Game Programming

Pre-Requisite

Install Microsoft vs 2010 professional and select visual c# development settings

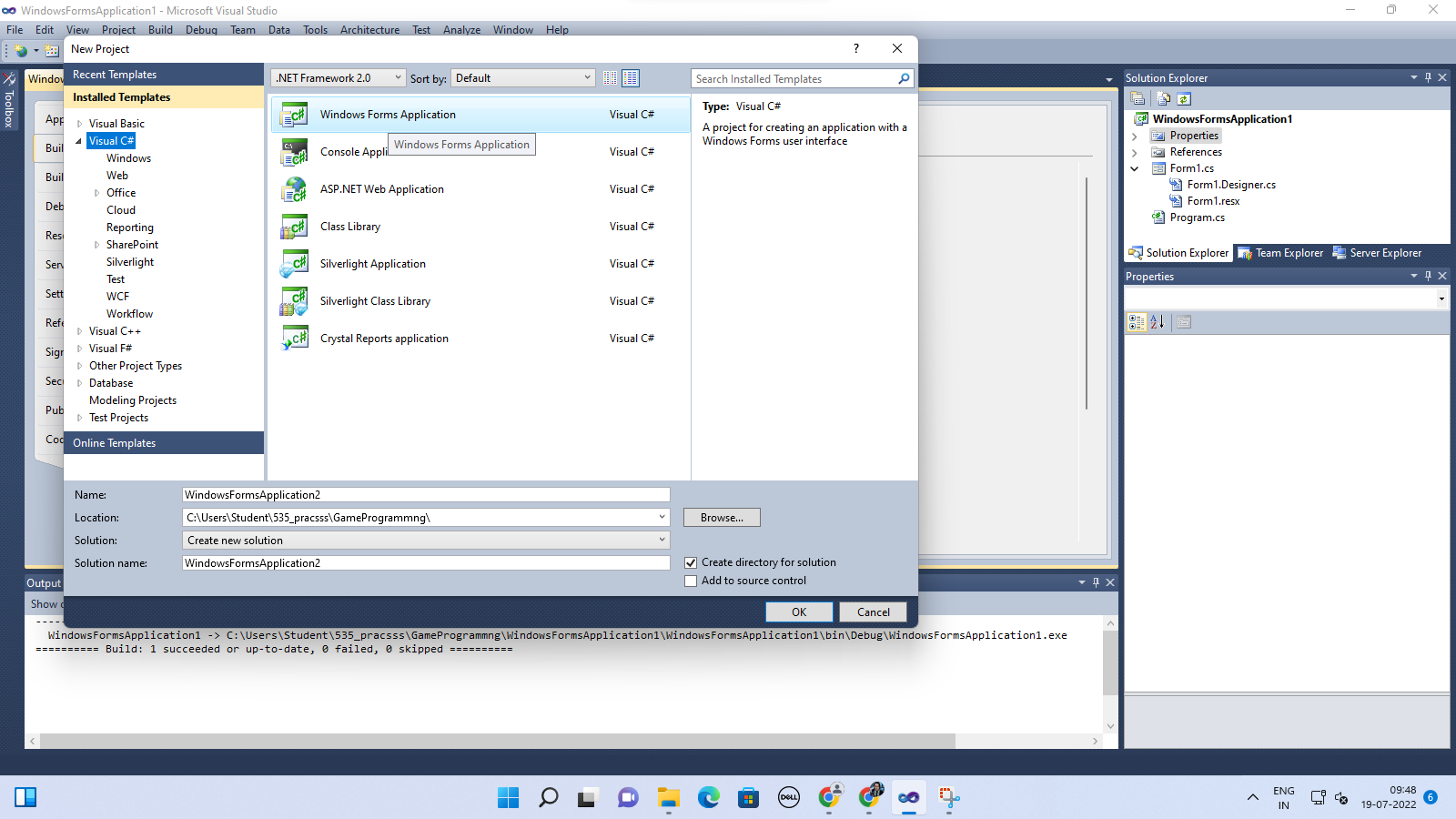
Practical1

Aim: Setup DirectX 11, Windows Framework and Initialize Direct3D Device.

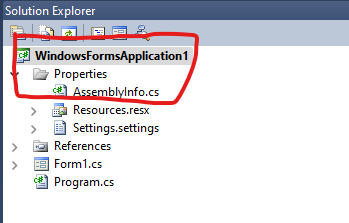
In this practical we are just learning the window framework and initializing a Direct3D device.

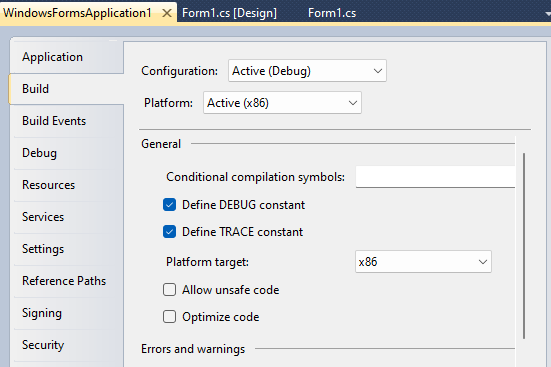
Step1:

1. Create new project, and select “Windows Forms Application “, and select .NET Framework as 2.0 in Visual C#.



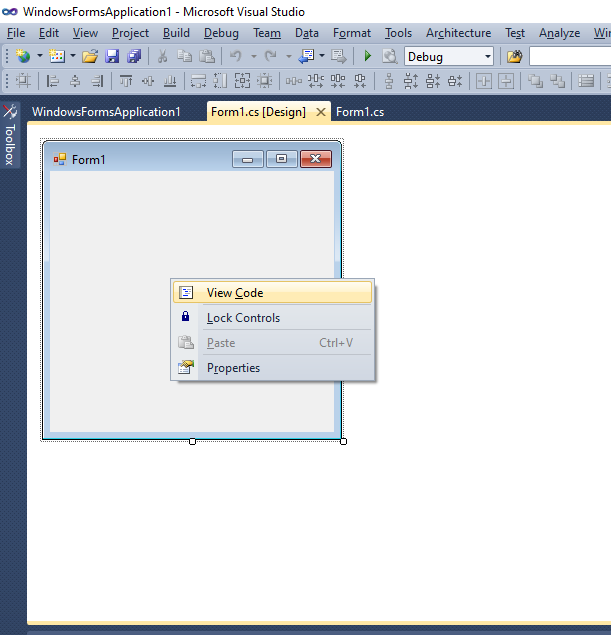
1. Right Click on Properties in Solution Explorer. Click on build Select Platform Target and Select x86.

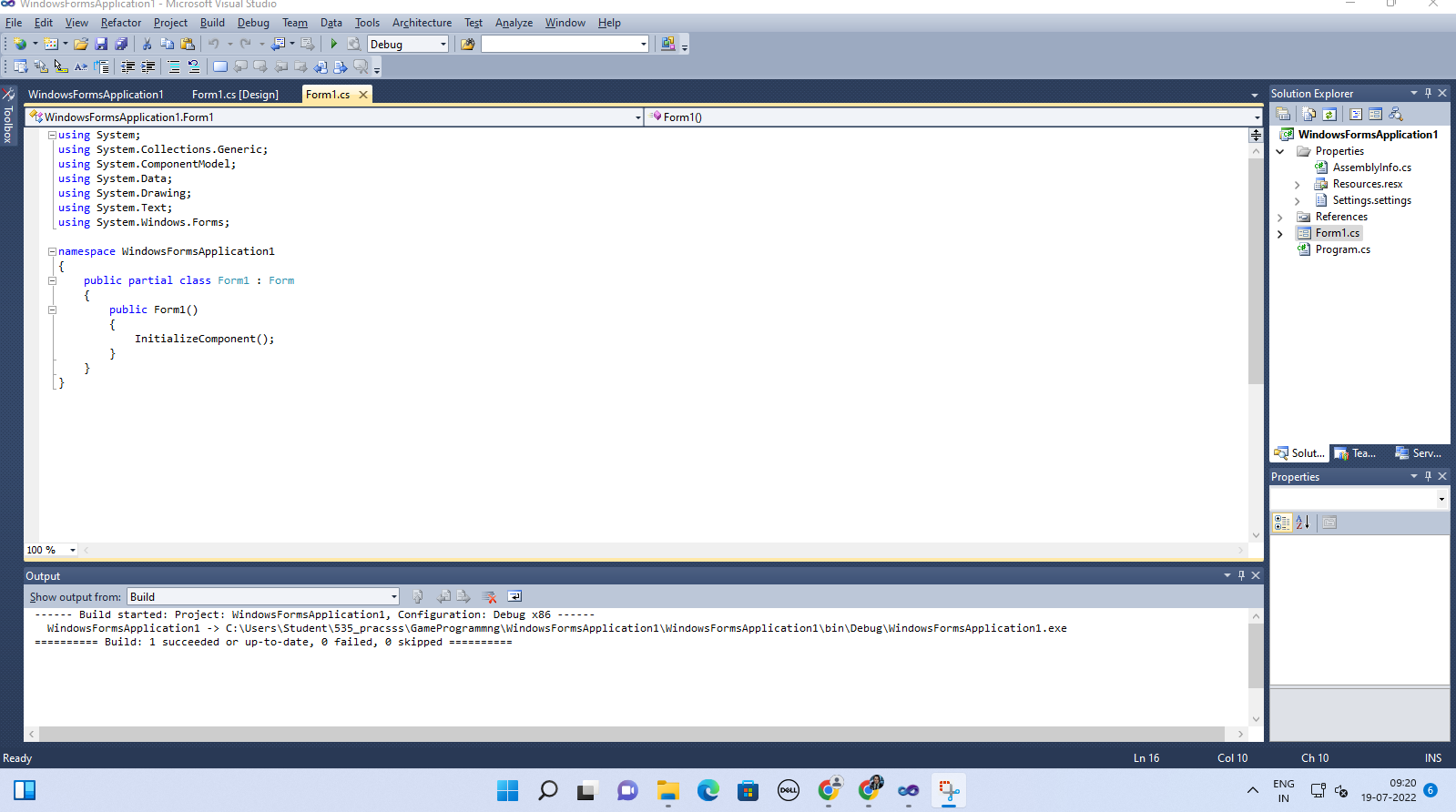




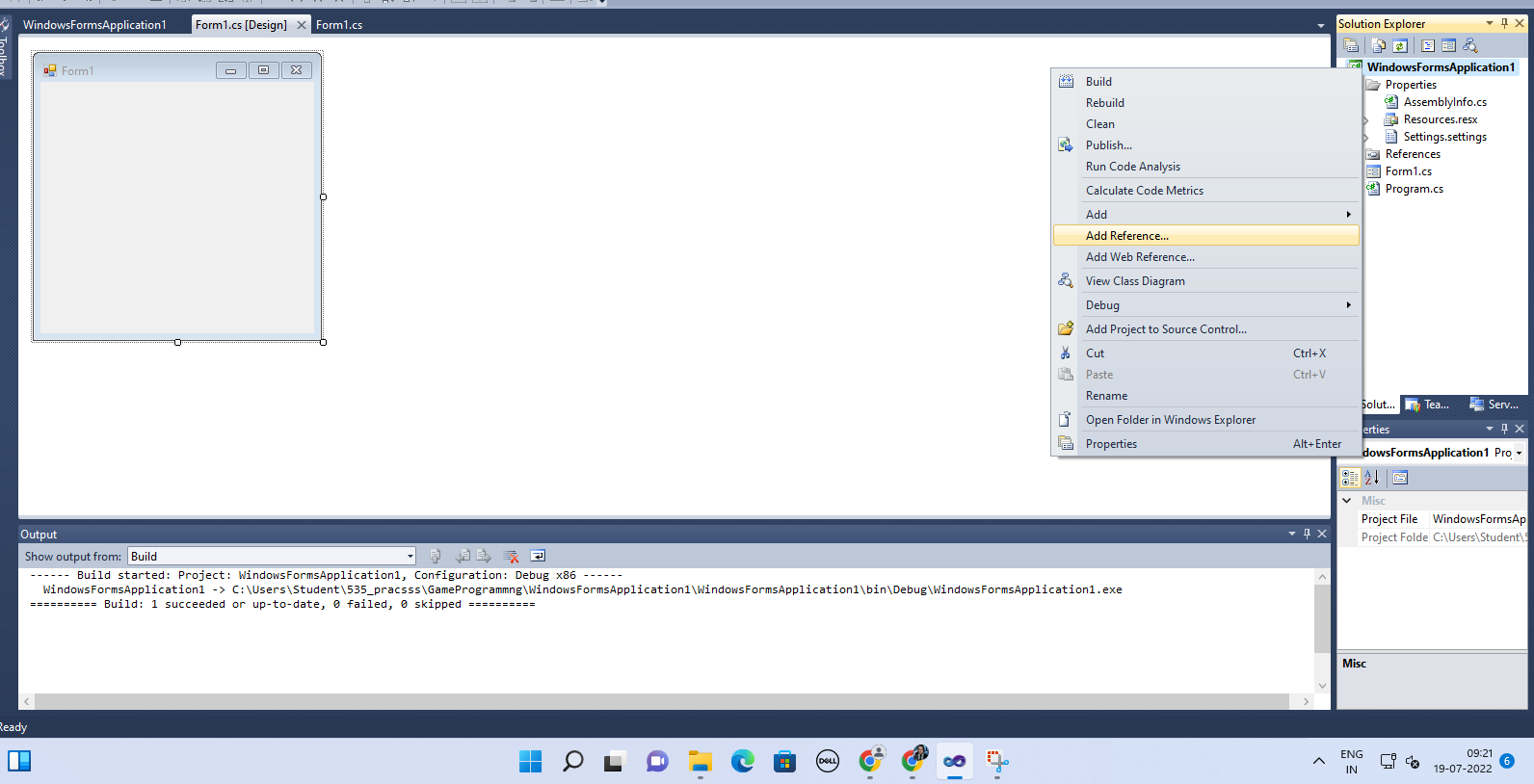
Step2: Right Click on the Form.

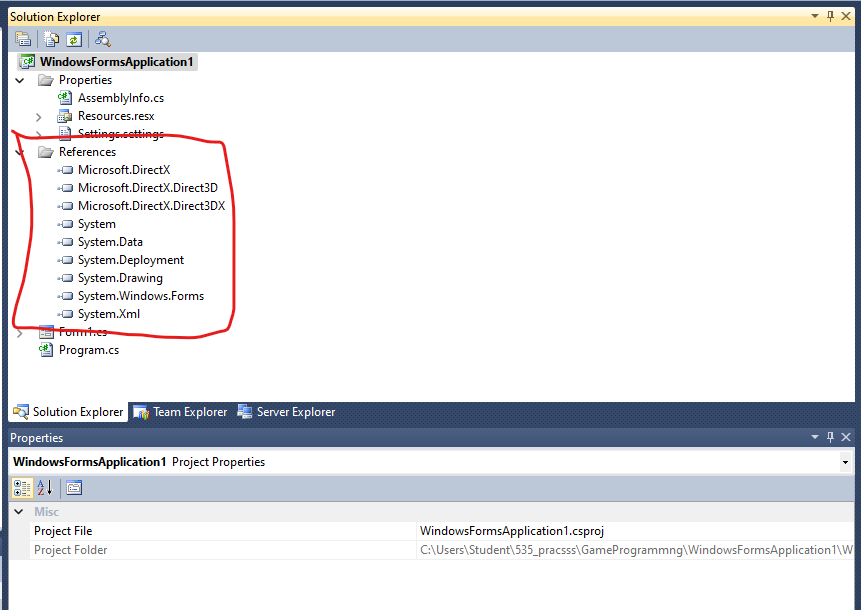
Click on View Code of Form 1.



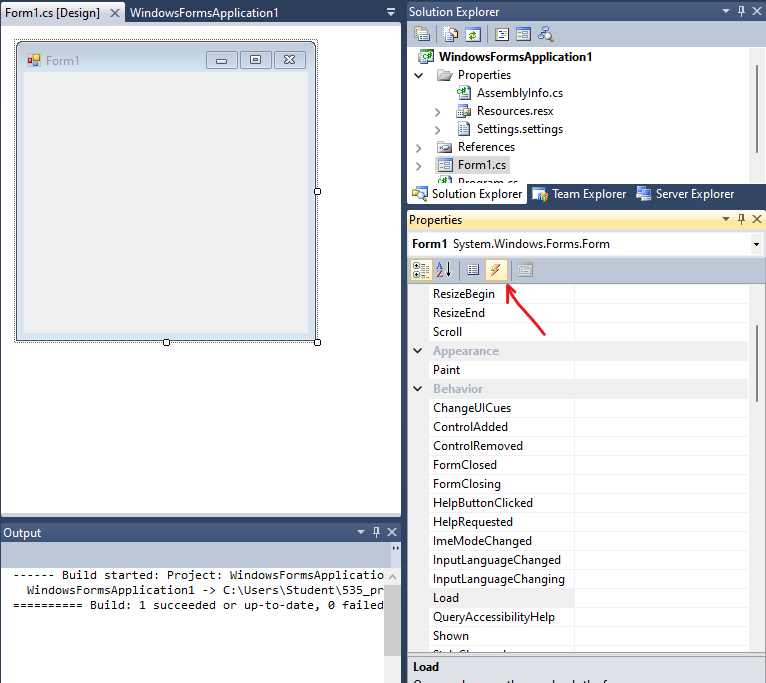


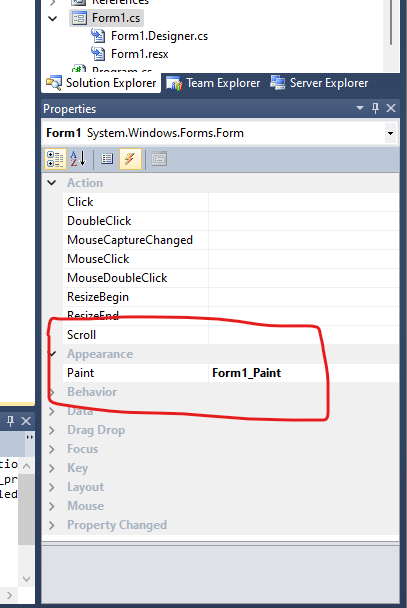
Step3: Go to Solution Explorer, right click on project name, and select Add Reference. Click on Browse and select the given .dll files which are “Microsoft.DircectX”,  “Microsoft.DirectX.Direct3D”, and “Microsfit.Directx.Direct3D.DirectX3DX”.





Step 4: Go to properties Section of Form (Design), double click Paint in the Event List (THUNDER SYMBOL) and enter as Form1\_Paint.





Step 5:  Edit the Form’s C# code file. Namespace must be as same as your project name.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;

using Microsoft.DirectX;

using Microsoft.DirectX.Direct3D;

using Microsoft.DirectX.Direct3DX;

namespace WindowsFormsApplication1

{

    public partial class Form1 : Form

    {

        Microsoft.DirectX.Direct3D.Device device;

        public Form1()

        {

            InitializeComponent();

            InitDevice();

        }

        public void InitDevice()

        {

            PresentParameters pp = new PresentParameters();

            pp.Windowed = true;

            pp.SwapEffect = SwapEffect.Discard;

            device = new Device(0, DeviceType.Hardware, this, CreateFlags.HardwareVertexProcessing,pp);

        }

        private void Render()

        {

            device.Clear(ClearFlags.Target, Color.Orange, 0, 1);

            device.Present();

        }

        private void Form1\_Paint(object sender, PaintEventArgs e)

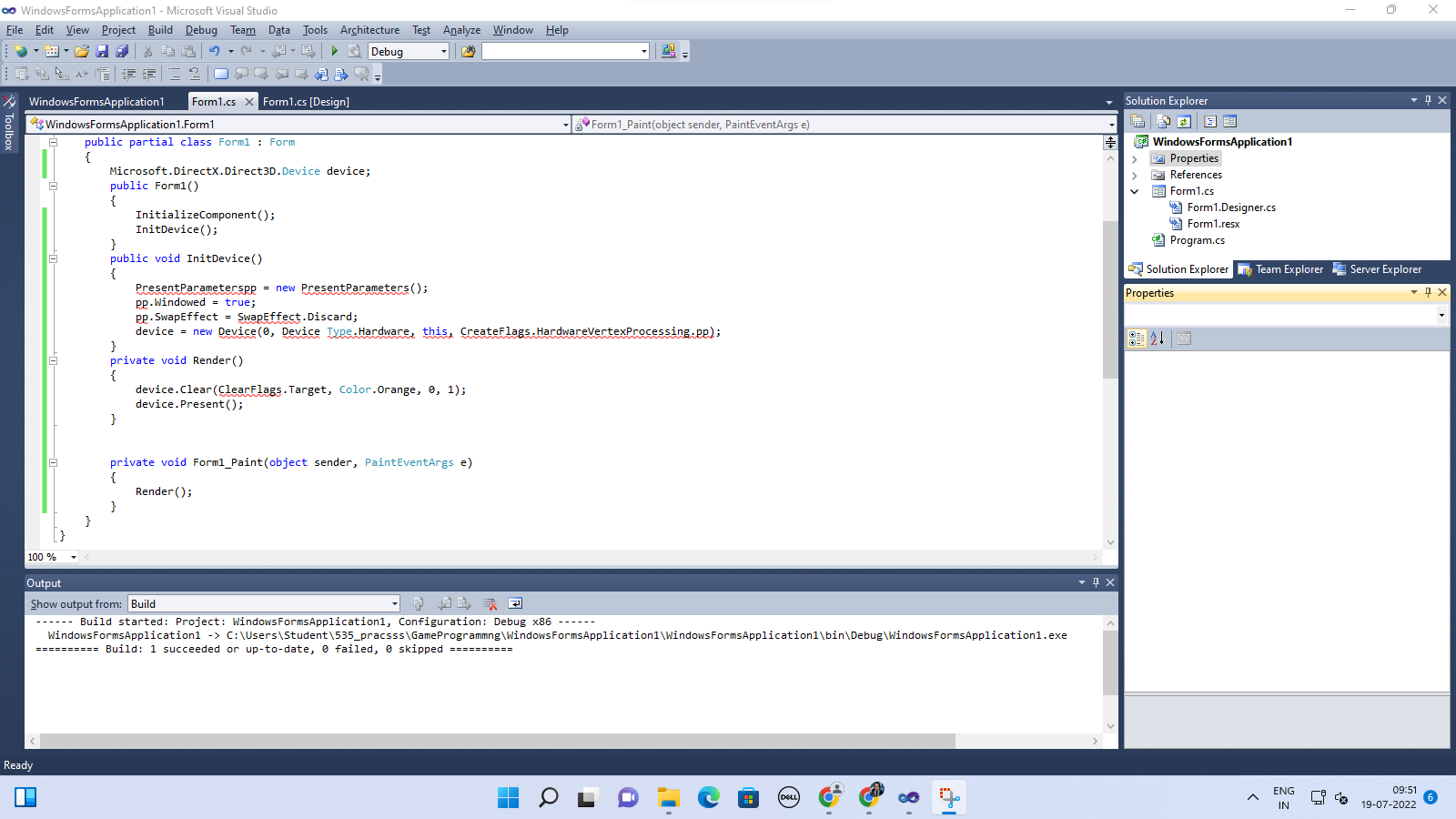
        {

            Render();

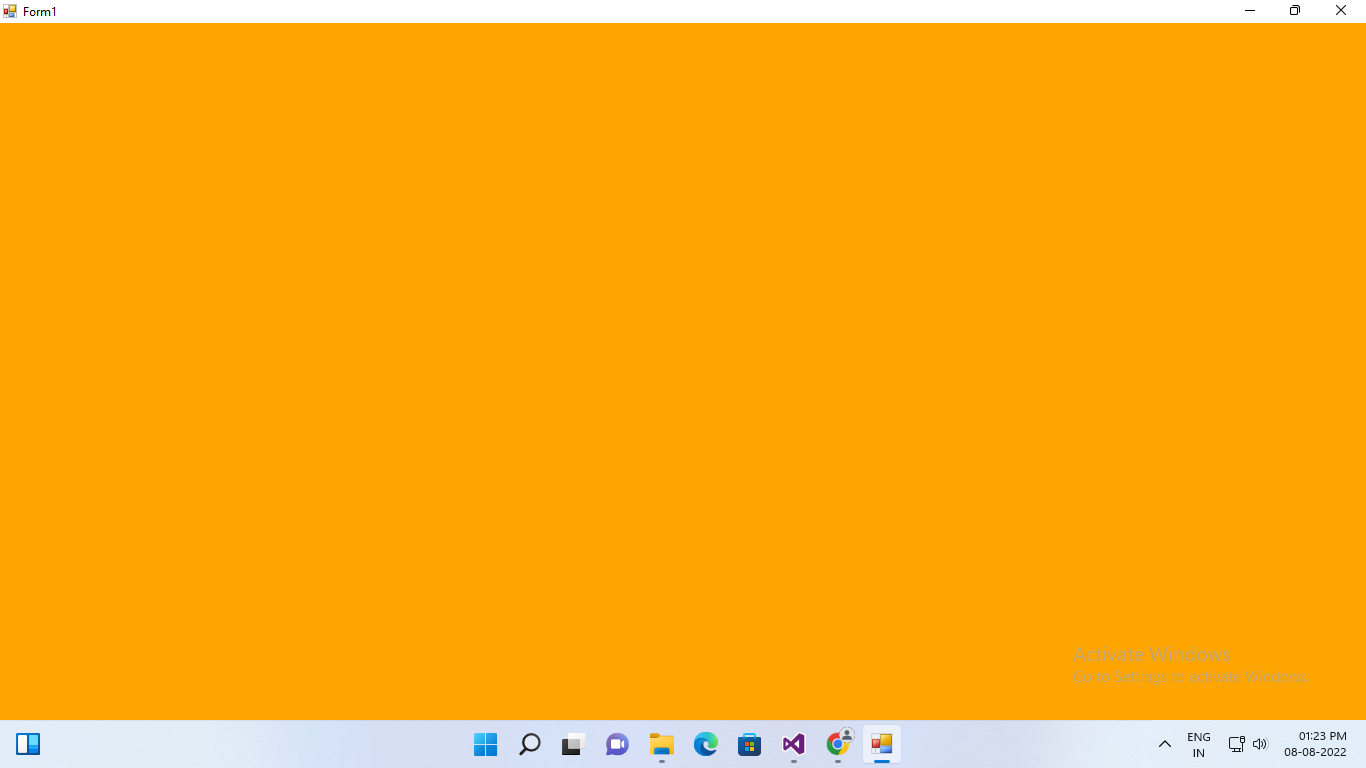
        }

    }

}



Output:



Practical2

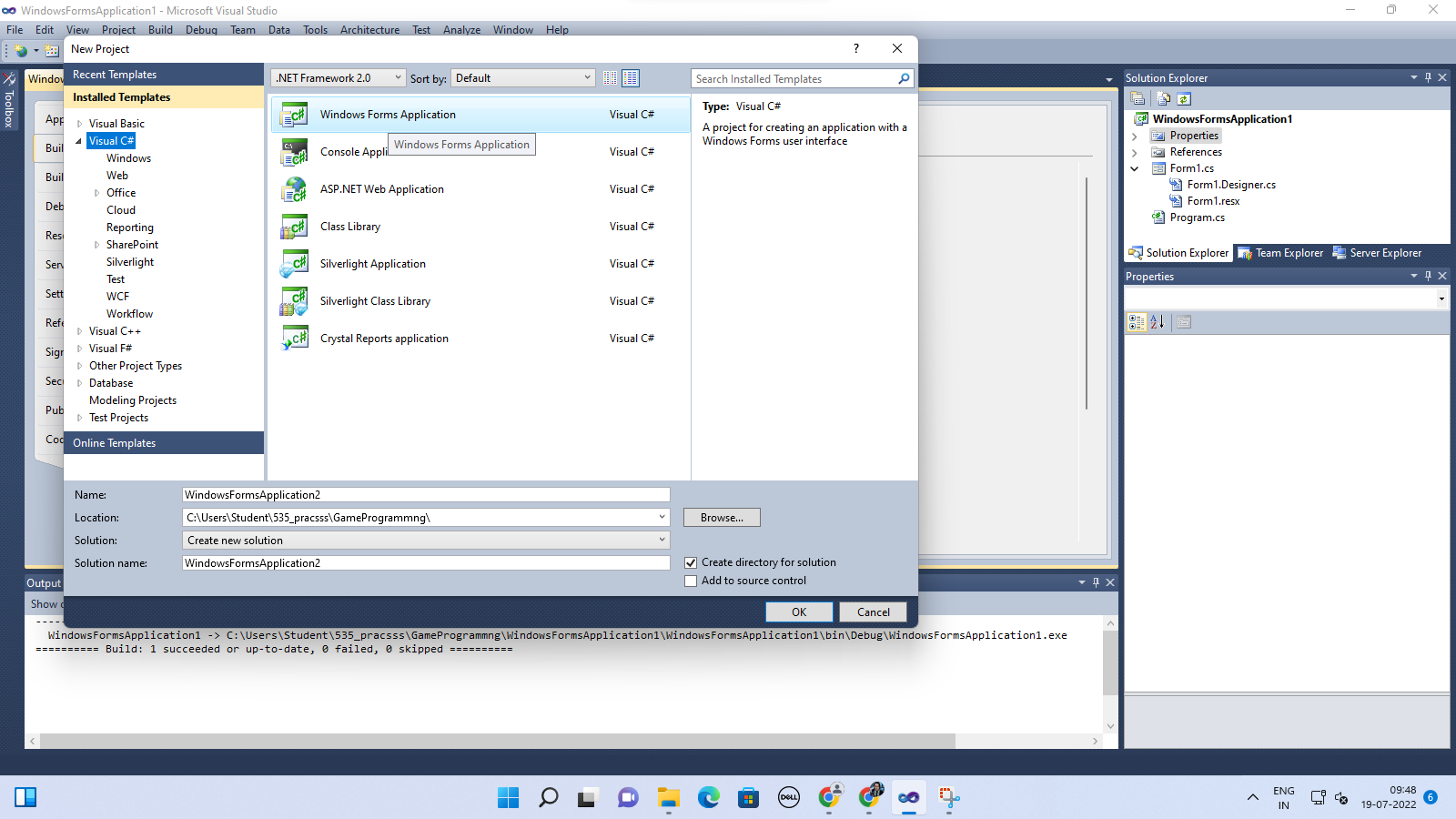
C > windows > microsoft.Net > DirectX for managed code > 1.0.2902.0 > import files

Aim: draw a triangle using Direct 3D 11

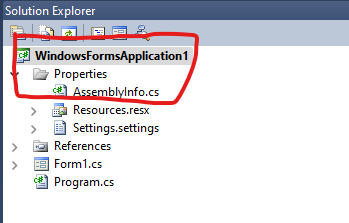
Steps

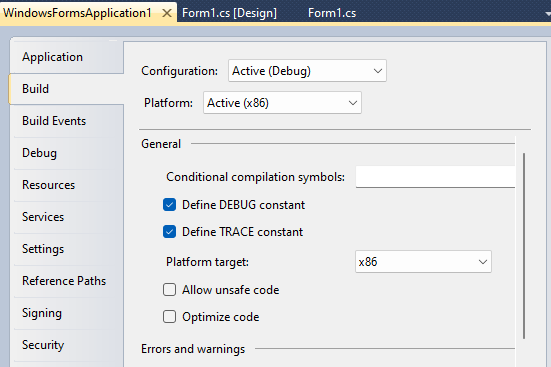
Step1:

i) Create new project and select “windows Form application”, select .NET Framework as 2.0 in Visual C#.



ii) Right Click on properties Click on open click on build Select Platform Target and select x86.

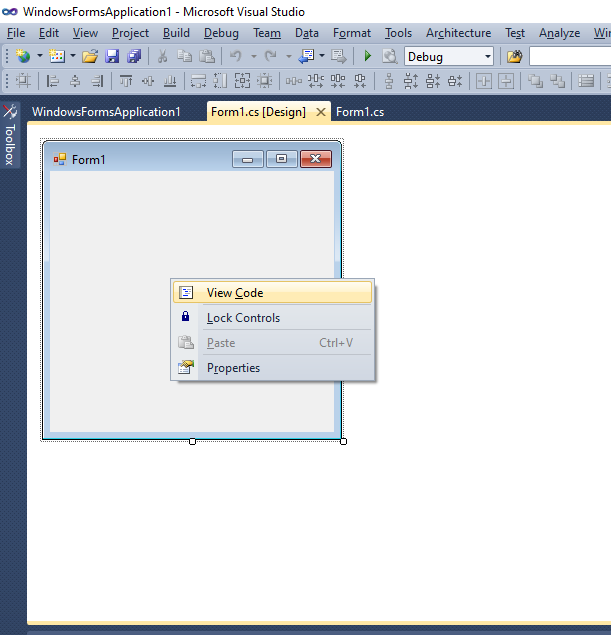


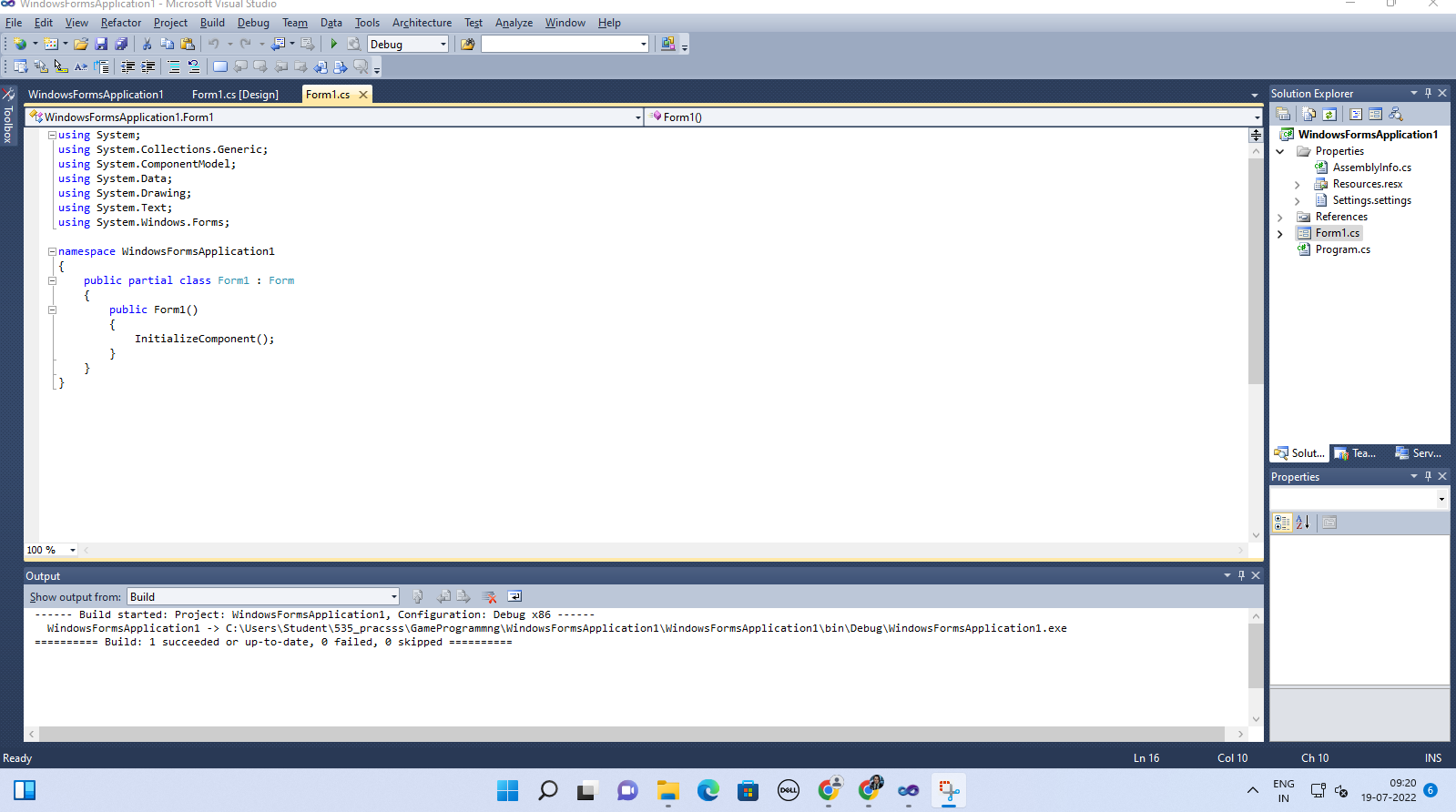


Step2:

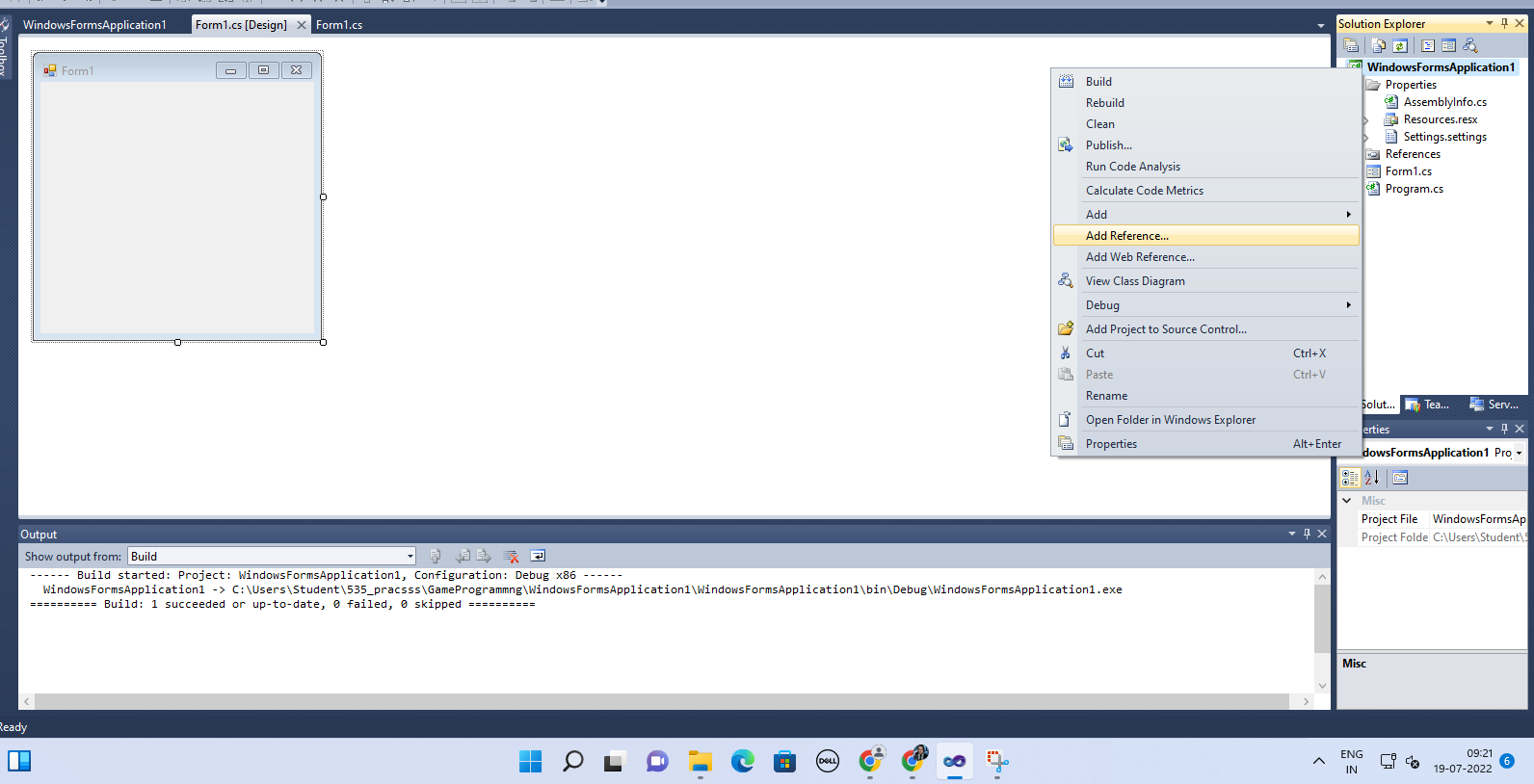
Right Click on the Form.

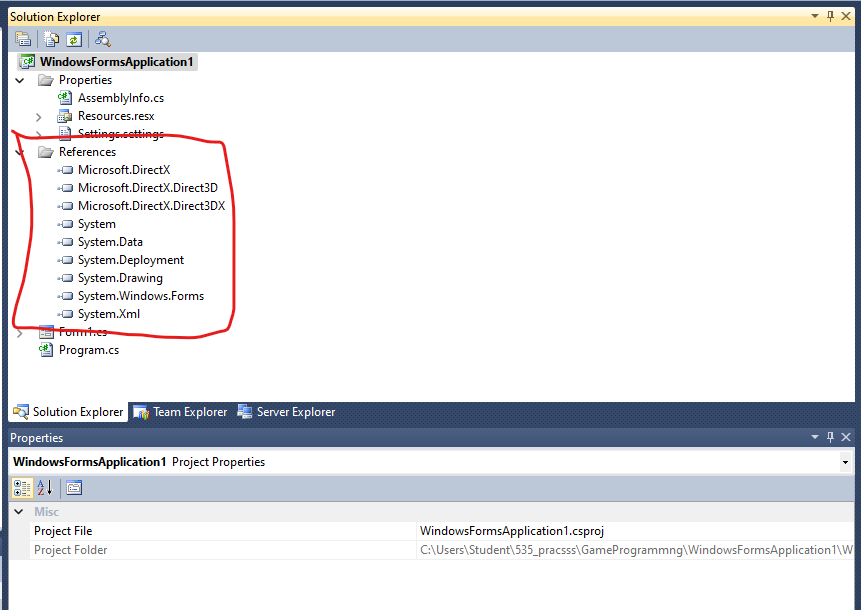
Click on View Code of Form 1.



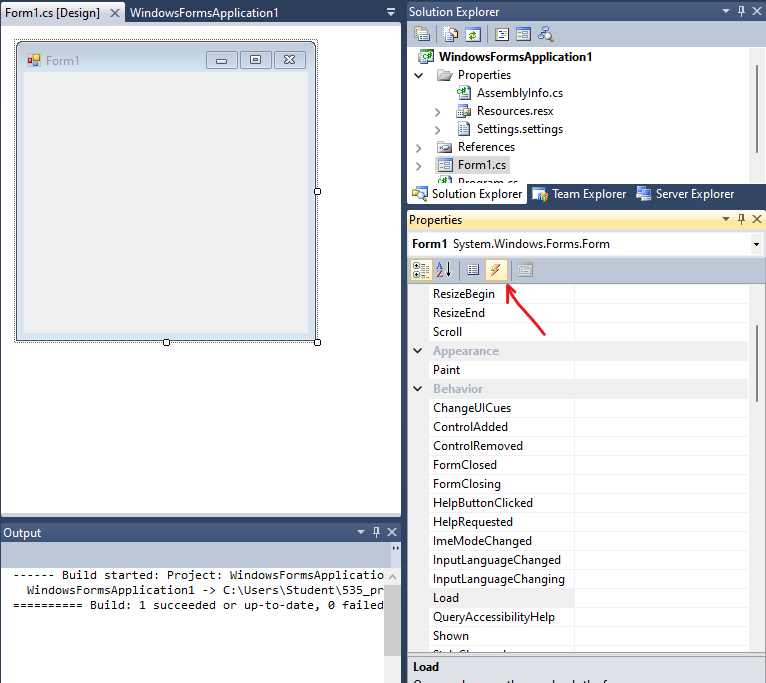


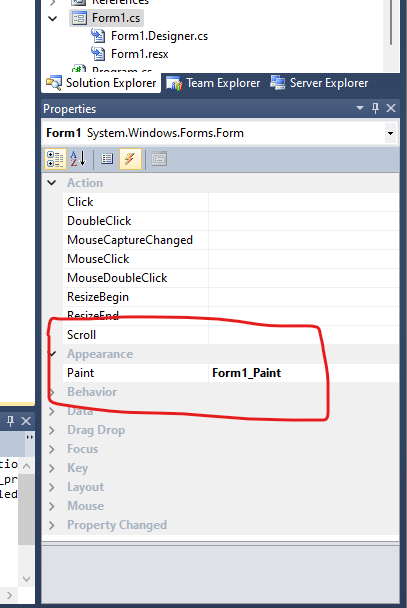
Step3: Go to Solution Explorer, right click on project name, and select Add Reference. Click on Browse and select the given .dll files which are “Microsoft.DircectX”,  “Microsoft.DirectX.Direct3D”, and “Microsfit.Directx.Direct3D.DirectX3DX”.





Step 4: Go to properties Section of Form (Design), double click Paint in the Event List (THUNDER SYMBOL) and enter as Form1\_Paint.





Step 5:  Edit the Form’s C# code file. Namespace must be as same as your project name.

Code:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;

using Microsoft.DirectX;

using Microsoft.DirectX.Direct3D;

namespace WindowsFormsApplication1

{

    public partial class Form1 : Form

    {

        Microsoft.DirectX.Direct3D.Device device;

        public Form1()

        {

            InitializeComponent();

            InitDevice();

        }

        public void InitDevice()

        {

            PresentParameters pp = new PresentParameters();

            pp.Windowed = true;

            pp.SwapEffect = SwapEffect.Discard;

            device = new Device(0, DeviceType.Hardware, this, CreateFlags.HardwareVertexProcessing, pp);

        }

        private void Render()

        {

          CustomVertex.TransformedColored[] vertexes = new

CustomVertex.TransformedColored[3];

vertexes[0].Position = new Vector4(240, 110, 0, 1.0f);//first point

vertexes[0].Color = System.Drawing.Color.FromArgb(0, 0, 255).ToArgb();

vertexes[1].Position = new Vector4(380, 420, 0, 1.0f);//second point

vertexes[1].Color = System.Drawing.Color.FromArgb(0, 255, 0).ToArgb();

vertexes[2].Position = new Vector4(110, 420, 0, 1.0f);//third point

vertexes[2].Color = System.Drawing.Color.FromArgb(255, 0, 0).ToArgb();

device.Clear(ClearFlags.Target, Color.CornflowerBlue, 1.0f, 0);

device.BeginScene();

device.VertexFormat = CustomVertex.TransformedColored.Format; // connect screen with object

device.DrawUserPrimitives(PrimitiveType.TriangleList, 1, vertexes); //to draw the object on screen

device.EndScene();

device.Present();

        }

        private void Form1\_Paint(object sender, PaintEventArgs e)

        {

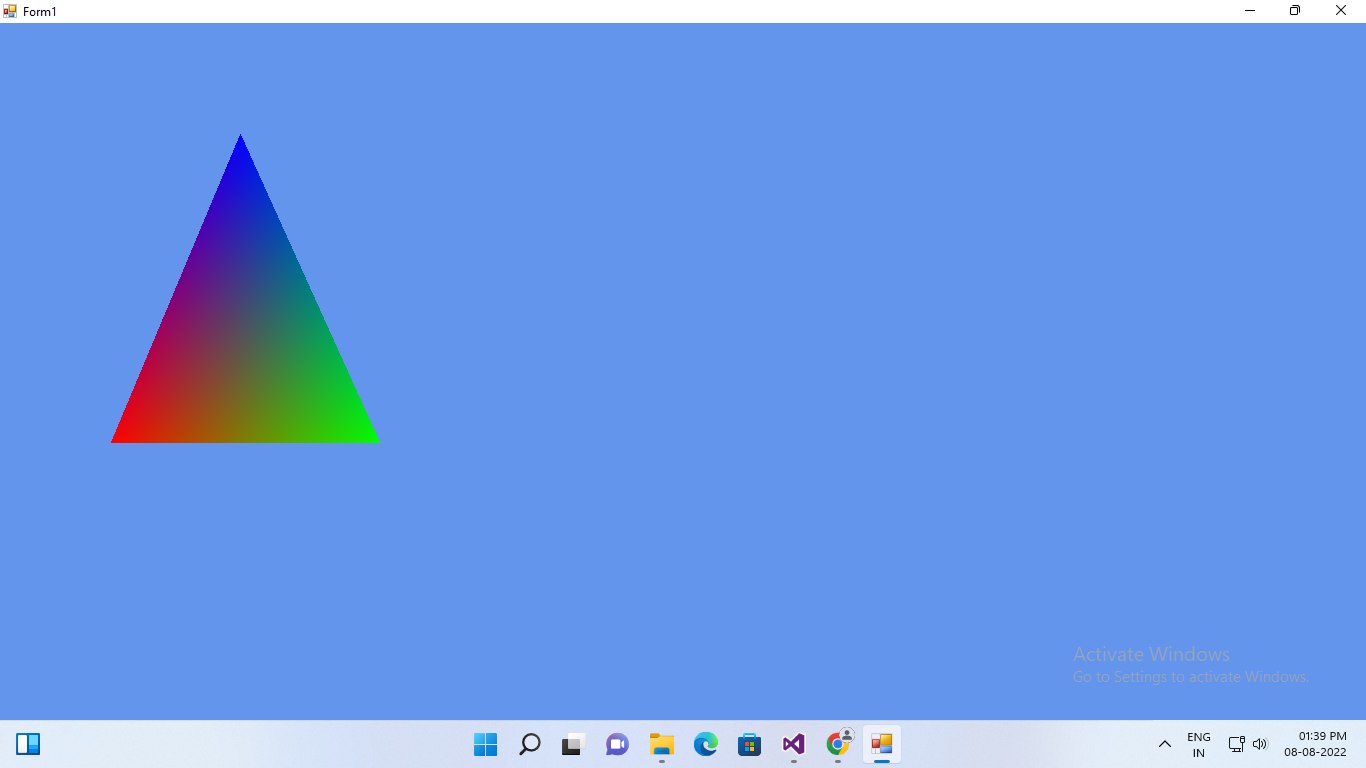
            Render();

        }

    }

}

OUTPUT:



Practical3

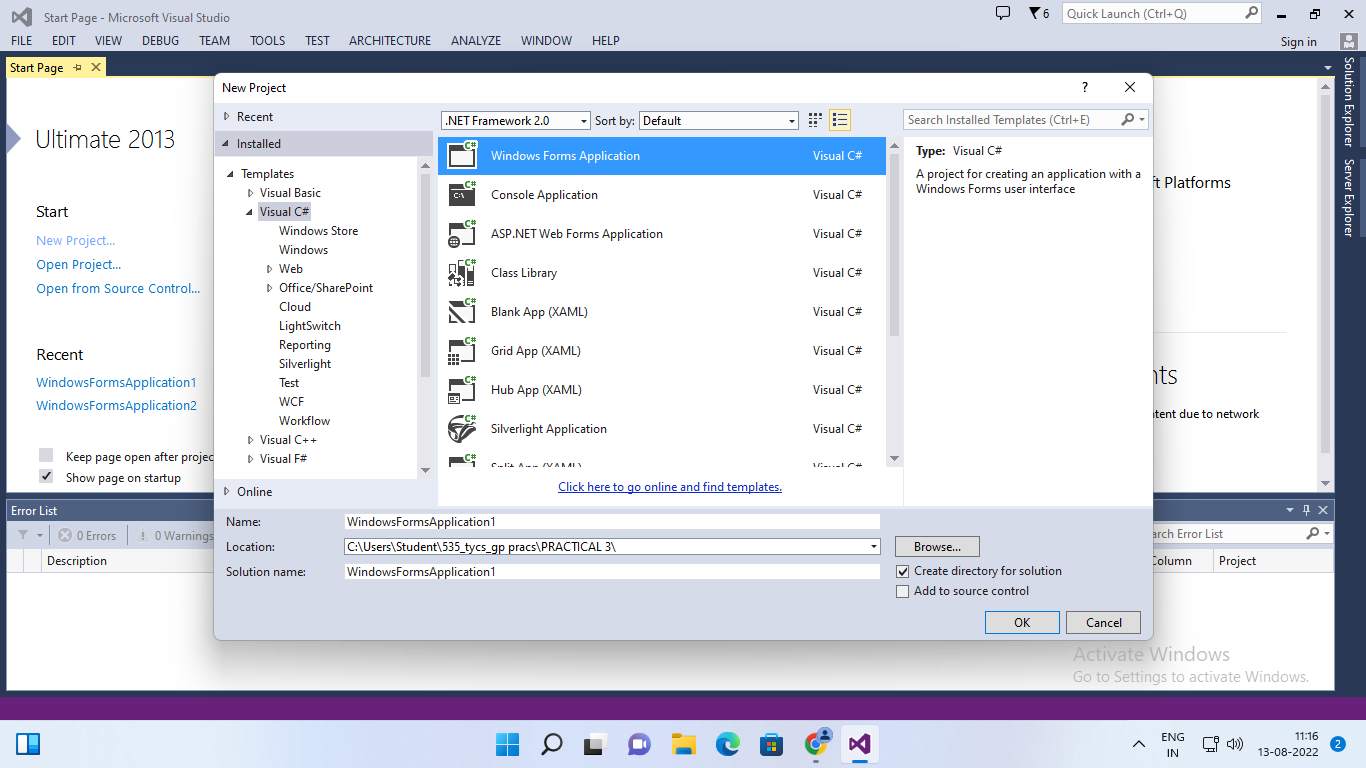
C > windows > microsoft.Net > DirectX for managed code > 1.0.2902.0 > import files

Aim: Texture the triangle using Direct 3D 11

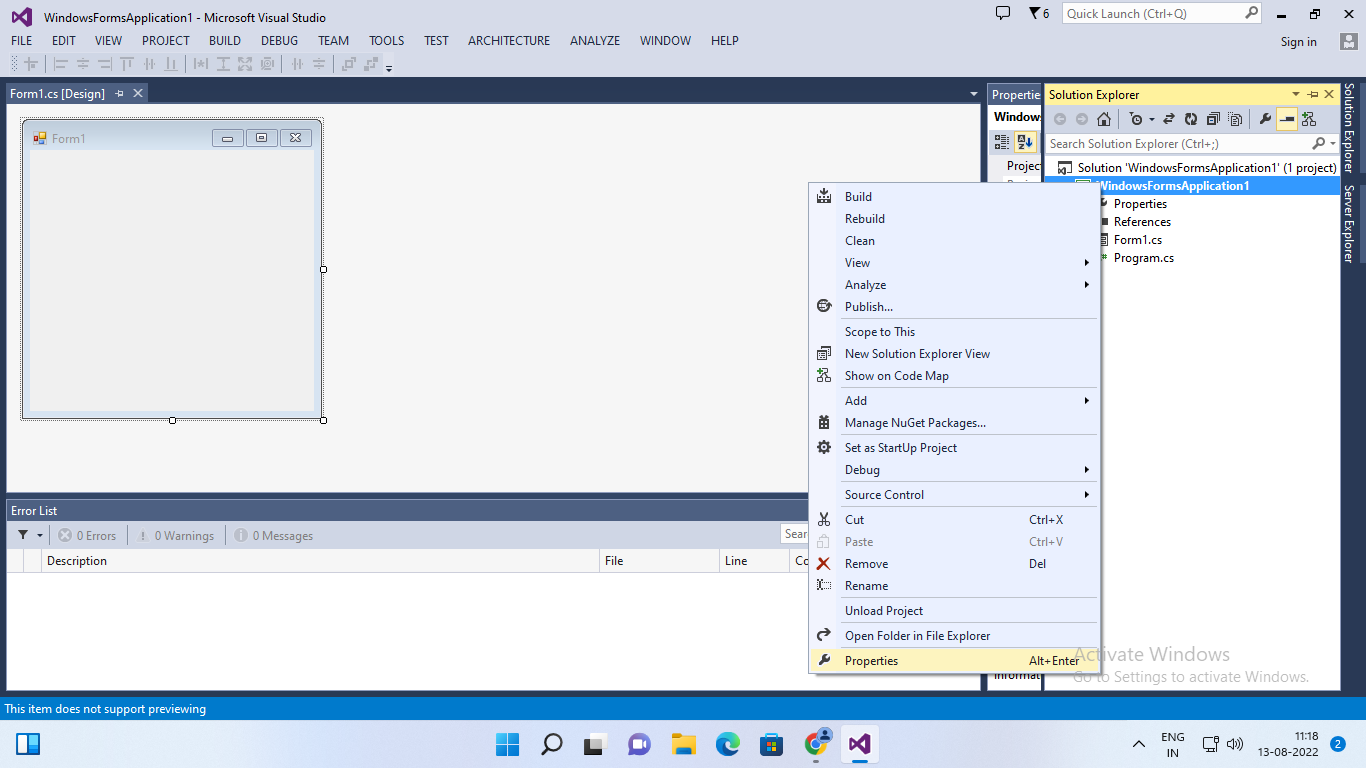
Steps

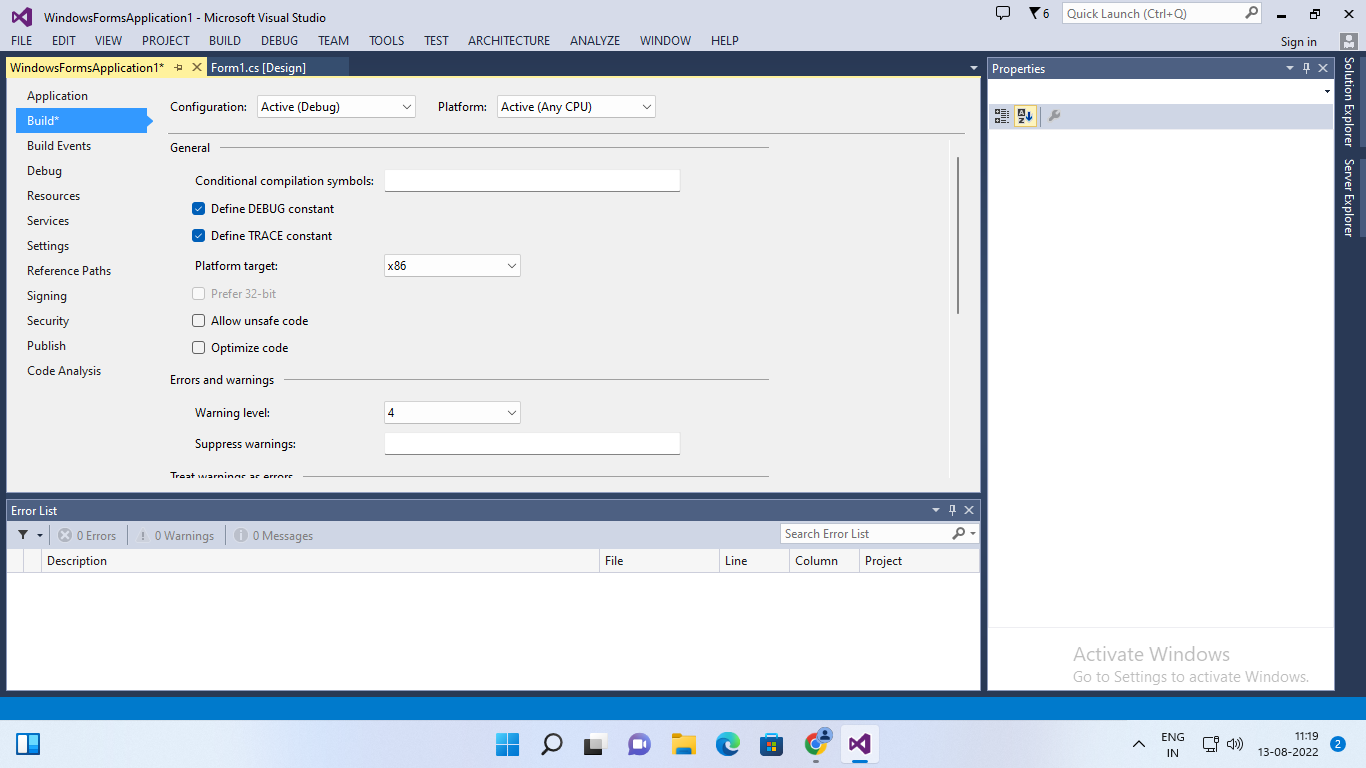
Step1:

i) Create new project and select “windows Form application”, select .NET Framework as 2.0 in Visual C#.



ii) Right Click on properties Click on open click on build Select Platform Target and select x86.

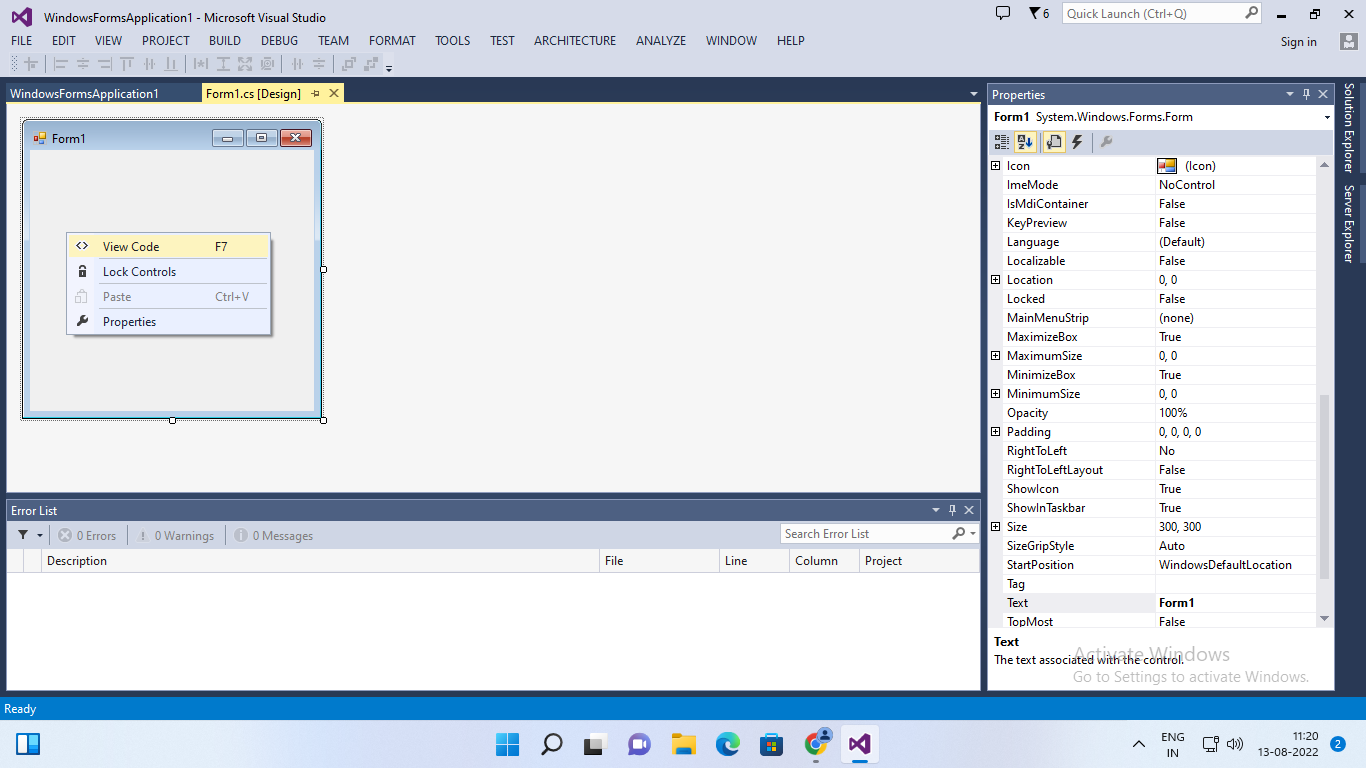


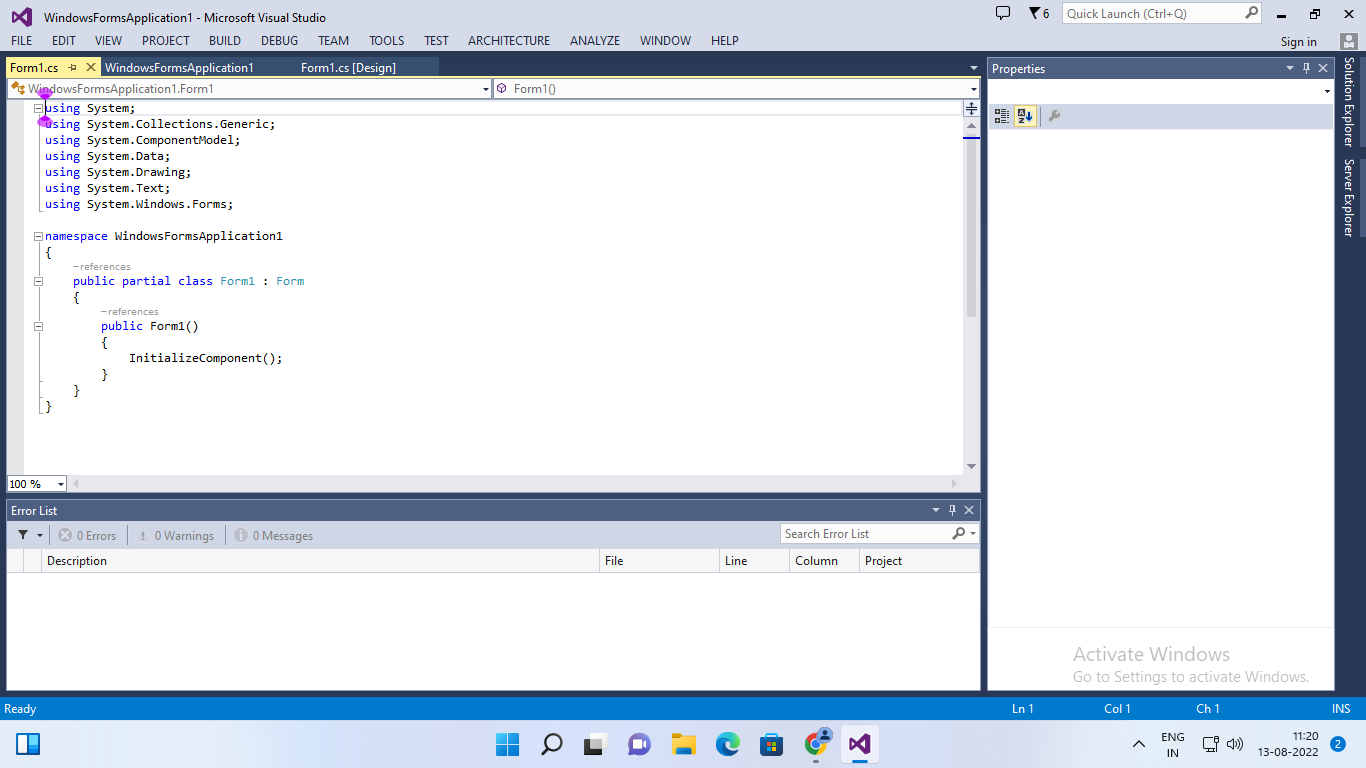


Step2:

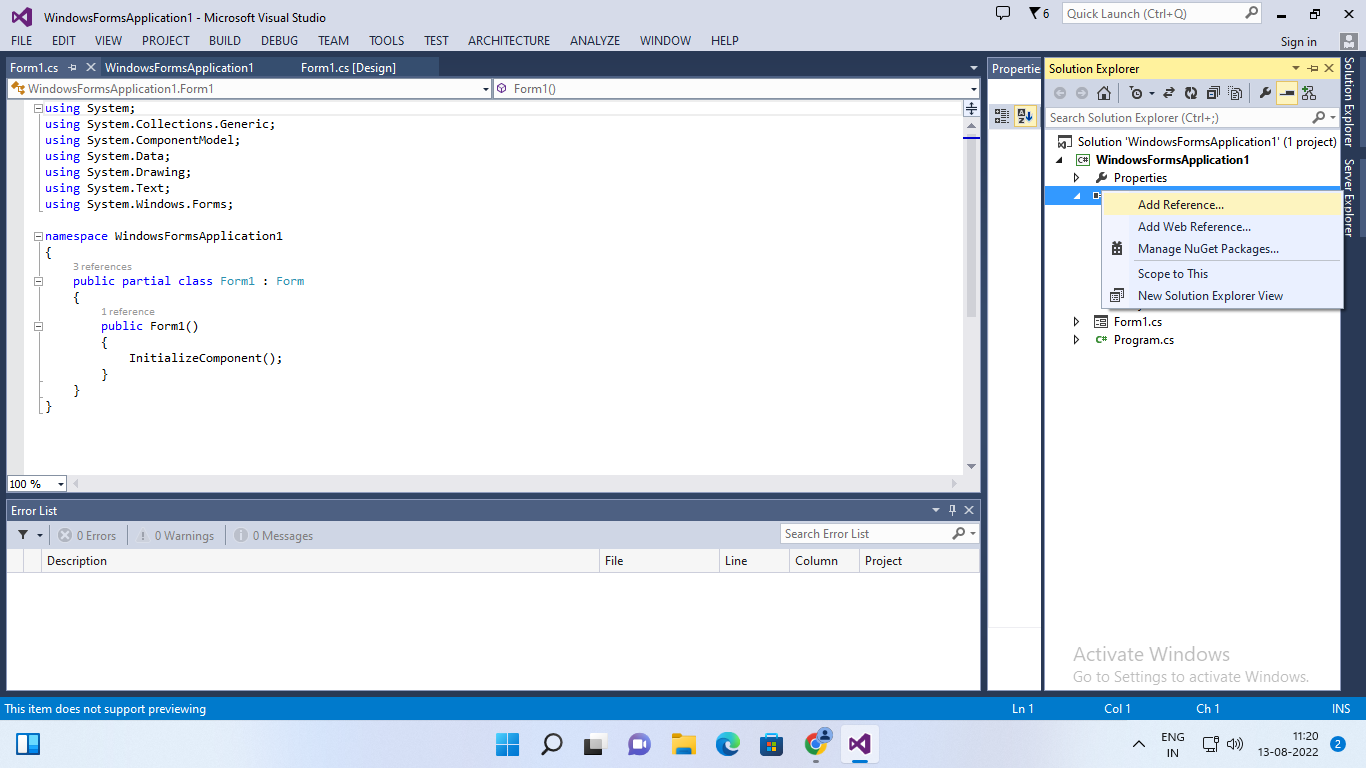
Right Click on the Form.

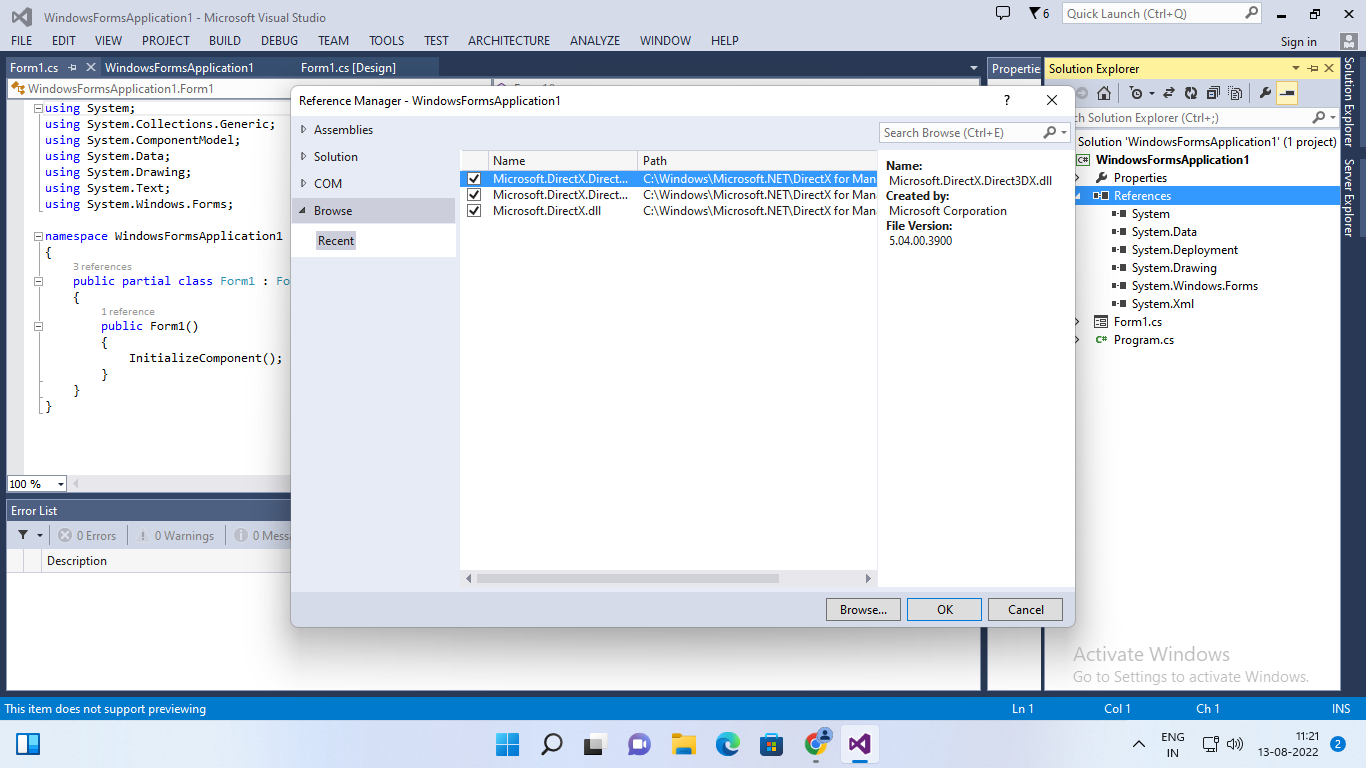
Click on View Code of Form 1.

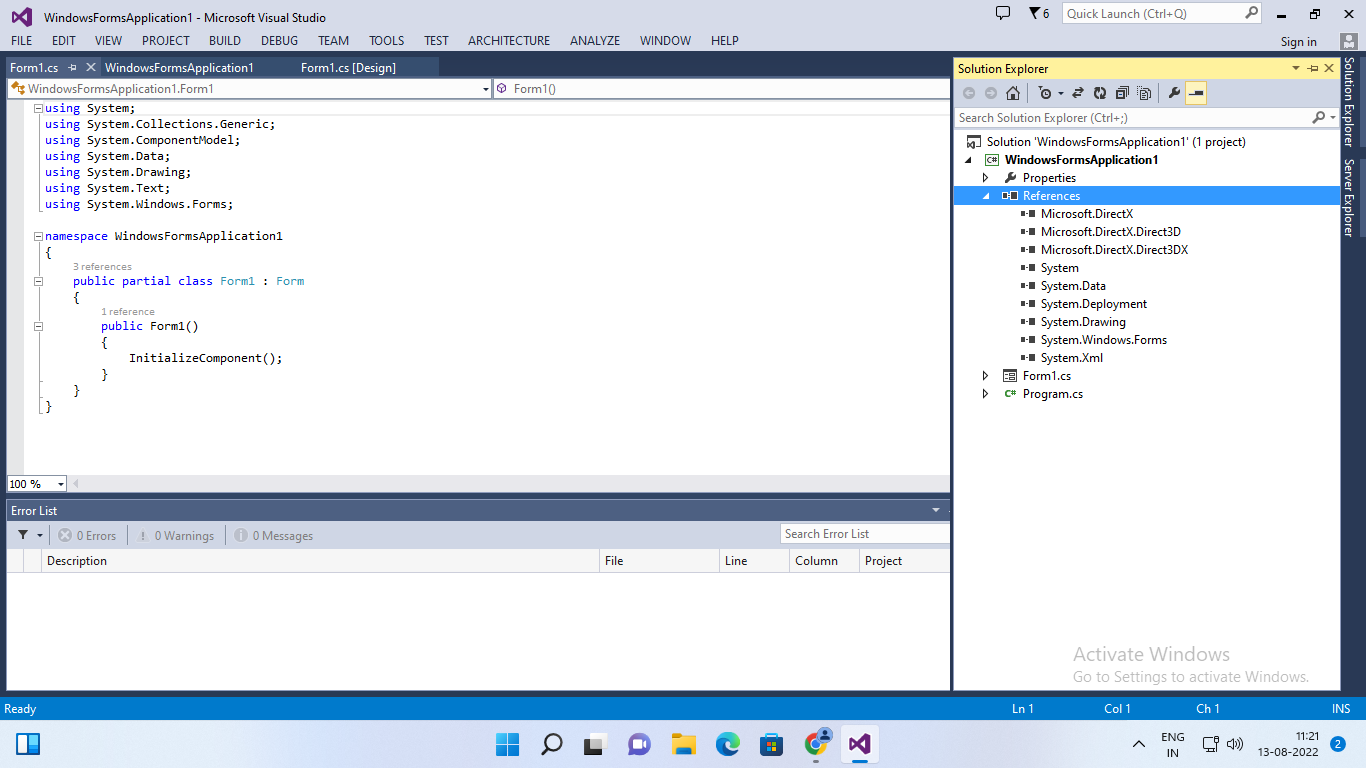




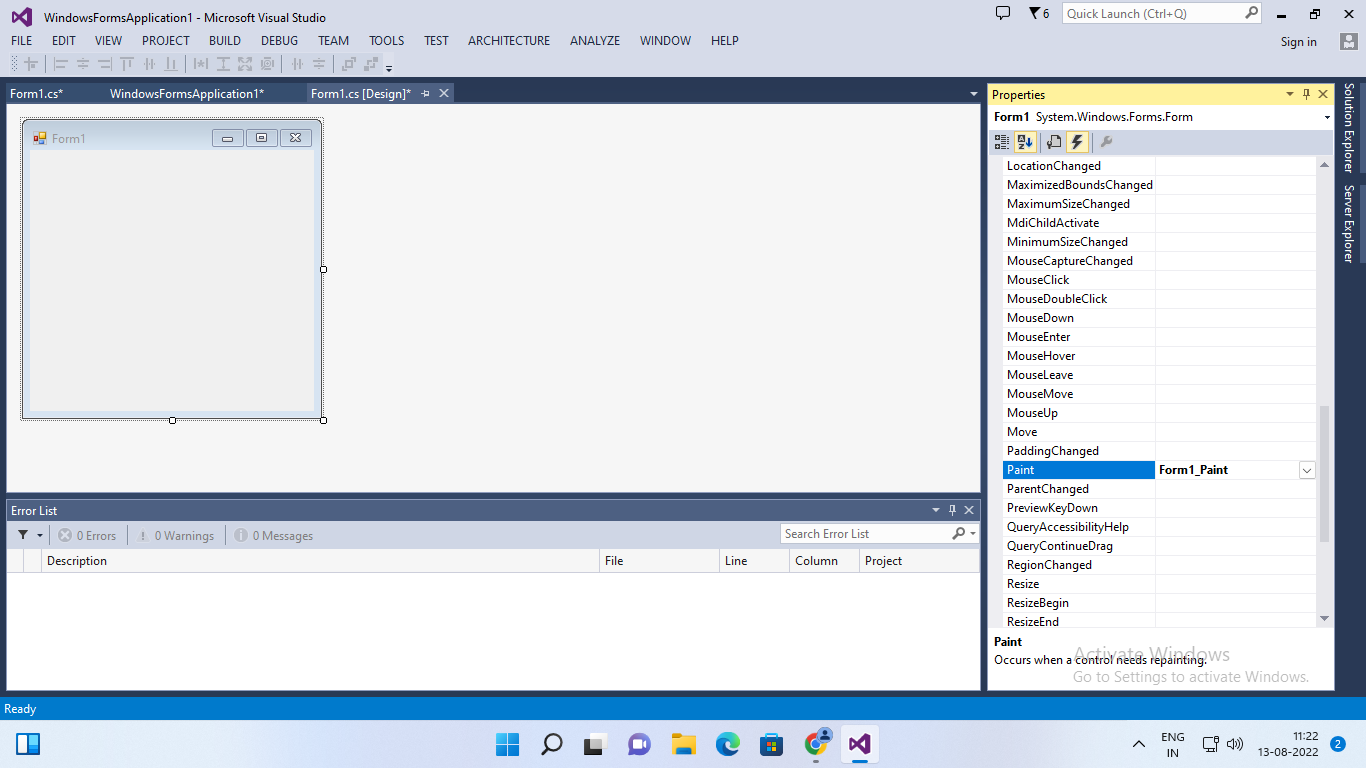
Step3: Go to Solution Explorer, right click on project name, and select Add Reference. Click on Browse and select the given .dll files which are “Microsoft.DircectX”,  “Microsoft.DirectX.Direct3D”, and “Microsfit.Directx.Direct3D.DirectX3DX”.

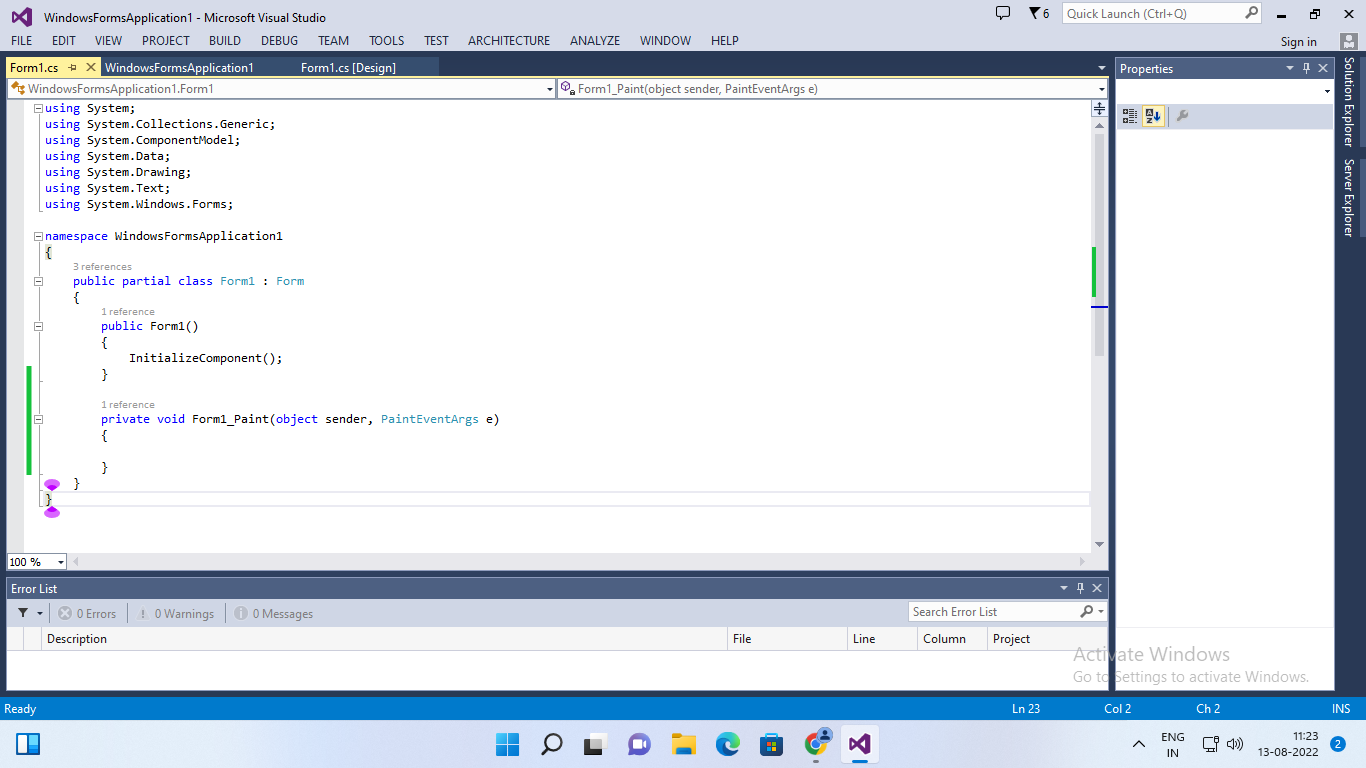






Step 4: Go to properties Section of Form (Design), double click Paint in the Event List (THUNDER SYMBOL) and enter as Form1\_Paint.





Step 5:  Edit the Form’s C# code file. Namespace must be as same as your project name.

CODE:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;

using Microsoft.DirectX;

using Microsoft.DirectX.Direct3D;

namespace WindowsFormsApplication3

{

public partial class Form1 : Form

{

private Microsoft.DirectX.Direct3D.Device device;

private CustomVertex.PositionTextured[] vertex = new CustomVertex.PositionTextured[3];

private Texture texture;

public Form1()

{

InitializeComponent();

InitDevice();

}

private void InitDevice()

{

PresentParameters pp = new PresentParameters();

pp.Windowed = true;

pp.SwapEffect = SwapEffect.Discard;

device = new Device(0, DeviceType.Hardware, this, CreateFlags.HardwareVertexProcessing, pp);

device.Transform.Projection = Matrix.PerspectiveFovLH(3.14f/4,device.Viewport.Width/device.Viewport.Height, 1f , 1000f);

device.Transform.View = Matrix.LookAtLH(new Vector3(0, 0, 20), new Vector3() , new Vector3(0,1,0));

device.RenderState.Lighting = false;

vertex[0] = new CustomVertex.PositionTextured(new Vector3(0, 0, 0), 0, 0);

vertex[1] = new CustomVertex.PositionTextured(new Vector3(5, 0, 0), 0, 1);

vertex[2] = new CustomVertex.PositionTextured(new Vector3(0, 5, 0), -1, 1);

texture = new Texture(device, new Bitmap("E:\\TYCS\\img1.png"), 0, Pool.Managed);

}

private void Form1\_Load(Object Sender, EventArgs e)

{ }

private void Form1\_Paint(object sender, PaintEventArgs e)

{

device.Clear(ClearFlags.Target, Color.CornflowerBlue, 1, 0);

device.BeginScene();

device.SetTexture(0, texture);

device.VertexFormat = CustomVertex.PositionTextured.Format;

device.DrawUserPrimitives(PrimitiveType.TriangleList, vertex.Length / 3, vertex);

device.EndScene();

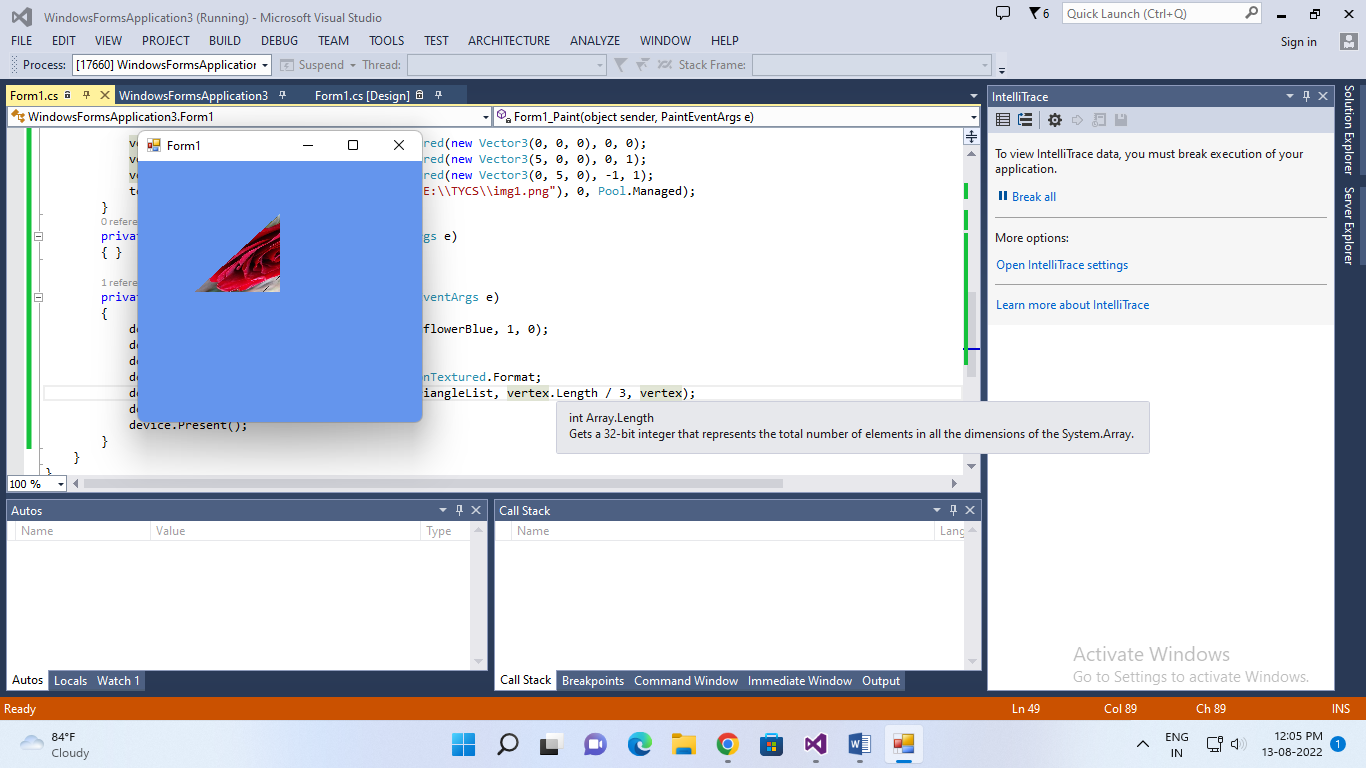
device.Present();

}

}

}

OUTPUT:



PRACTICAL 4

AIM:

Steps

Step1:

i) Create new project and select “windows Form application”, select .NET Framework as 2.0 in Visual C#.

ii) Right Click on properties Click on open click on build Select Platform Target and select x86.

Step2:

Right Click on the Form.

Click on View Code of Form 1.

Step3: Go to Solution Explorer, right click on project name, and select Add Reference. Click on Browse and select the given .dll files which are “Microsoft.DircectX”,  “Microsoft.DirectX.Direct3D”, and “Microsfit.Directx.Direct3D.DirectX3DX”.

Step 4: Go to properties Section of Form (Design), double click Paint in the Event List (THUNDER SYMBOL) and enter as Form1\_Paint.

Step 5:  Edit the Form’s C# code file. Namespace must be as same as your project name.

CODE:

OUTPUT: