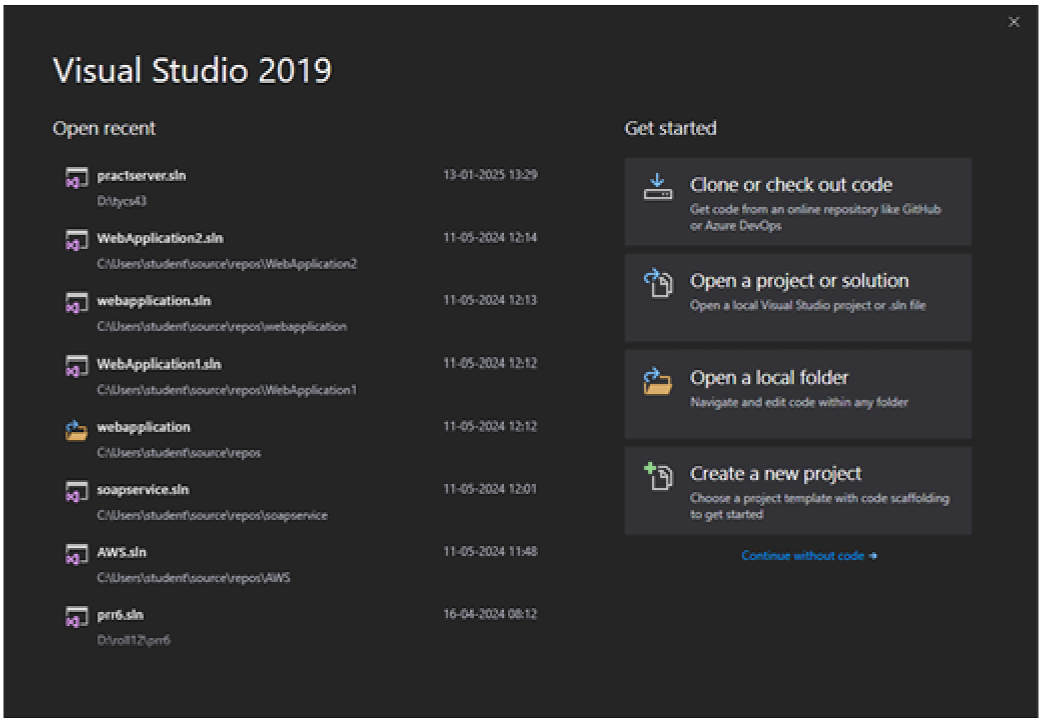
Practical No.6

Aim: - Develop application to download image/video from server or upload image/video to server using MTOM techniques.

Step 1: -Open Visual Studio 2019


Visual Studio 2019

Open recent

pract1server.sln D:\tycs43	13-01-2025 13:29
WebApplication2.sln C:\Users\student\source\repos\WebApplication2	11-05-2024 12:14
webapplication.sln C:\Users\student\source\repos\webapplication	11-05-2024 12:13
WebApplication1.sln C:\Users\student\source\repos\WebApplication1	11-05-2024 12:12
webapplication C:\Users\student\source\repos	11-05-2024 12:12
soapservice.sln C:\Users\student\source\repos\soapservice	11-05-2024 12:01
AWS.sln C:\Users\student\source\repos\AWS	11-05-2024 11:48
prr6.sln D:\roll12\prr6	16-04-2024 08:12

Get started

Clone or check out code
Get code from an online repository like GitHub or Azure DevOps

Open a project or solution
Open a local Visual Studio project or .sln file

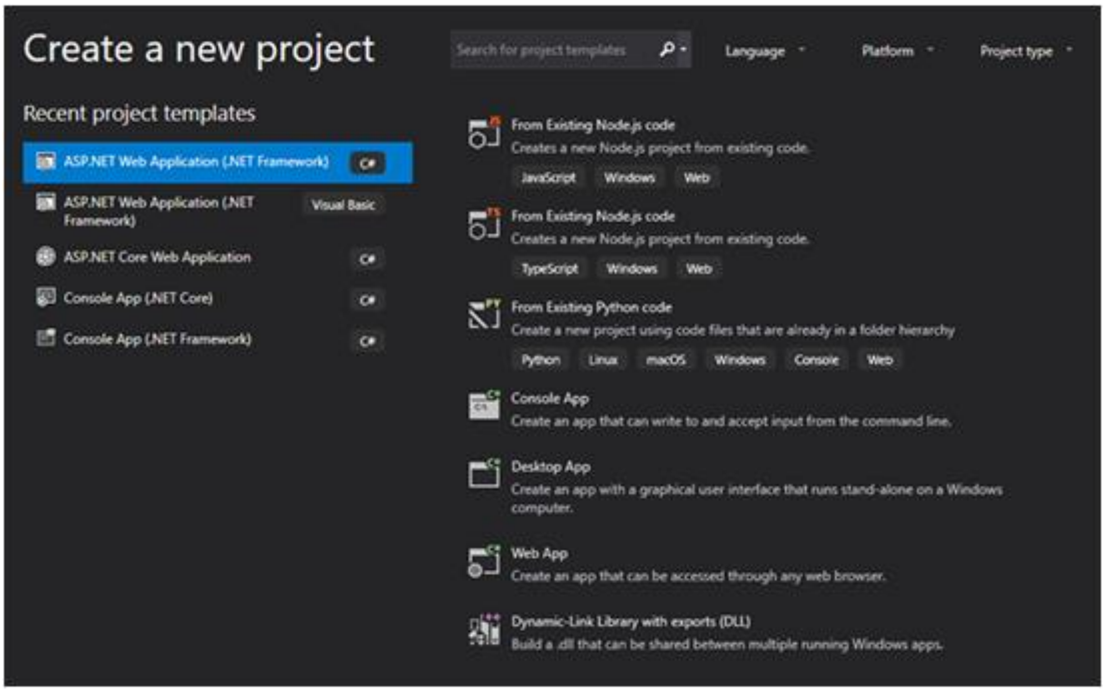
Open a local folder
Navigate and edit code within any folder

Create a new project
Choose a project template with code scaffolding to get started

Continue without code →

Step 2: -Create a new Project

Select ASP.NET Web Application (.NET framework)


Create a new project

Recent project templates

- ASP.NET Web Application (.NET Framework) C#
- ASP.NET Web Application (.NET Framework) Visual Basic
- ASP.NET Core Web Application C#
- Console App (.NET Core) C#
- Console App (.NET Framework) C#

Search for project templates | Language | Platform | Project type

From Existing Node.js code
Creates a new Node.js project from existing code.
JavaScript Windows Web

From Existing Node.js code
Creates a new Node.js project from existing code.
TypeScript Windows Web

From Existing Python code
Create a new project using code files that are already in a folder hierarchy
Python Linux macOS Windows Console Web

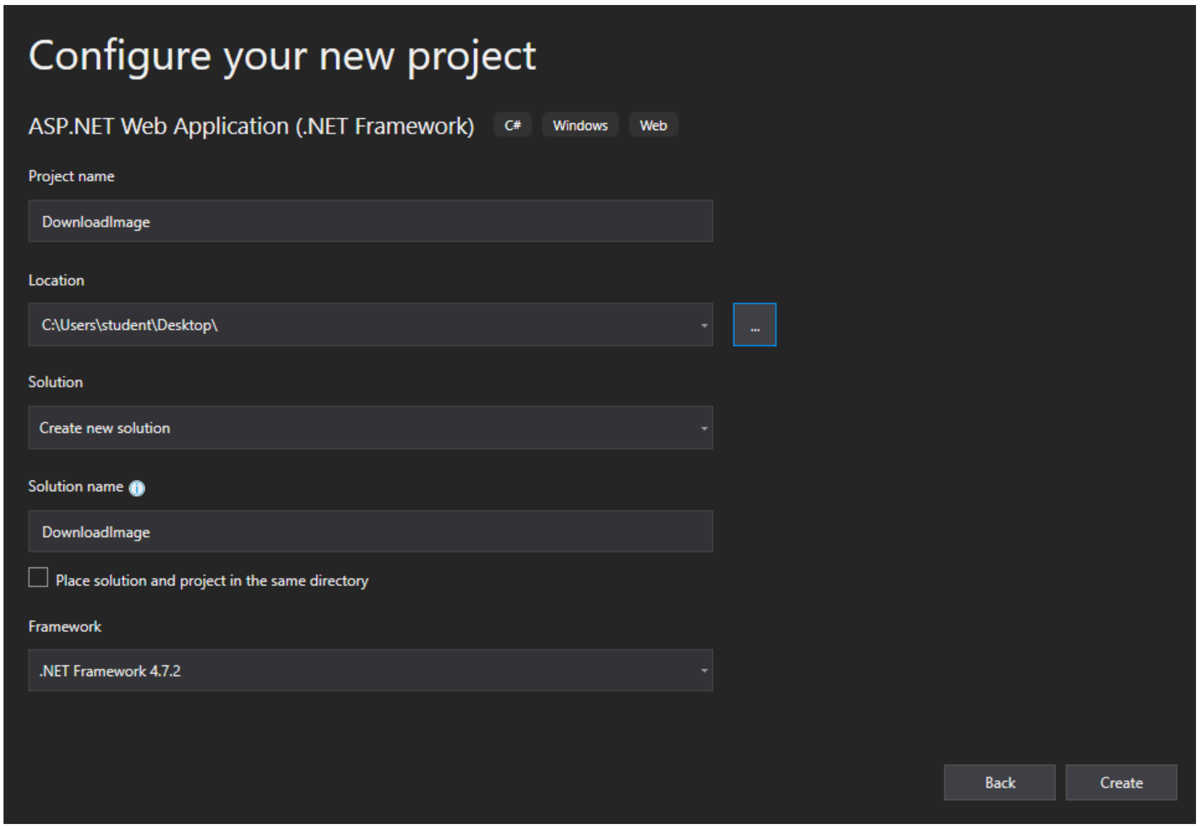
Console App
Create an app that can write to and accept input from the command line.

Desktop App
Create an app with a graphical user interface that runs stand-alone on a Windows computer.

Web App
Create an app that can be accessed through any web browser.

Dynamic-Link Library with exports (DLL)
Build a .dll that can be shared between multiple running Windows apps.

Step 3: -Give project name and click create



Configure your new project

ASP.NET Web Application (.NET Framework) C# Windows Web

Project name: DownloadImage

Location: C:\Users\student\Desktop\

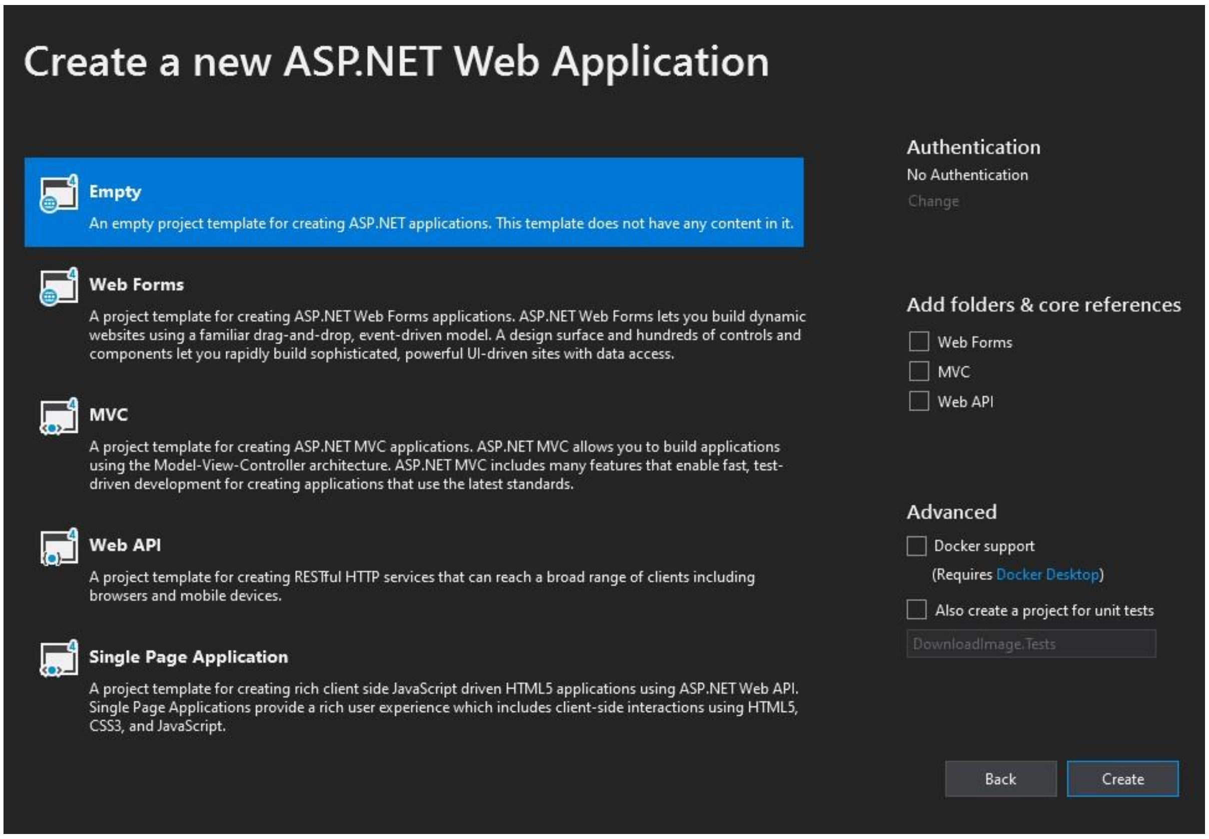
Solution: Create new solution

Solution name: DownloadImage

☐ Place solution and project in the same directory

Framework: .NET Framework 4.7.2

Step 4: -Select Empty and click create



Create a new ASP.NET Web Application

Empty
An empty project template for creating ASP.NET applications. This template does not have any content in it.

Web Forms
A project template for creating ASP.NET Web Forms applications. ASP.NET Web Forms lets you build dynamic websites using a familiar drag-and-drop, event-driven model. A design surface and hundreds of controls and components let you rapidly build sophisticated, powerful UI-driven sites with data access.

MVC
A project template for creating ASP.NET MVC applications. ASP.NET MVC allows you to build applications using the Model-View-Controller architecture. ASP.NET MVC includes many features that enable fast, test-driven development for creating applications that use the latest standards.

Web API
A project template for creating RESTful HTTP services that can reach a broad range of clients including browsers and mobile devices.

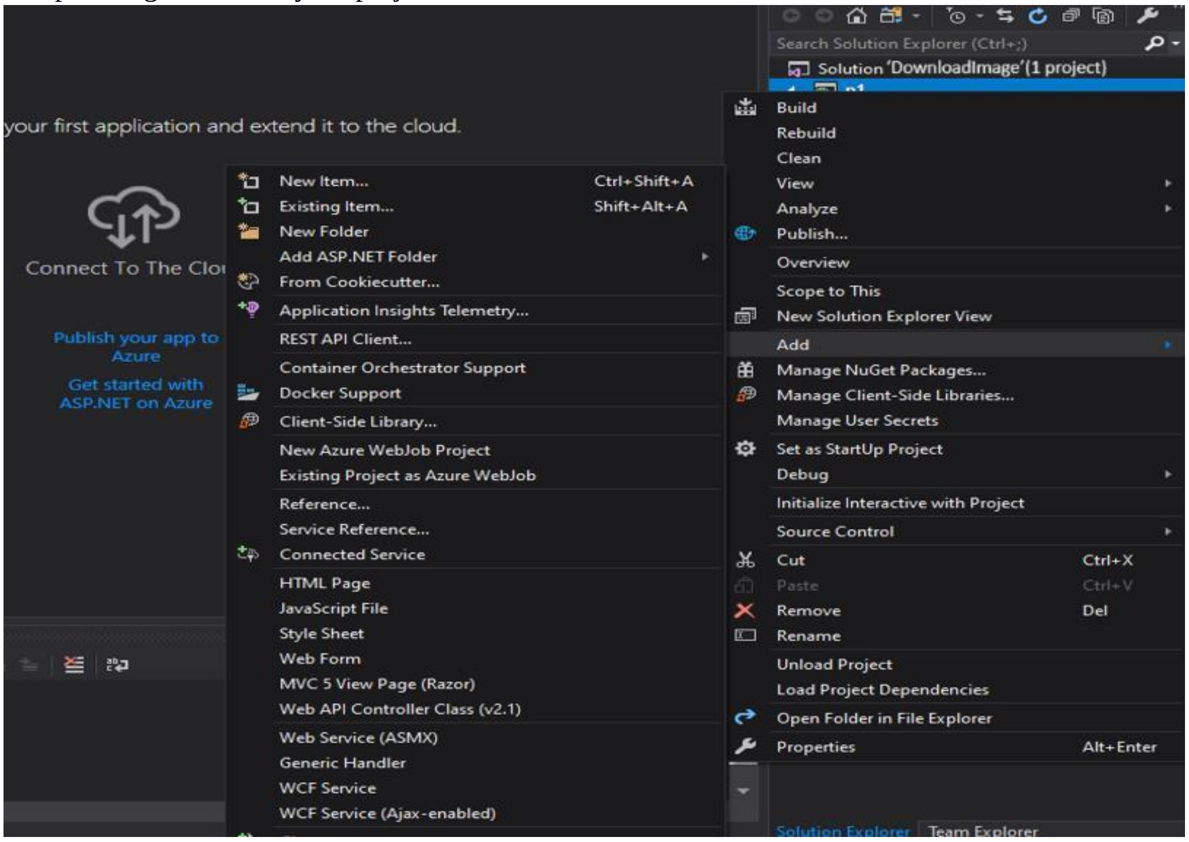
Single Page Application
A project template for creating rich client side JavaScript driven HTML5 applications using ASP.NET Web API. Single Page Applications provide a rich user experience which includes client-side interactions using HTML5, CSS3, and JavaScript.

Authentication
No Authentication
Change

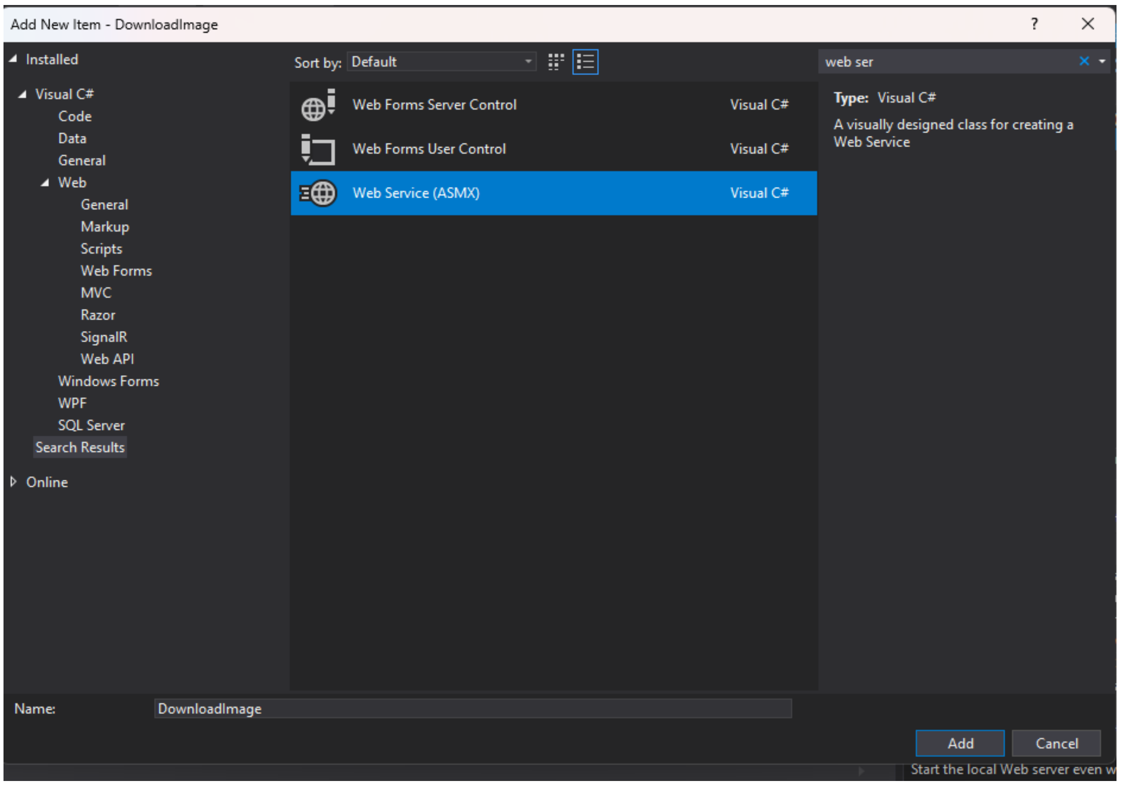
Add folders & core references
☐ Web Forms
☐ MVC
☐ Web API

Advanced
☐ Docker support
(Requires Docker Desktop)
☐ Also create a project for unit tests
DownloadImage.Tests

Step 5: -Right Click on your project name > Go to Add > New Item



Step6: - Create a Web Service
Click on Web Service (ASMX) > Give Name to the web service as DownloadImage.asmx

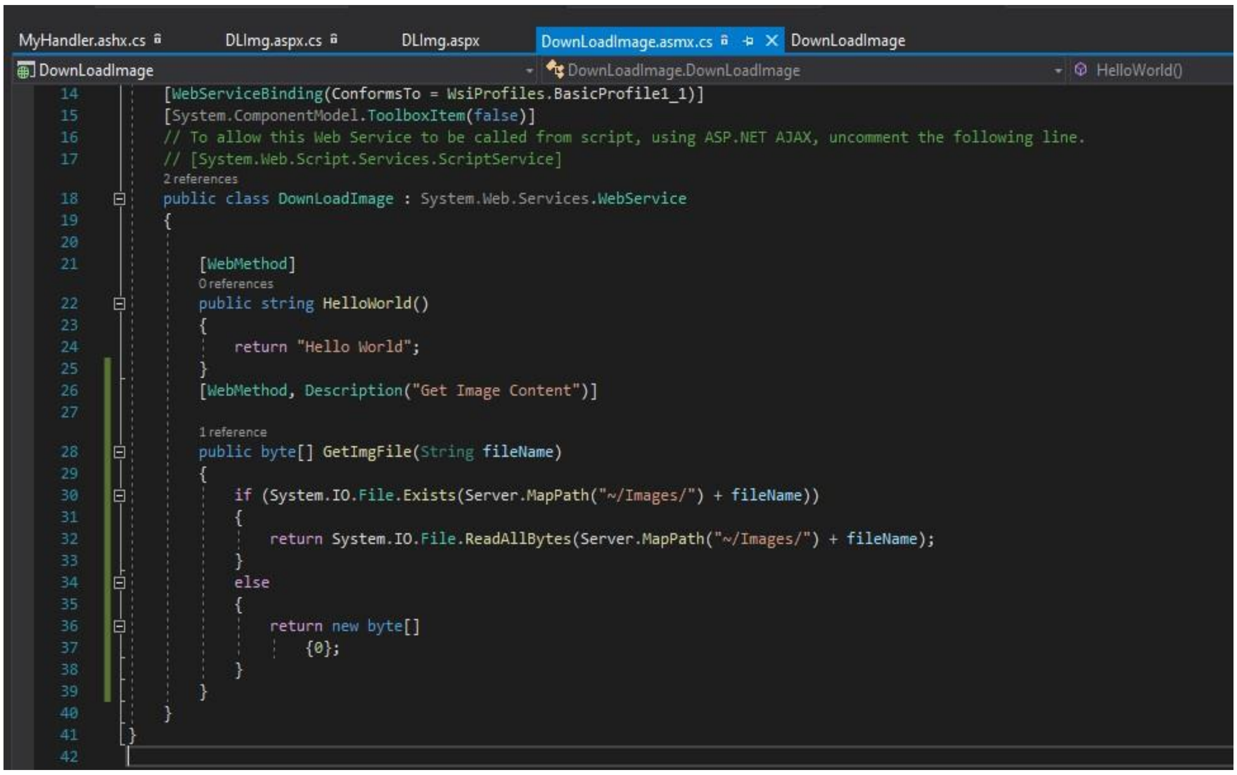


Type: Visual C#

A visually designed class for creating a Web Service

Name: DownloadImage

Step 7: -Write this code in your DownloadImage.asmx.cs file



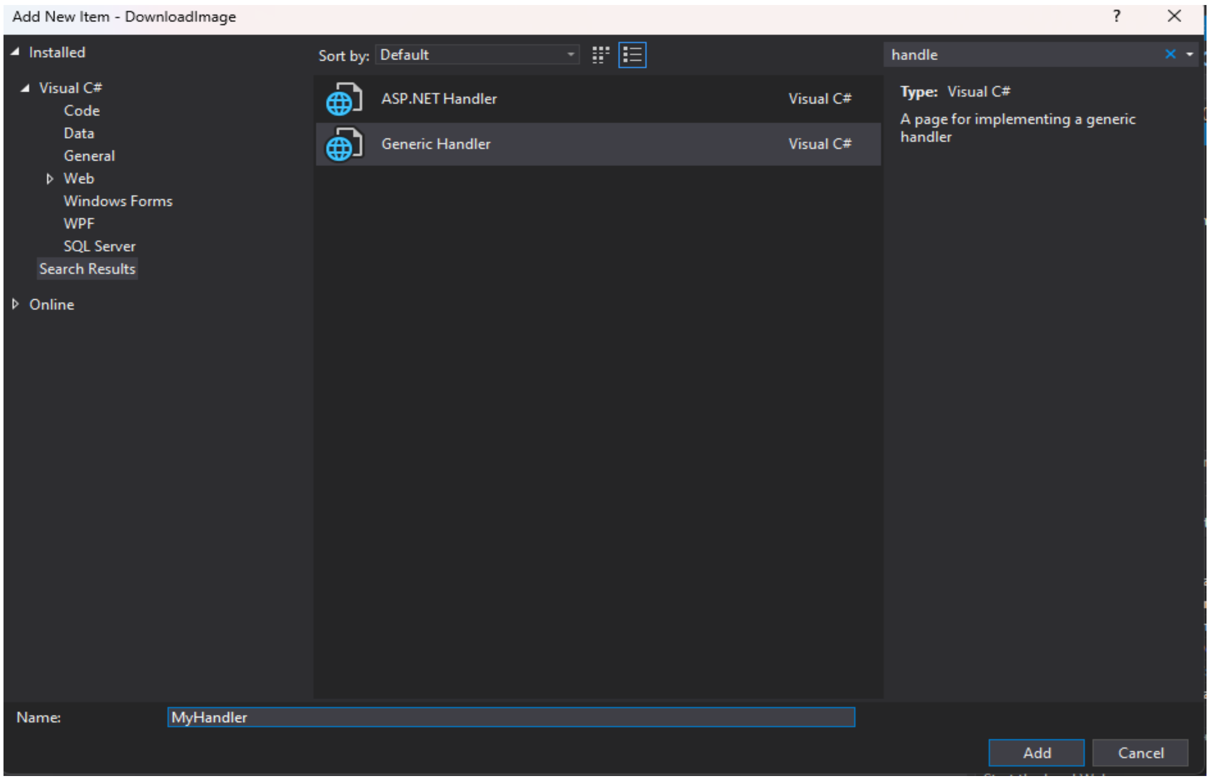
MyHandler.ashx.cs DLImg.aspx.cs DLImg.aspx DownLoadImage.asmx.cs DownLoadImage

DownLoadImage DownLoadImage.DownLoadImage HelloWorld()

```
[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]
2 references
public class DownLoadImage : System.Web.Services.WebService
{

    [WebMethod]
    0 references
    public string HelloWorld()
    {
        return "Hello World";
    }
    [WebMethod, Description("Get Image Content")]

    1 reference
    public byte[] GetImgFile(String fileName)
    {
        if (System.IO.File.Exists(Server.MapPath("~/Images/") + fileName))
        {
            return System.IO.File.ReadAllBytes(Server.MapPath("~/Images/") + fileName);
        }
        else
        {
            return new byte[]
                {0};
        }
    }
}
```

Step 8: - Right Click on your project name > Go to Add > New Item

Step 9: -Create a Web Handler : Click on Generic Handler (ASMX) > Give Name to the Handler as MyHandler.ashx.cs



Add New Item - DownloadImage

Installed
- Visual C#
  - Code
  - Data
  - General
  - Web
  - Windows Forms
  - WPF
  - SQL Server
  - Search Results
- Online

Sort by: Default

ASP.NET Handler	Visual C#
Generic Handler	Visual C#

handle

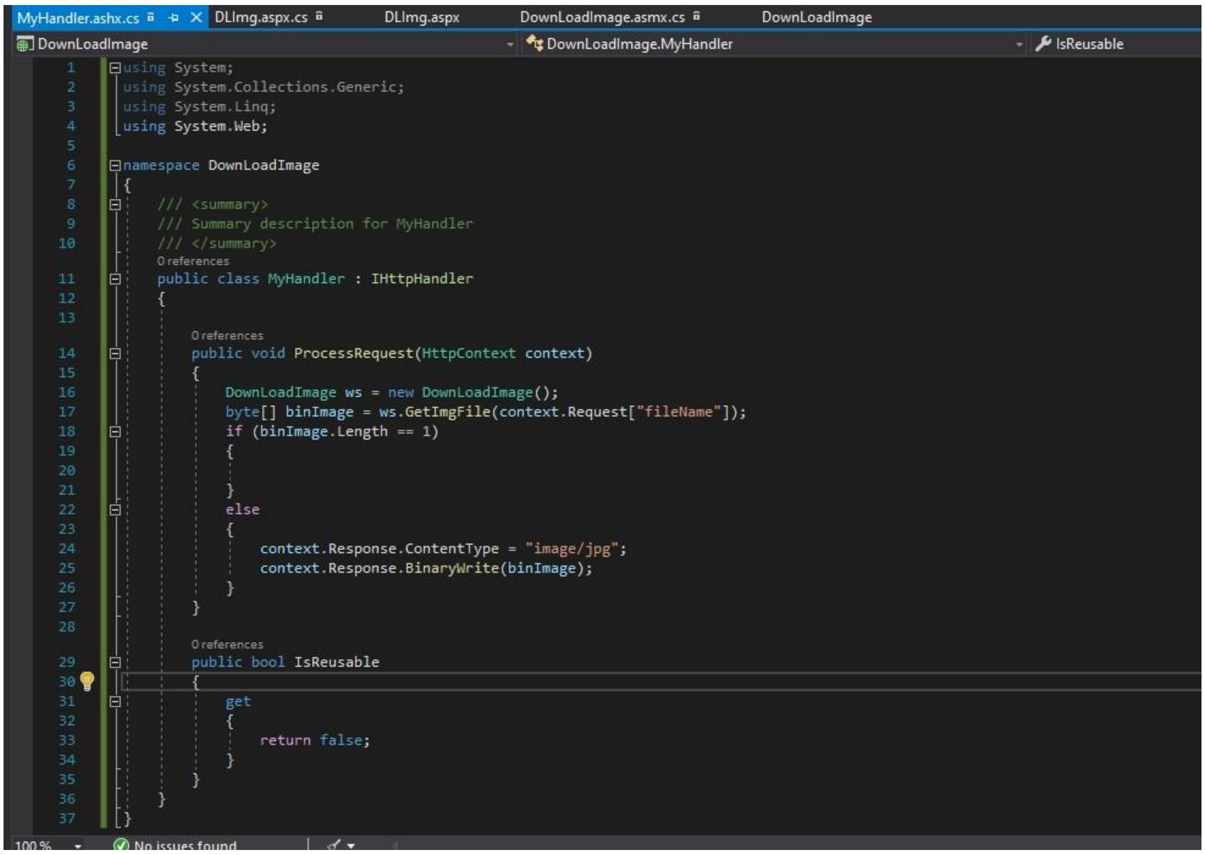
Type: Visual C#

A page for implementing a generic handler

Name: MyHandler

Add Cancel

Step 10: - Write this code in your MyHandler.ashx.cs file



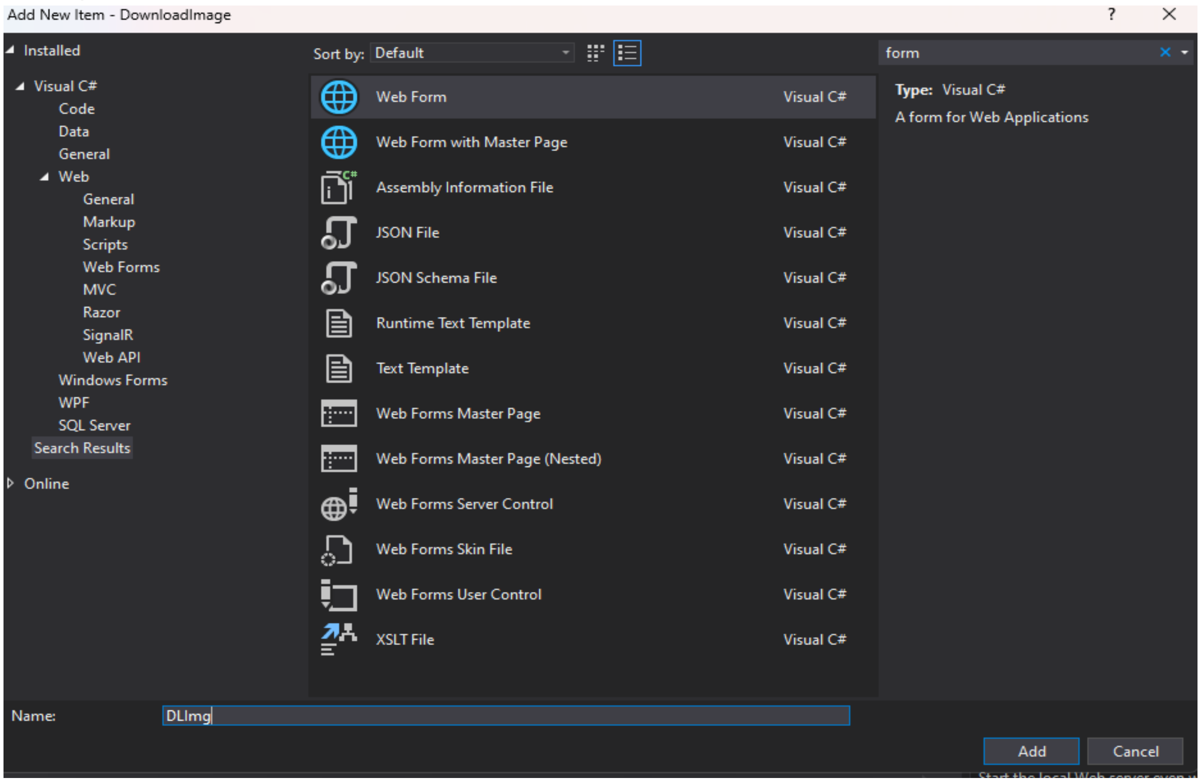
MyHandler.ashx.cs | DLImg.aspx.cs | DLImg.aspx | DownLoadImage.asmx.cs | DownLoadImage

DownLoadImage | DownLoadImage.MyHandler | IsReusable

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

namespace DownLoadImage
{
    /// <summary>
    /// Summary description for MyHandler
    /// </summary>
    public class MyHandler : IHttpHandler
    {

        public void ProcessRequest(HttpContext context)
        {
            DownLoadImage ws = new DownLoadImage();
            byte[] binImage = ws.GetImgFile(context.Request["fileName"]);
            if (binImage.Length == 1)
            {

            }
            else
            {
                context.Response.ContentType = "image/jpg";
                context.Response.BinaryWrite(binImage);
            }
        }

        public bool IsReusable
        {
            get
            {
                return false;
            }
        }
    }
}
```

100 % | No issues found

Step 11: - Right Click on your project name > Go to Add > New Item
Step 12: - Click on Web Form > Give Name to the form as DLImg.aspx.cs



Add New Item - DownloadImage

Installed: Visual C# (Code, Data, General, Web (General, Markup, Scripts, Web Forms, MVC, Razor, SignalR, Web API), Windows Forms, WPF, SQL Server), Search Results; Online

Sort by: Default | Search: form

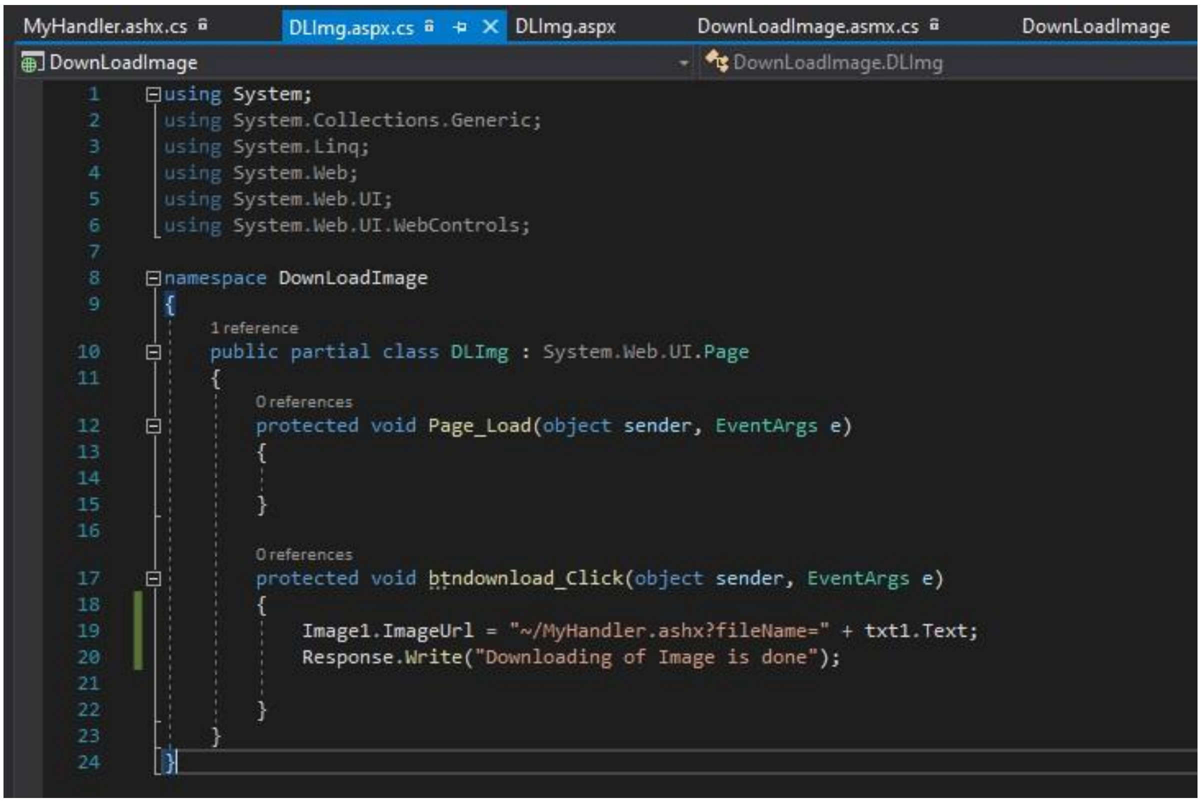
Web Form	Visual C#
Web Form with Master Page	Visual C#
Assembly Information File	Visual C#
JSON File	Visual C#
JSON Schema File	Visual C#
Runtime Text Template	Visual C#
Text Template	Visual C#
Web Forms Master Page	Visual C#
Web Forms Master Page (Nested)	Visual C#
Web Forms Server Control	Visual C#
Web Forms Skin File	Visual C#
Web Forms User Control	Visual C#
XSLT File	Visual C#

Type: Visual C#
A form for Web Applications

Name: DLImg

Add | Cancel

Step 13: -Write this code in DLImg file



MyHandler.ashx.cs | DLImg.aspx.cs | DLImg.aspx | DownLoadImage.asmx.cs | DownLoadImage

DownLoadImage | DownLoadImage.DLImg

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace DownLoadImage
{
    1 reference
    public partial class DLImg : System.Web.UI.Page
    {
        0 references
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        0 references
        protected void btndownload_Click(object sender, EventArgs e)
        {
            Image1.ImageUrl = "~/MyHandler.ashx?fileName=" + txt1.Text;
            Response.Write("Downloading of Image is done");

        }
    }
}
```

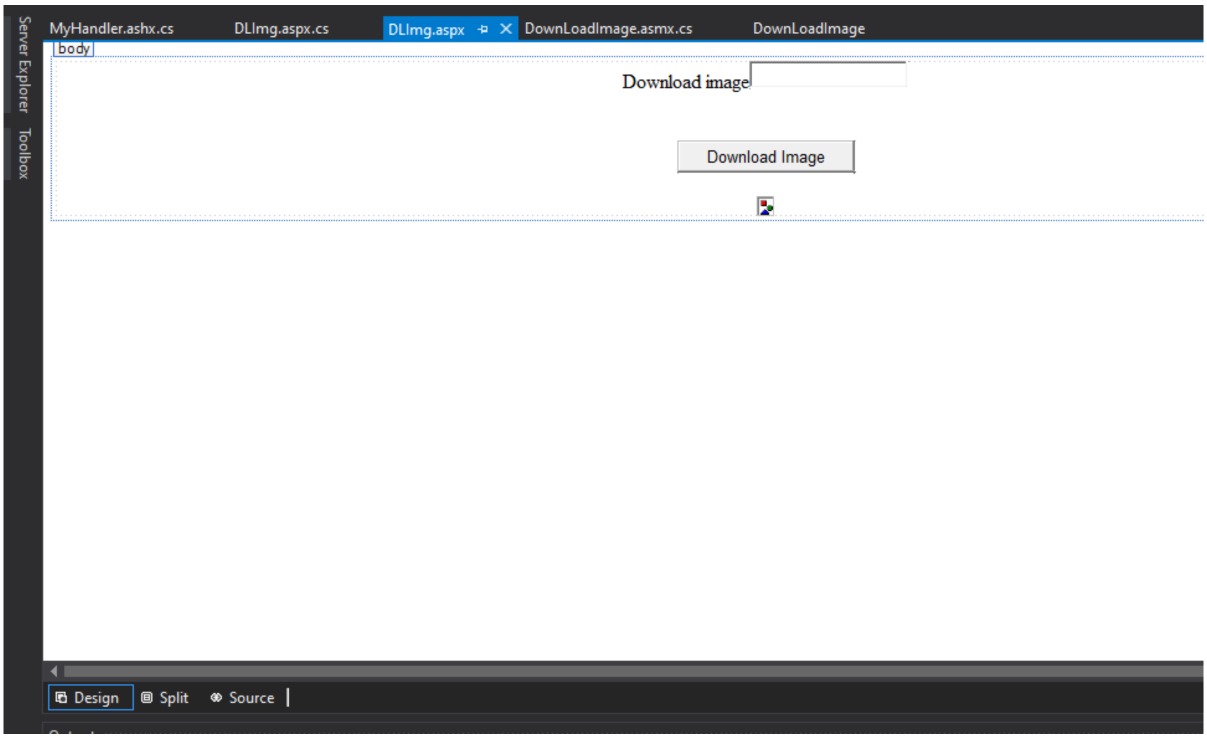
Step 14: -Design the

DLImg.aspx Go to toolbox >

Standard

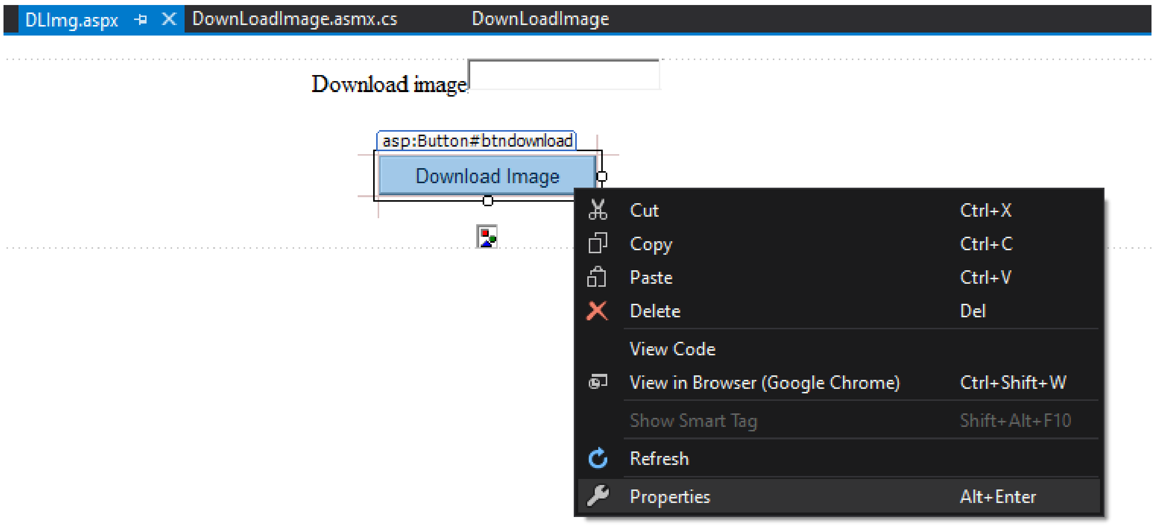
Drag and drop

- Text Field
- Label
- Button
- ImageMap

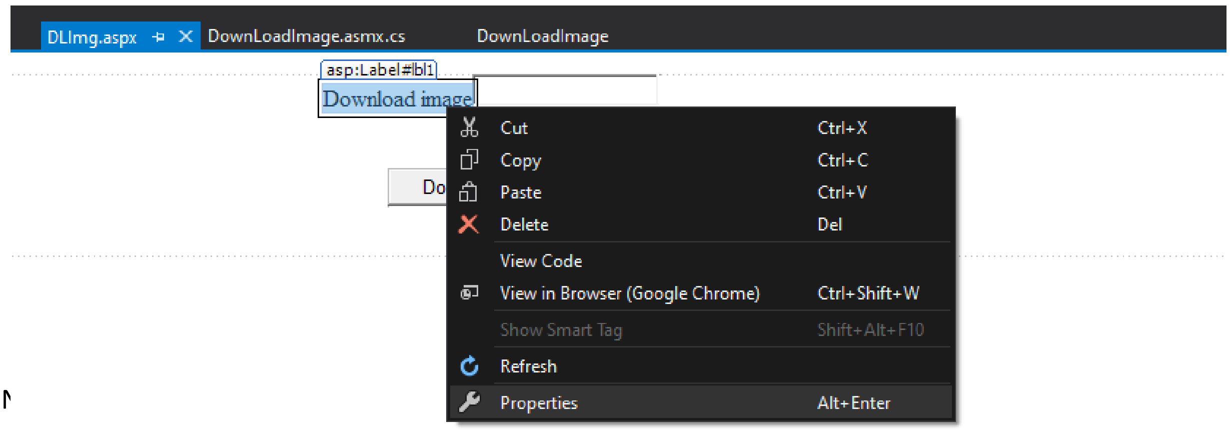


Step 15: -Right click on the button > Properties > Give Button ID – btndownload

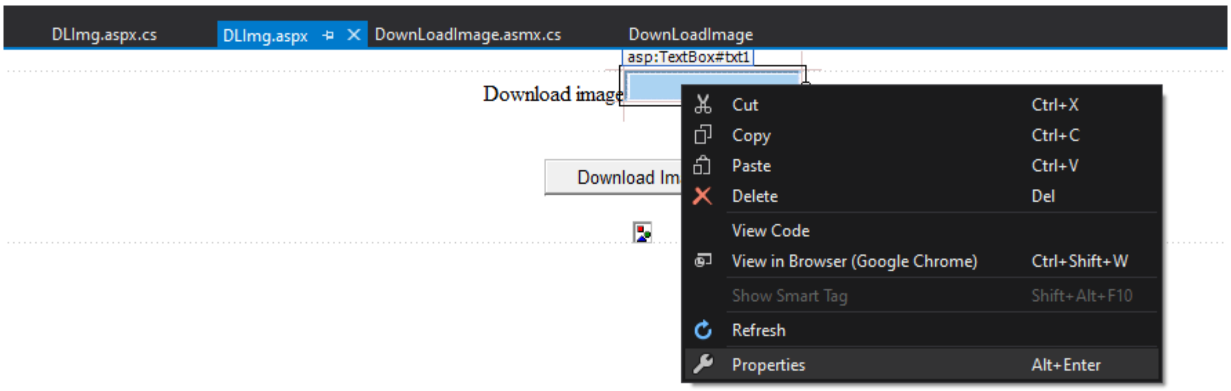
and text – Download Image



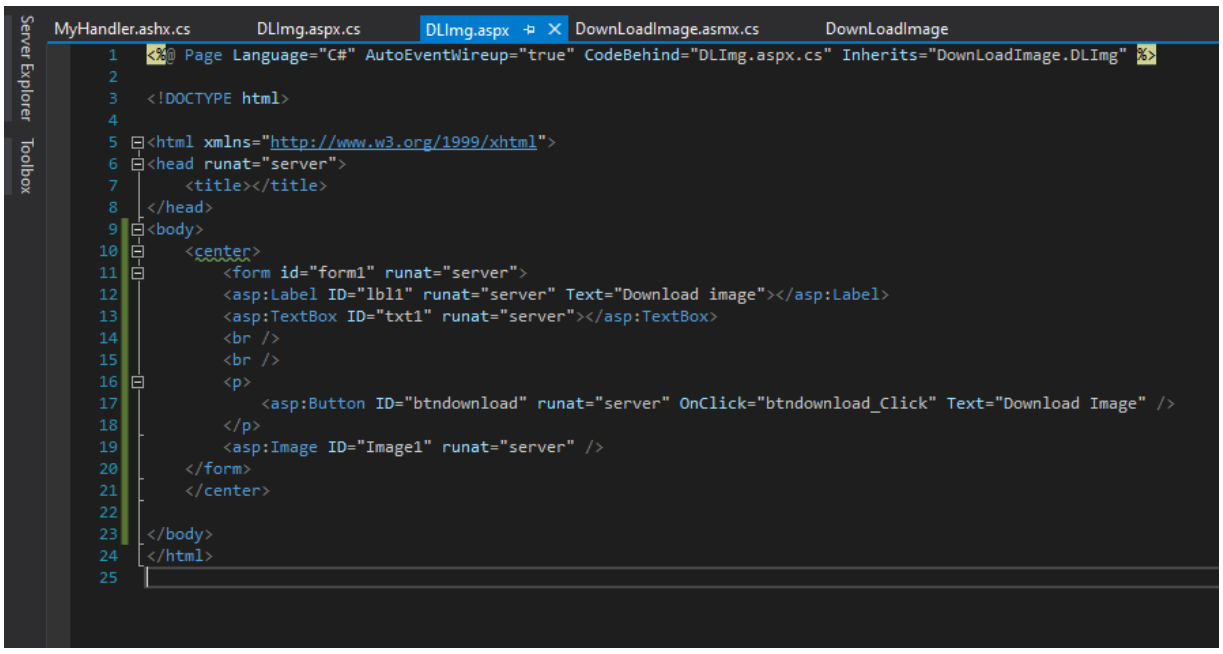
Step 16: -Right click on the label > Properties > Give label ID – lbl1 and Text – Download Image



Step 17: -Right click on the Text Field > Properties > Give ID – txt1



Step 18: -DLImg.aspx source code

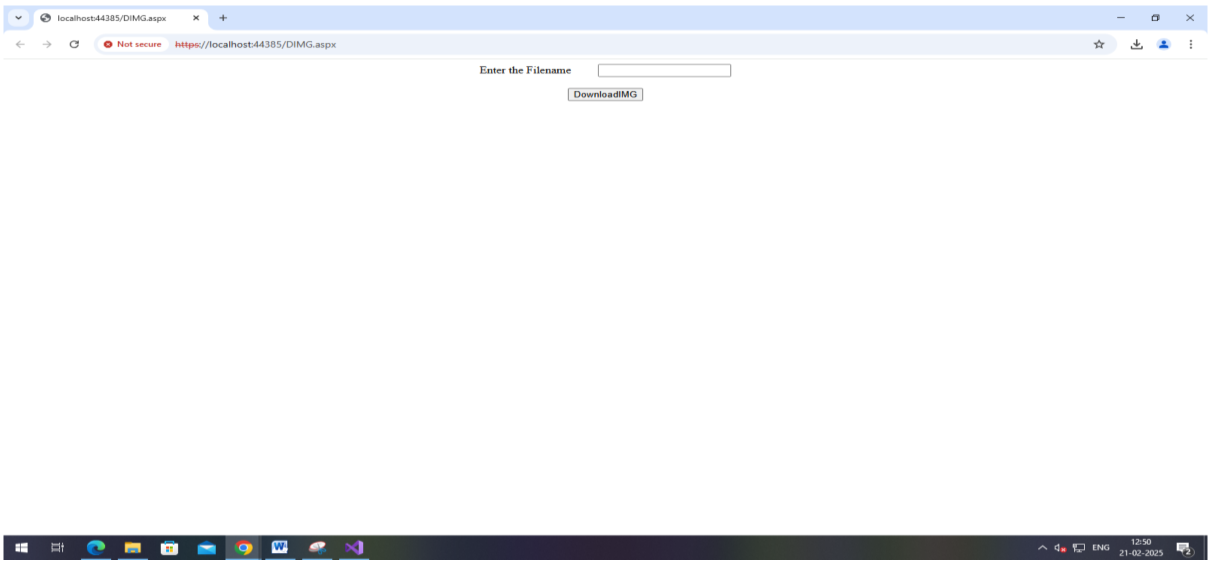


```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="DLImg.aspx.cs" Inherits="DownLoadImage.DLImg" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <center>
        <form id="form1" runat="server">
        <asp:Label ID="lbl1" runat="server" Text="Download image"></asp:Label>
        <asp:TextBox ID="txt1" runat="server"></asp:TextBox>
        <br />
        <br />
        <p>
            <asp:Button ID="btndownload" runat="server" OnClick="btndownload_Click" Text="Download Image" />
        </p>
        <asp:Image ID="Image1" runat="server" />
    </form>
    </center>

</body>
</html>
```

Step 19: - Create an Images folder and copy two jpg images in it.

Step 20: -Run your Project.

Enter your image name.



localhost:44385/DIMG.aspx

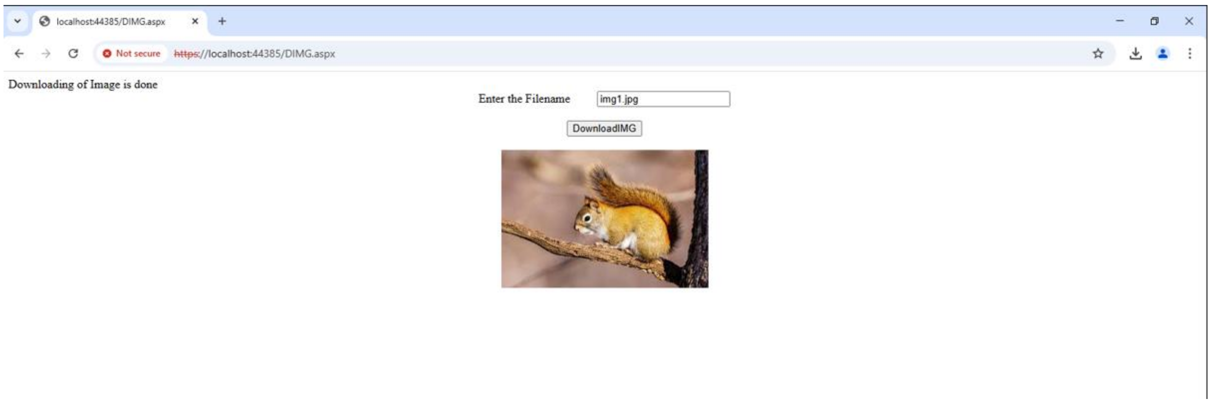
Not secure https://localhost:44385/DIMG.aspx

Enter the Filename

DownloadIMG

Output:-

img1.jpg



localhost:44385/DIMG.aspx

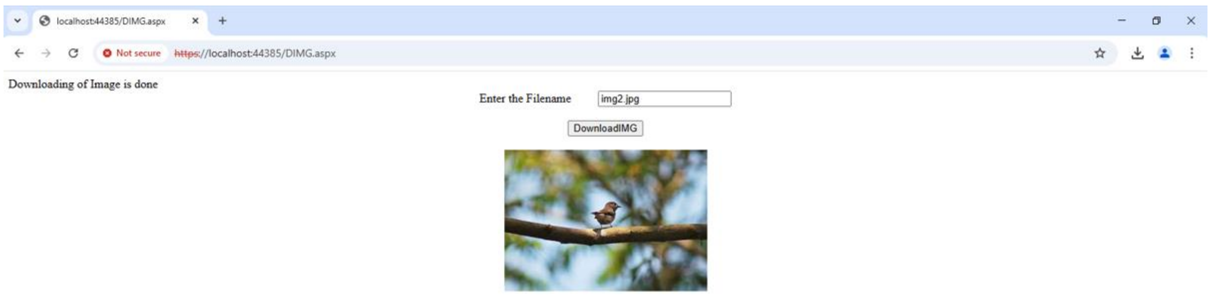
Not secure https://localhost:44385/DIMG.aspx

Downloading of Image is done

Enter the Filename img1.jpg

DownloadIMG

img2.jpg



localhost:44385/DIMG.aspx

Not secure https://localhost:44385/DIMG.aspx

Downloading of Image is done

Enter the Filename img2.jpg

DownloadIMG