) As String

Return (a - b).ToString()

End Function

<WebMethod>

Public Function Multiply(a As Double, b As Double) As String

Return (a * b).ToString()

End Function

<WebMethod>

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

Let's Install IIS:

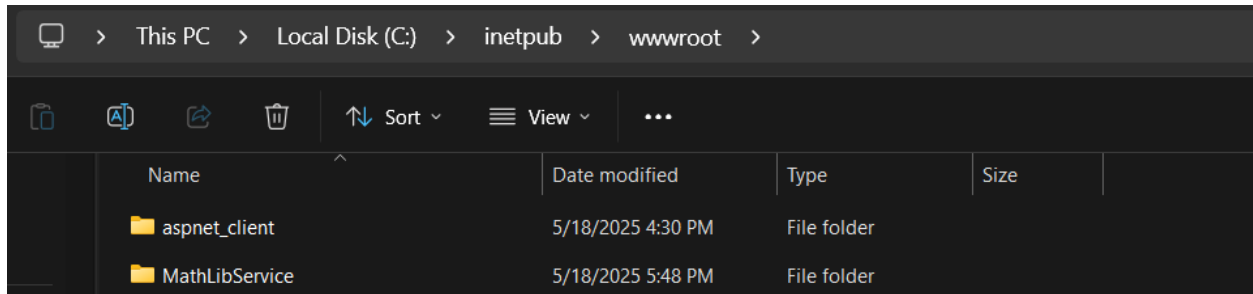
Open "**Turn Windows features on or off**" and enable following options:

1. Internet Information Services
2. ASP.NET (under .NET Framework 4.x)

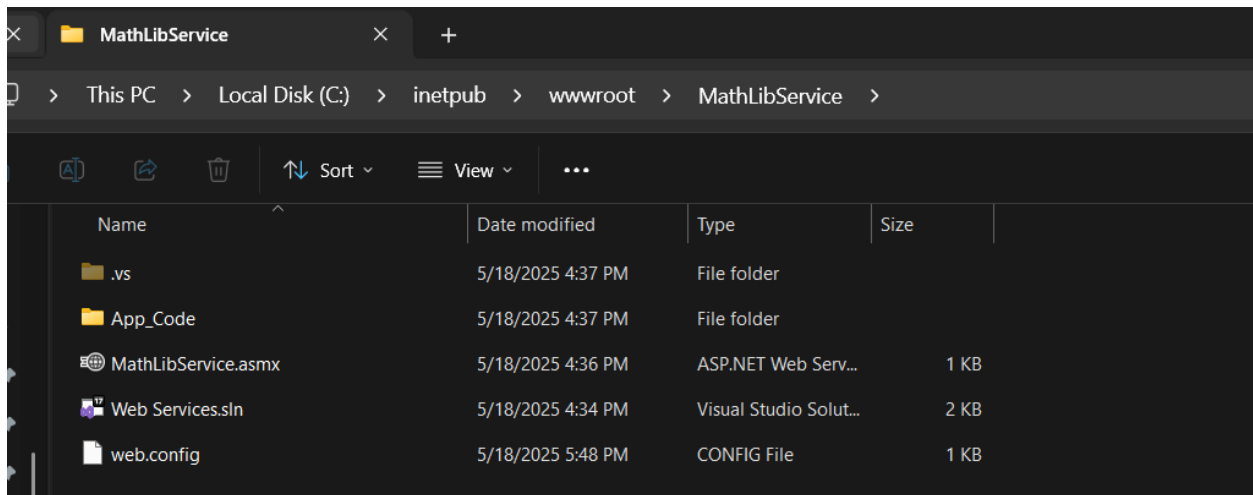
3. Application Development Features → ASP.NET, ISAPI Extensions, ISAPI Filters

This will install IIS on our machine.

I created a folder named MathLibService on root directory of IIS



And inside that folder I copied all the files



Then I opened IIS Manager

Add Website

Site name: MathLibService Application pool: MathLibService Select...

Content Directory

Physical path: C:\inetpub\wwwroot\MathLibService ...

Pass-through authentication

Connect as... Test Settings...

Binding

Type: http IP address: All Unassigned Port: 6767

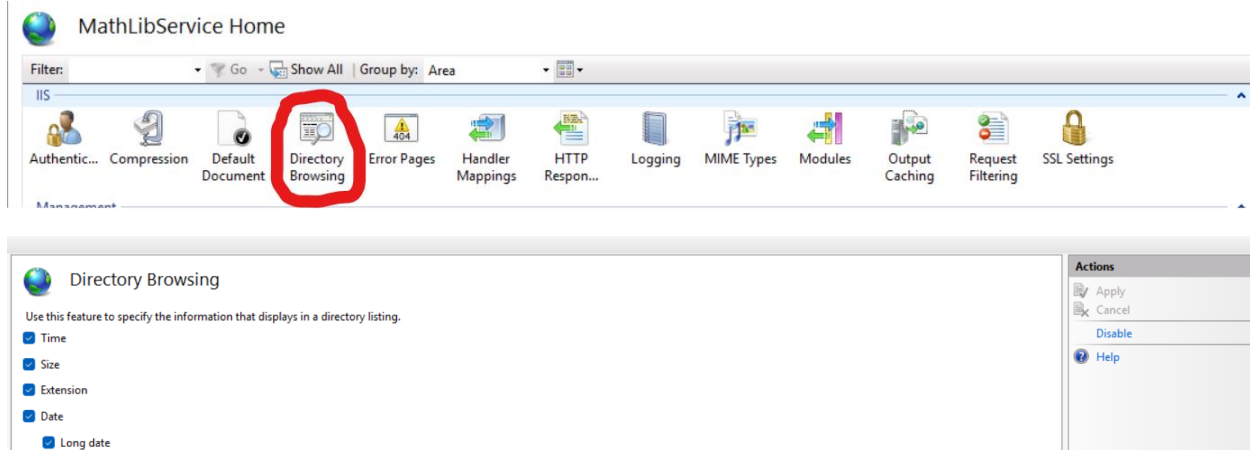
Host name:

Example: www.contoso.com or marketing.contoso.com

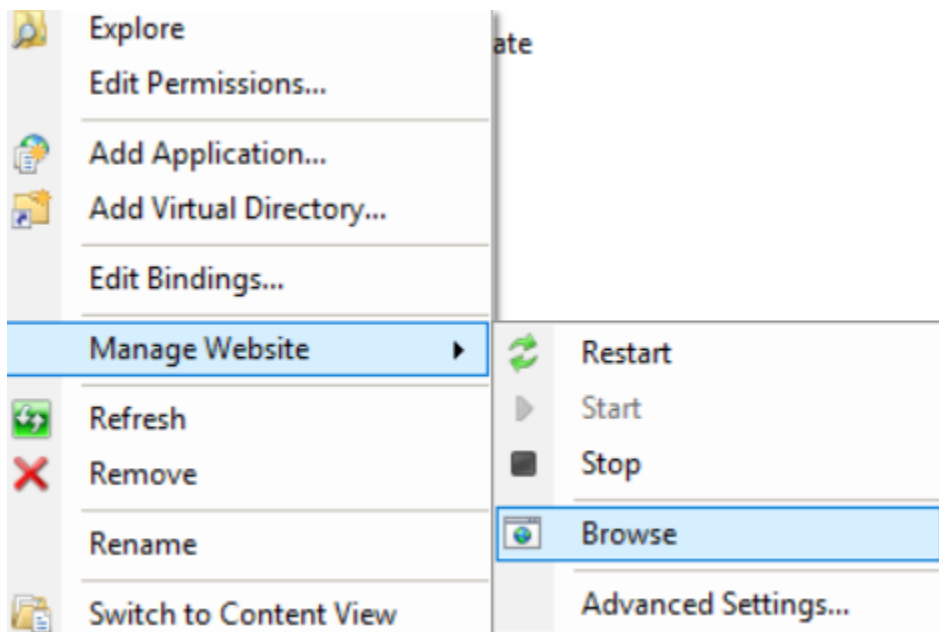
☒ Start Website immediately

OK Cancel

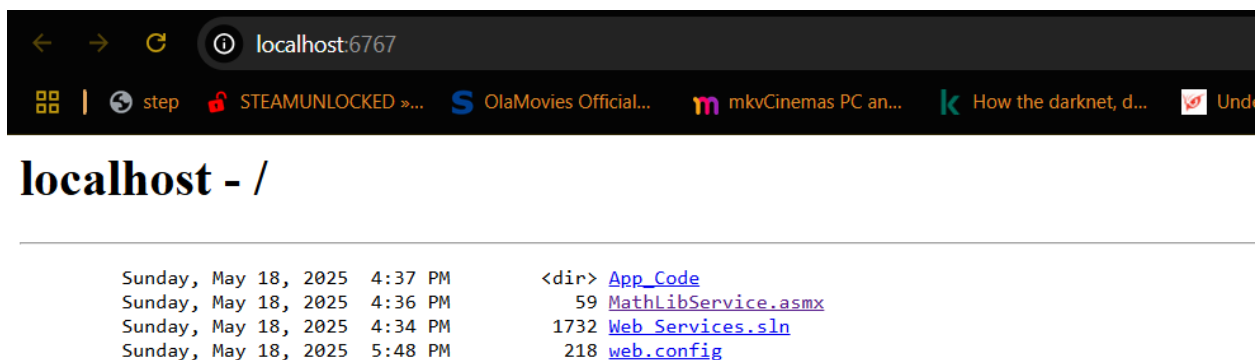
I also enabled the Directory Browsing Option



Then As we Right click our application and click on Browse.



It shows



Now as we click on MathLibService.asmx it shows:

localhost6767/MathLibService.asmx

MathLibService

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 on the Web. (XML Web service namespaces are URIs.)

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 name 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 applied to the class that contains 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](#).

Lets check the Add feature:

MathLibService

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

Add

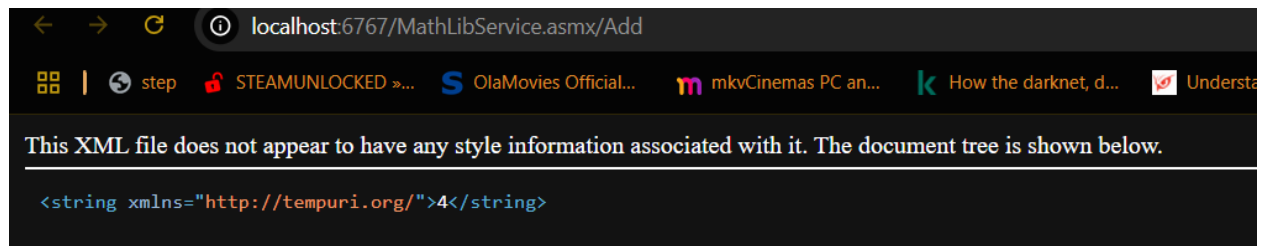
Test

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

Parameter	Value
a:	1
b:	3

Invoke

As you can see that it is giving us a single string:



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

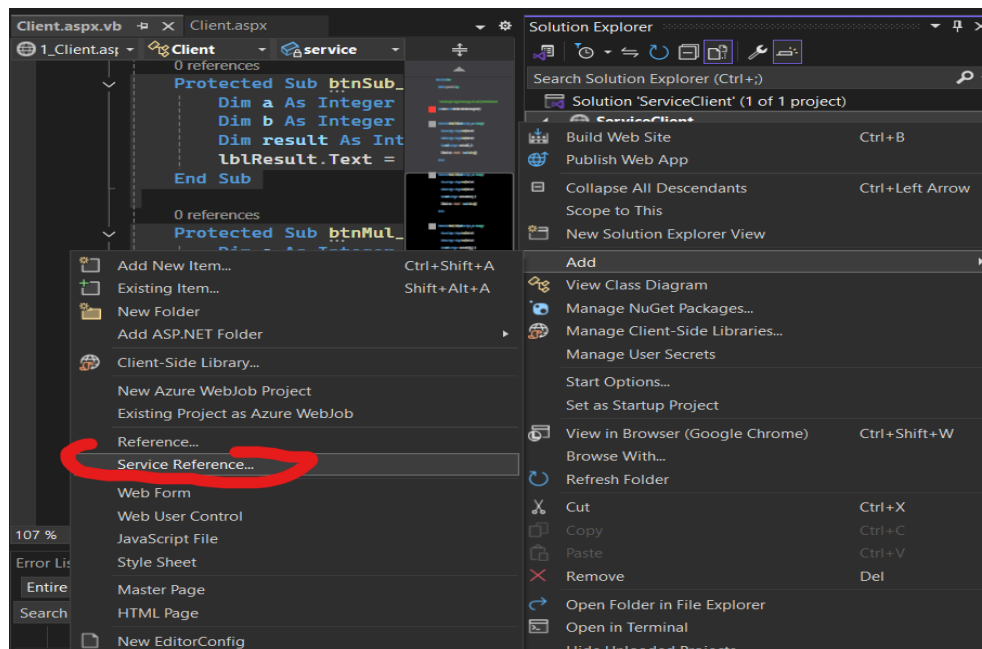
(b) ServiceClient

As I already installed IIS by enabling following options in **“Turn Windows features on or off”**:

1. Internet Information Services
2. ASP.NET (under .NET Framework 4.x)
3. Application Development Features → ASP.NET, ISAPI Extensions, ISAPI Filters

As we created MathLibService already, lets cerate ServiceClient (another application).

Lets add Service Reference



Add Service Reference ? X

To see a list of available services on a specific server, enter a service URL and click Go. To browse for available services, click Discover.

Address:

Go Discover

Services: Operations:

- MathLibService - MathLibServiceSoap	- Add - Divide - Multiply - Subtract
--	--

1 service(s) found at address 'http://localhost/MathLibService/MathLibService.asmx'.

Namespace:

Advanced... OK Cancel

Click OK.

I created a Client.aspx webform.

Client.aspx

```
<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Client.aspx.vb" Inherits="Client" %>
```

```

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

    <title>MathLib Service Client</title>

</head>

<body>

    <form id="form1" runat="server">

        <div>

            <h2>Math Operations using Web Service</h2>

            <asp:Label ID="lblA" runat="server" Text="Enter A: " />

            <asp:TextBox ID="txtA" runat="server" /><br /><br />

            <asp:Label ID="lblB" runat="server" Text="Enter B: " />

            <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 />

            <asp:Label ID="lblResult" runat="server" Font-Bold="True" />

        </div>

    </form>

</body>

</html>

```

Client.aspx.vb

Partial Class Client

Inherits System.Web.UI.Page

' Create the object using the correct proxy class name (from Add Web Reference)

Dim service As New MathLibRef.MathLibServiceSoapClient()

Protected Sub btnAdd_Click(sender As Object, e As EventArgs)

Dim a As Integer = Integer.Parse(txtA.Text)

Dim b As Integer = Integer.Parse(txtB.Text)

Dim result As Integer = service.Add(a, b)

lblResult.Text = "Result: " & result.ToString()

End Sub

Protected Sub btnSub_Click(sender As Object, e As EventArgs)

Dim a As Integer = Integer.Parse(txtA.Text)

Dim b As Integer = Integer.Parse(txtB.Text)

Dim result As Integer = service.Subtract(a, b)

lblResult.Text = "Result: " & result.ToString()

End Sub

Protected Sub btnMul_Click(sender As Object, e As EventArgs)

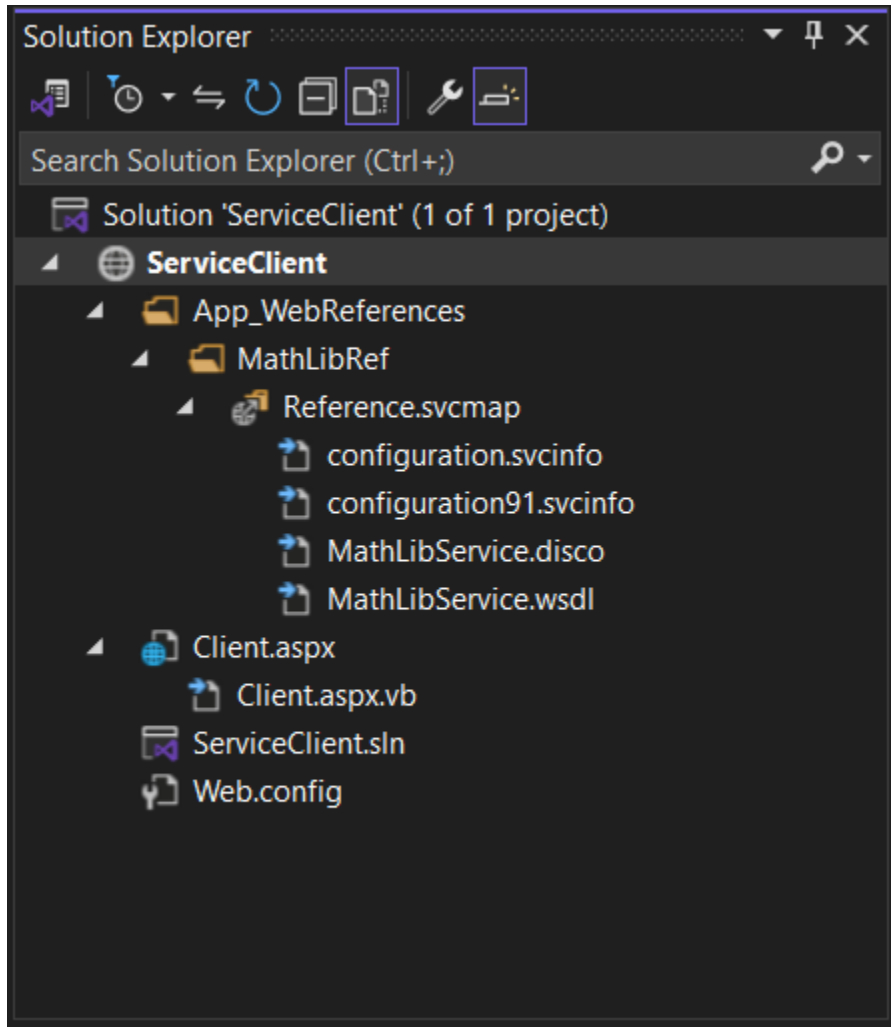
Dim a As Integer = Integer.Parse(txtA.Text)

Dim b As Integer = Integer.Parse(txtB.Text)

```
    Dim result As Integer = service.Multiply(a, b)
    lblResult.Text = "Result: " & result.ToString()
End Sub

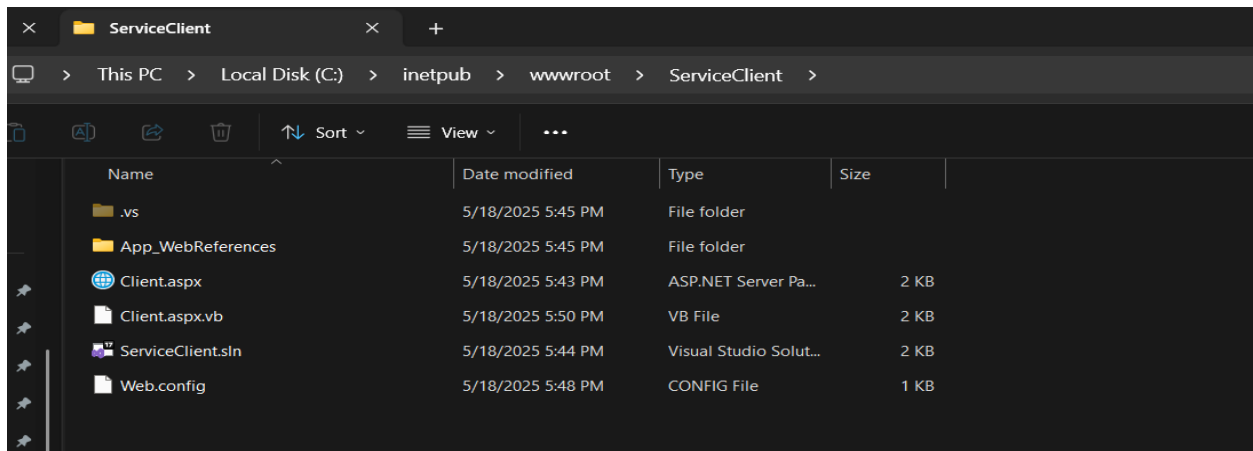
Protected Sub btnDiv_Click(sender As Object, e As EventArgs)
    Dim a As Integer = Integer.Parse(txtA.Text)
    Dim b As Integer = Integer.Parse(txtB.Text)
    Dim result As Integer = service.Divide(a, b)
    lblResult.Text = "Result: " & result.ToString()
End Sub
End Class
```


The Directory Structure looks like this:



Now lets create application on IIS

I copied files to C:\inetpub\wwwroot\ServiceClient so that IIS can access it



Now open IIS Manager and add another application:

Add Website?×

Site name:
ServiceClient

Application pool:
ServiceClient

Select...

Content Directory

Physical path:
C:\inetpub\wwwroot\ServiceClient

...

Pass-through authentication

Connect as...

Test Settings...

Binding

Type:
http

IP address:
All Unassigned

Port:
9494

Host name:

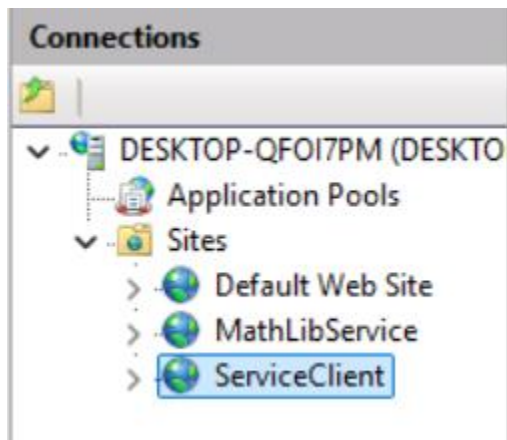
Example: www.contoso.com or marketing.contoso.com

☒ Start Website immediately

OK

Cancel

As you can see that there are two applications on IIS:



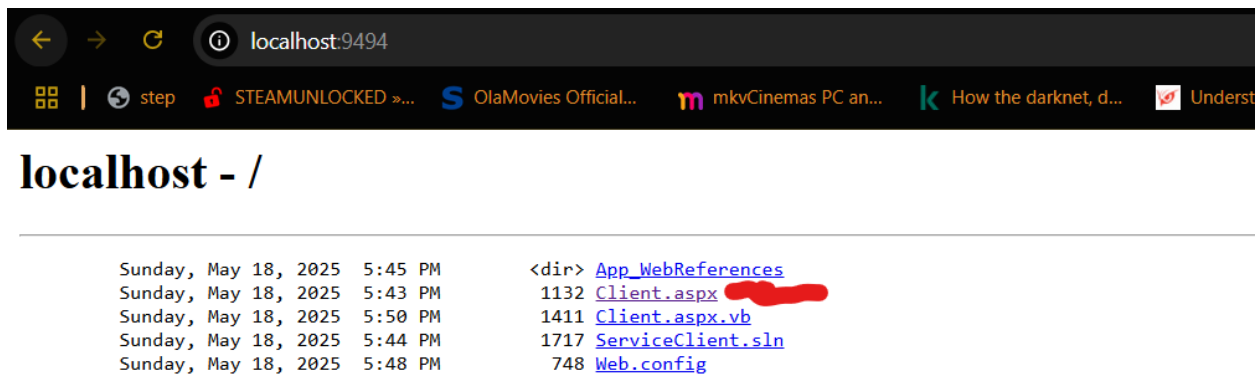
Q4) Implement a web application “ServiceClient” which will use the MathLibService as a web service. Develop a suitable web page for its demonstration. Use some text boxes to take input from user. Use proxy class to get answer (response from web service) and display it on page to user.

We already implemented the web applications in Question 2 and 3.

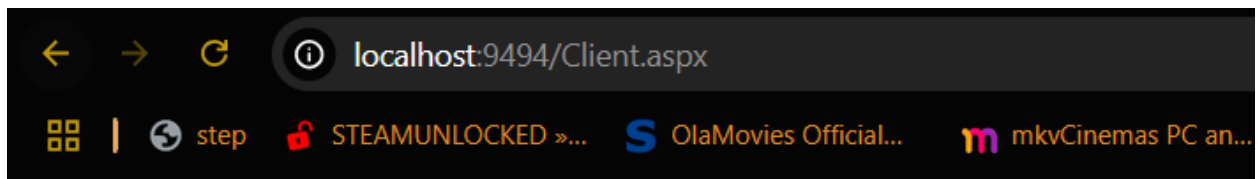
Now lets test:



As I enter the URL it shows all the files.



As I click on Client.aspx, it displays:



Math Operations using Web Service

Enter A:

Enter B:

ADD:

Math Operations using Web Service

Enter A:

Enter B:

Result: 12

Subtract:

Math Operations using Web Service

Enter A:

Enter B:

Add

Subtract

Multiply

Divide

Result: 8

Multiply:

Math Operations using Web Service

Enter A:

Enter B:

Add

Subtract

Multiply

Divide

Result: 20

Division:

Math Operations using Web Service

Enter A:

Enter B:

Add

Subtract

Multiply

Divide

Result: 20