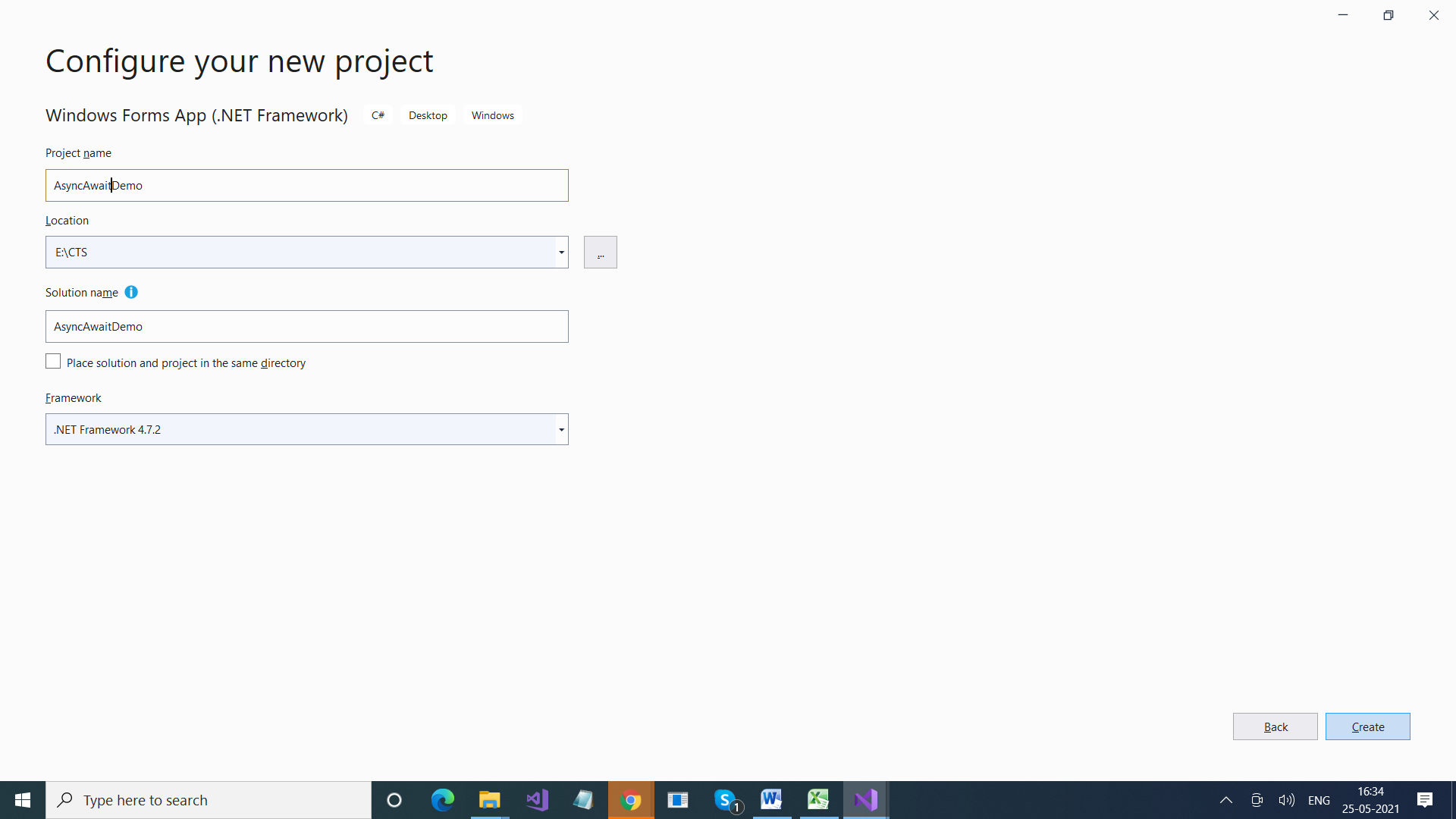
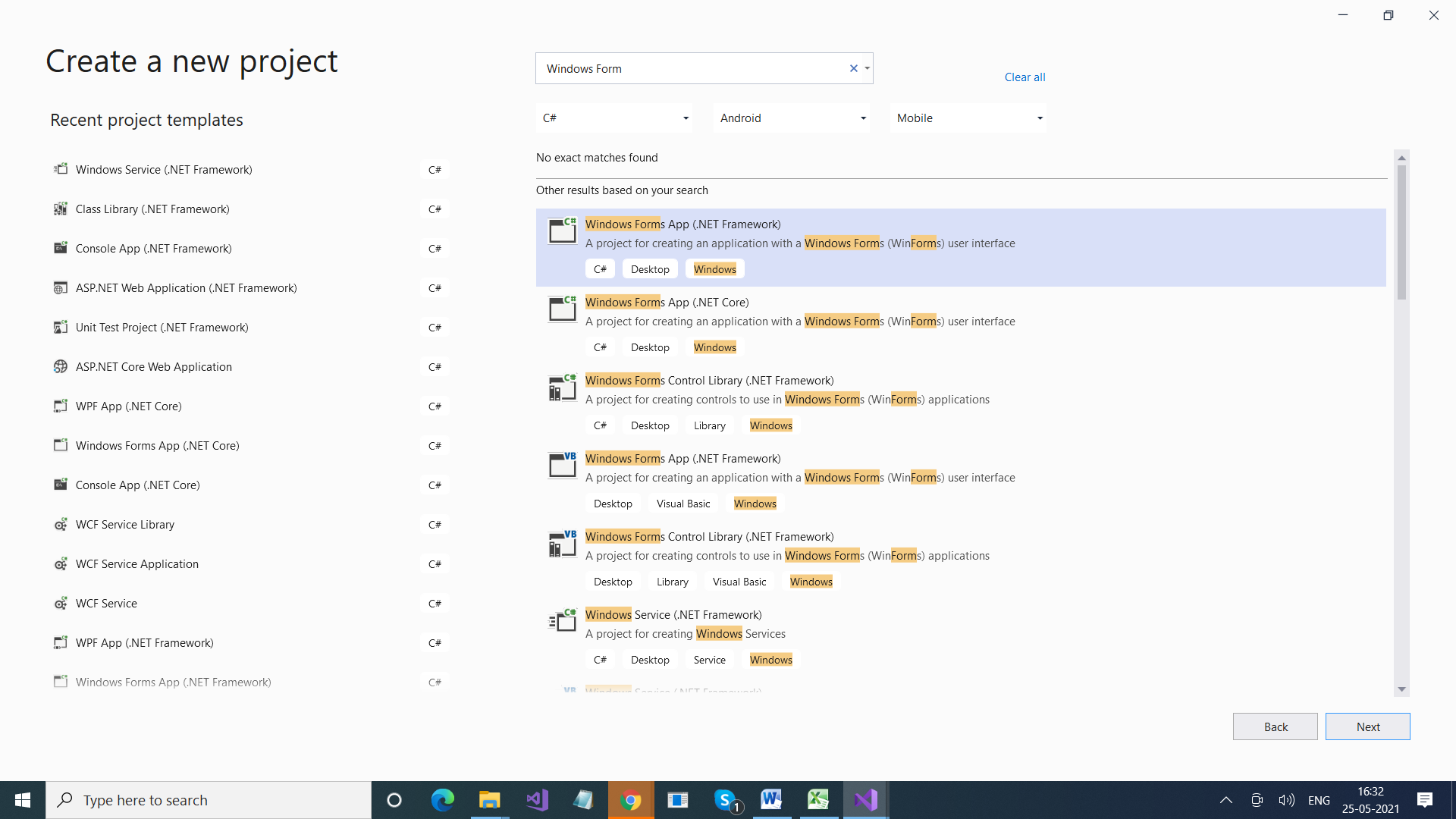
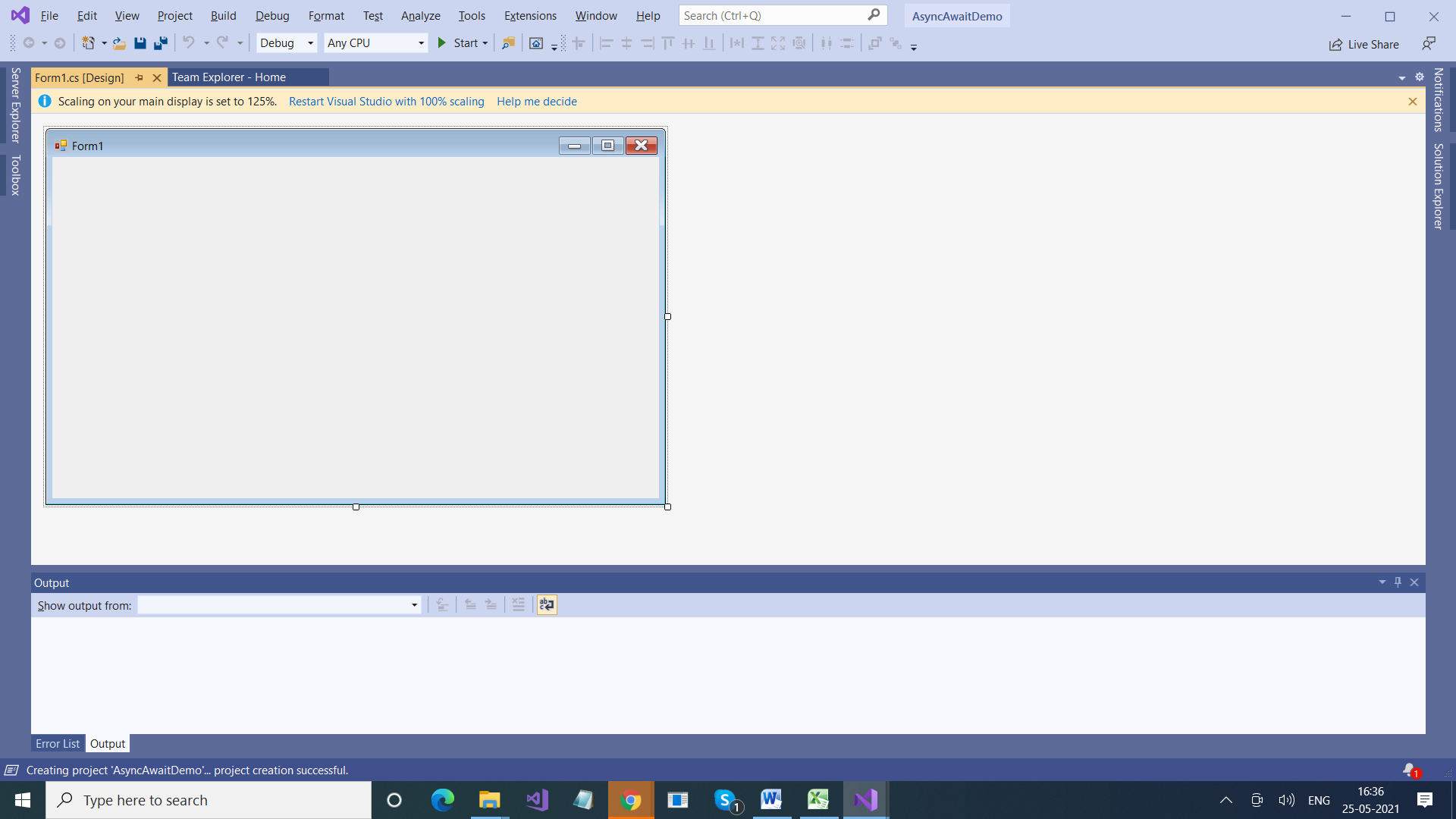
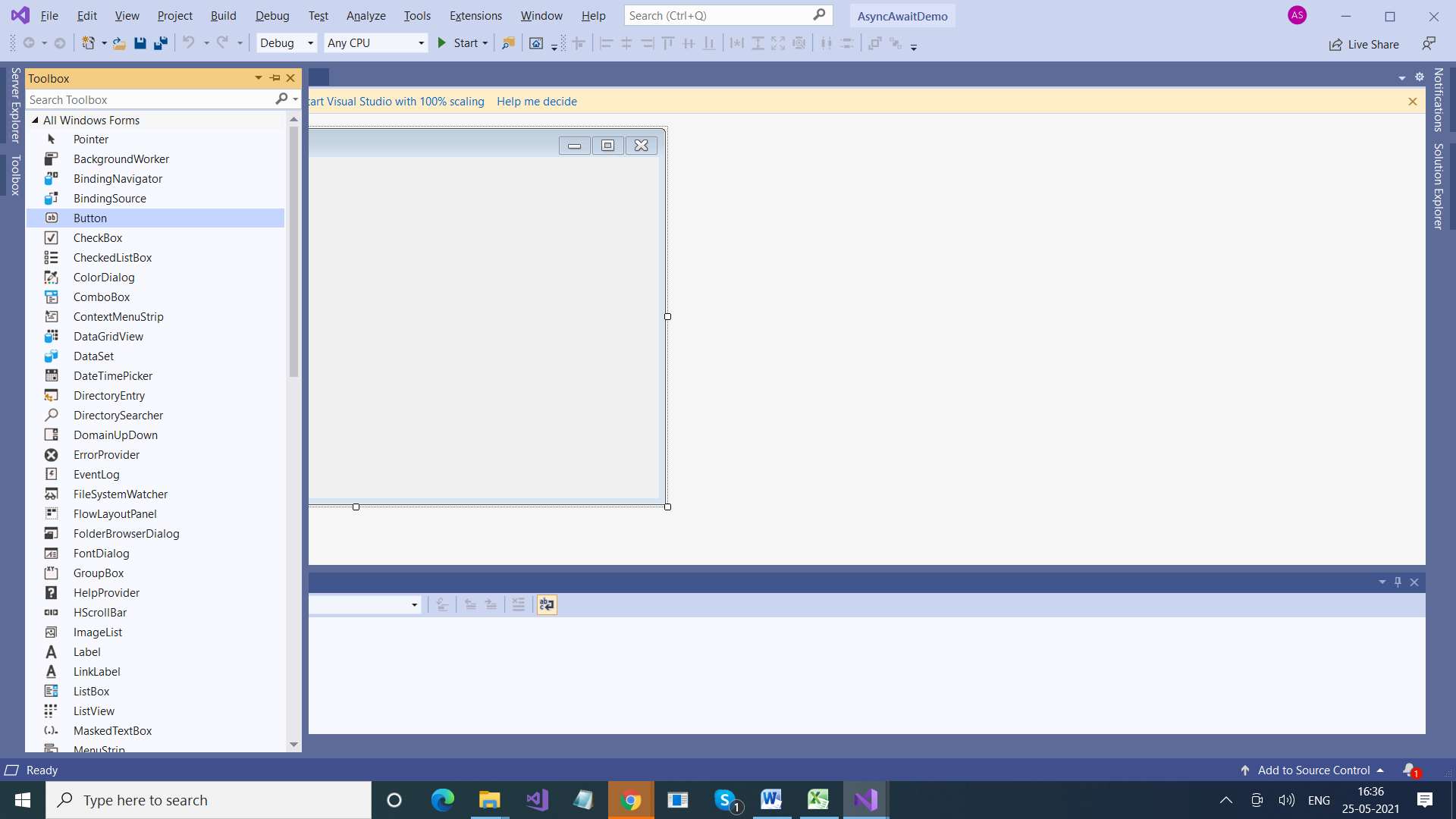
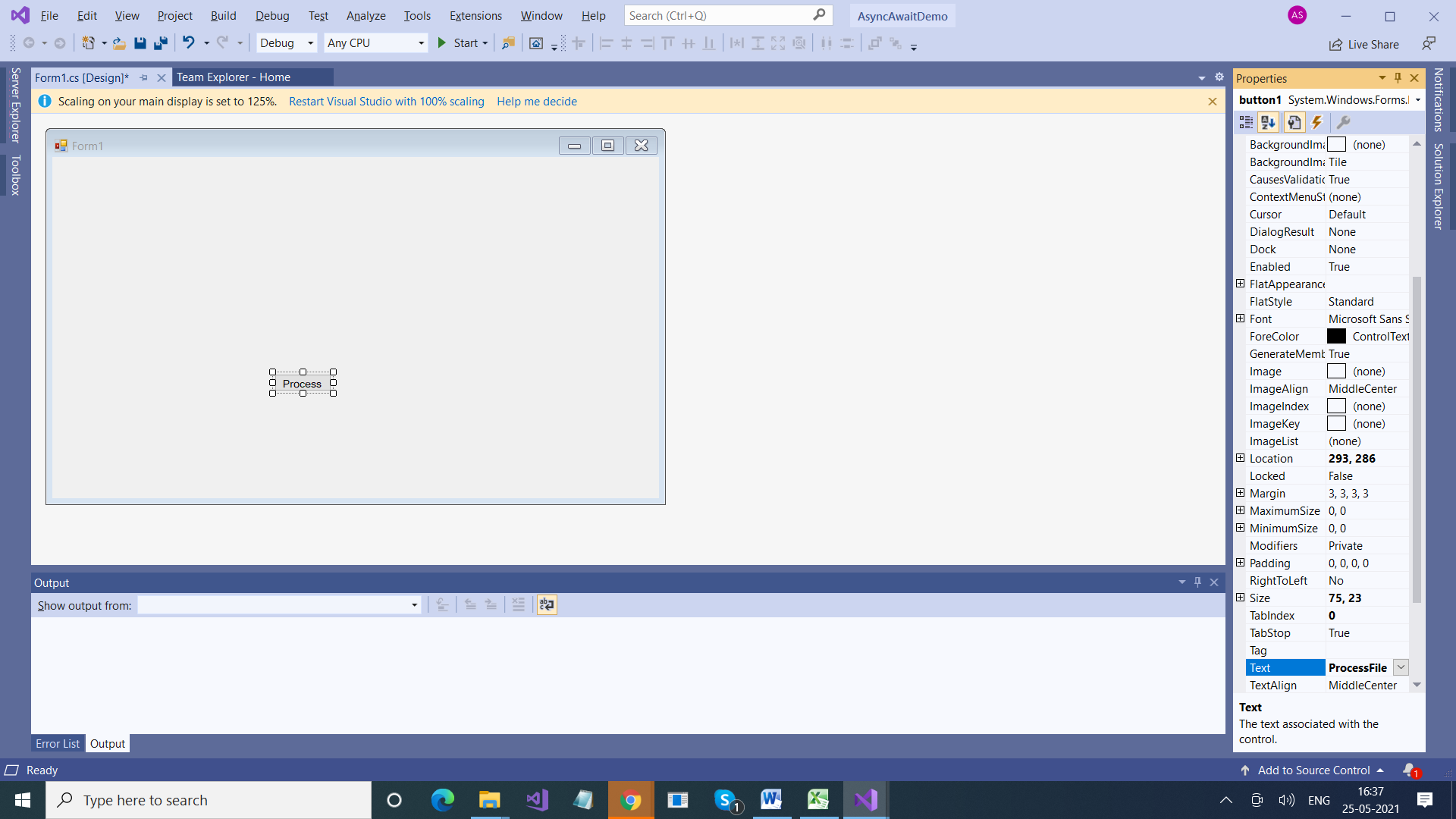
Async Await

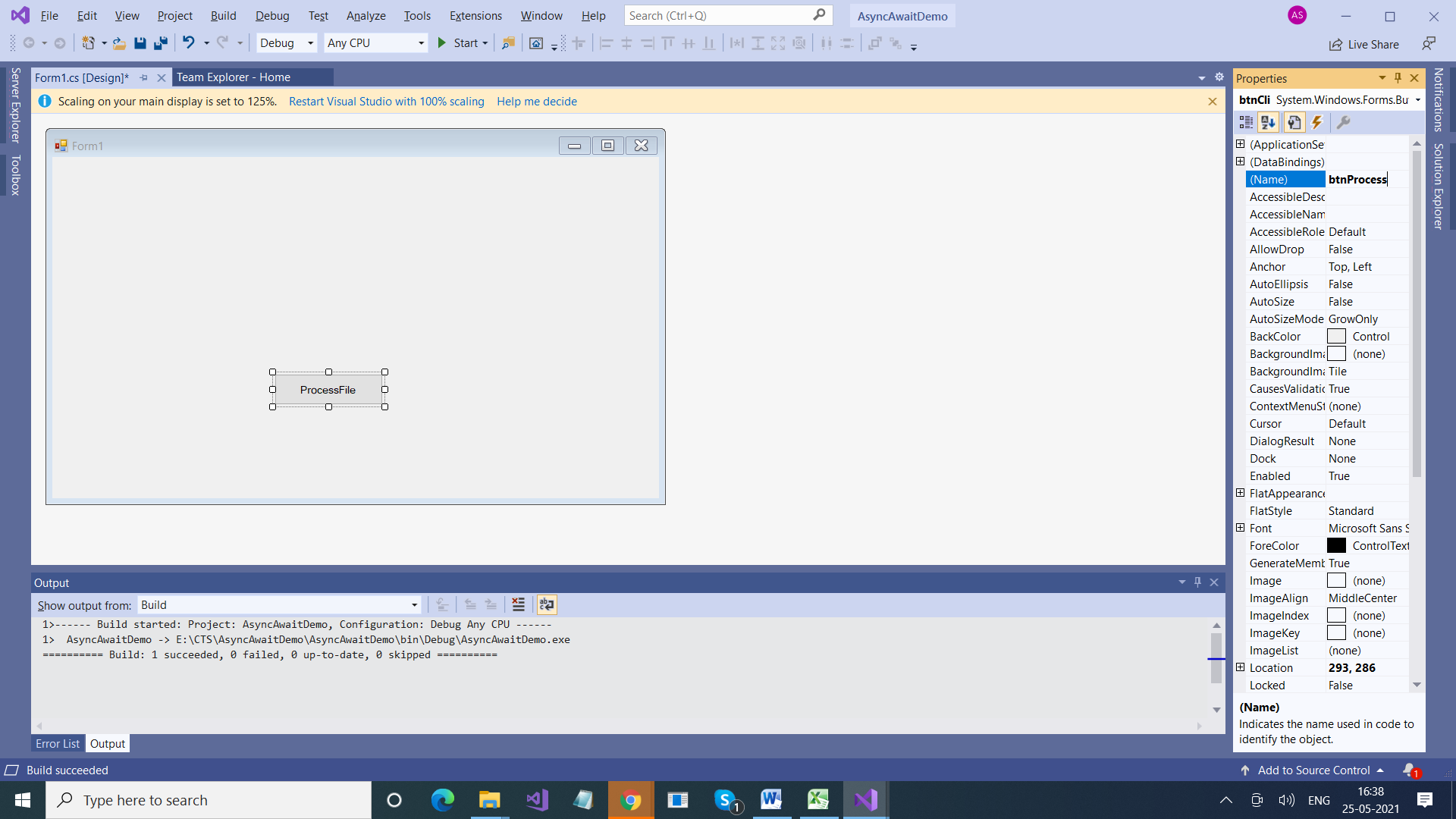
Create a Windows Application

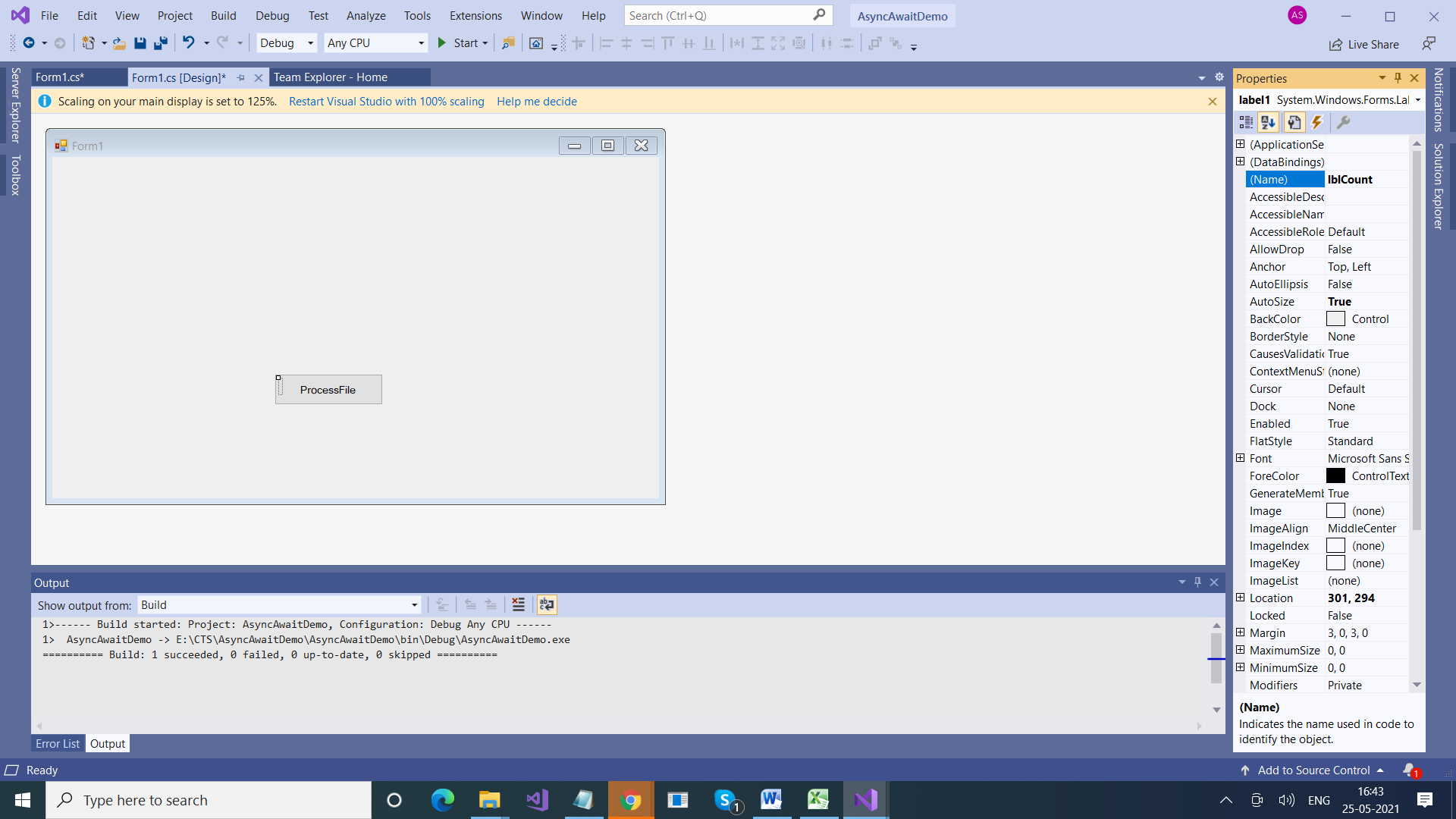












using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AsyncAwaitDemo

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

int CountCharacters()

{

int count = 0;

using (StreamReader reader = new StreamReader(@"E:\data\data.txt"))

{

string content = reader.ReadToEnd();

count = content.Length;

Thread.Sleep(5000);

}

return count;

}

private void btnProcess\_Click(object sender, EventArgs e)

{

lblCount.Text = "Processing File, please wait .... ";

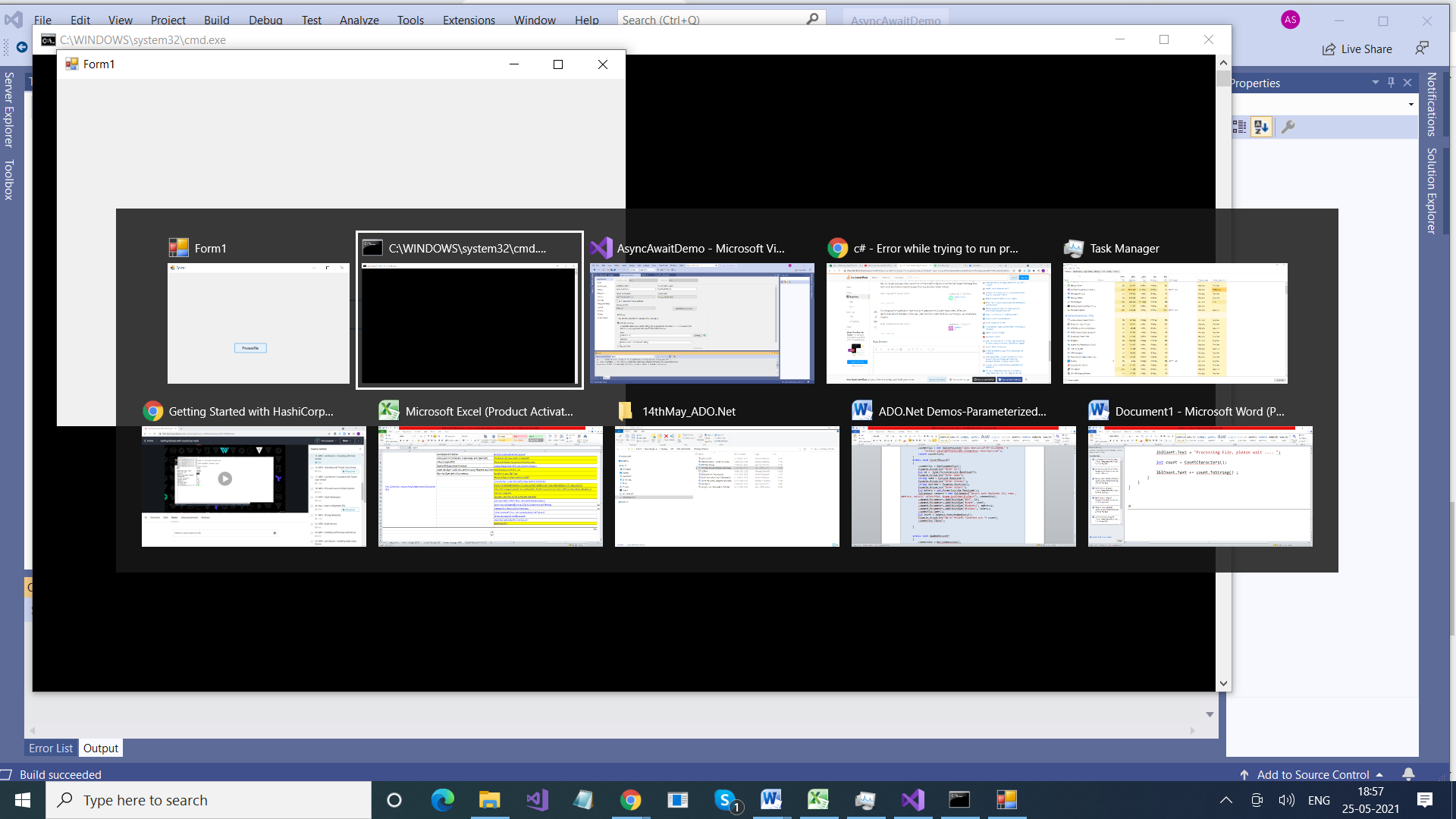
int count = CountCharacters();

lblCount.Text += count.ToString() ;

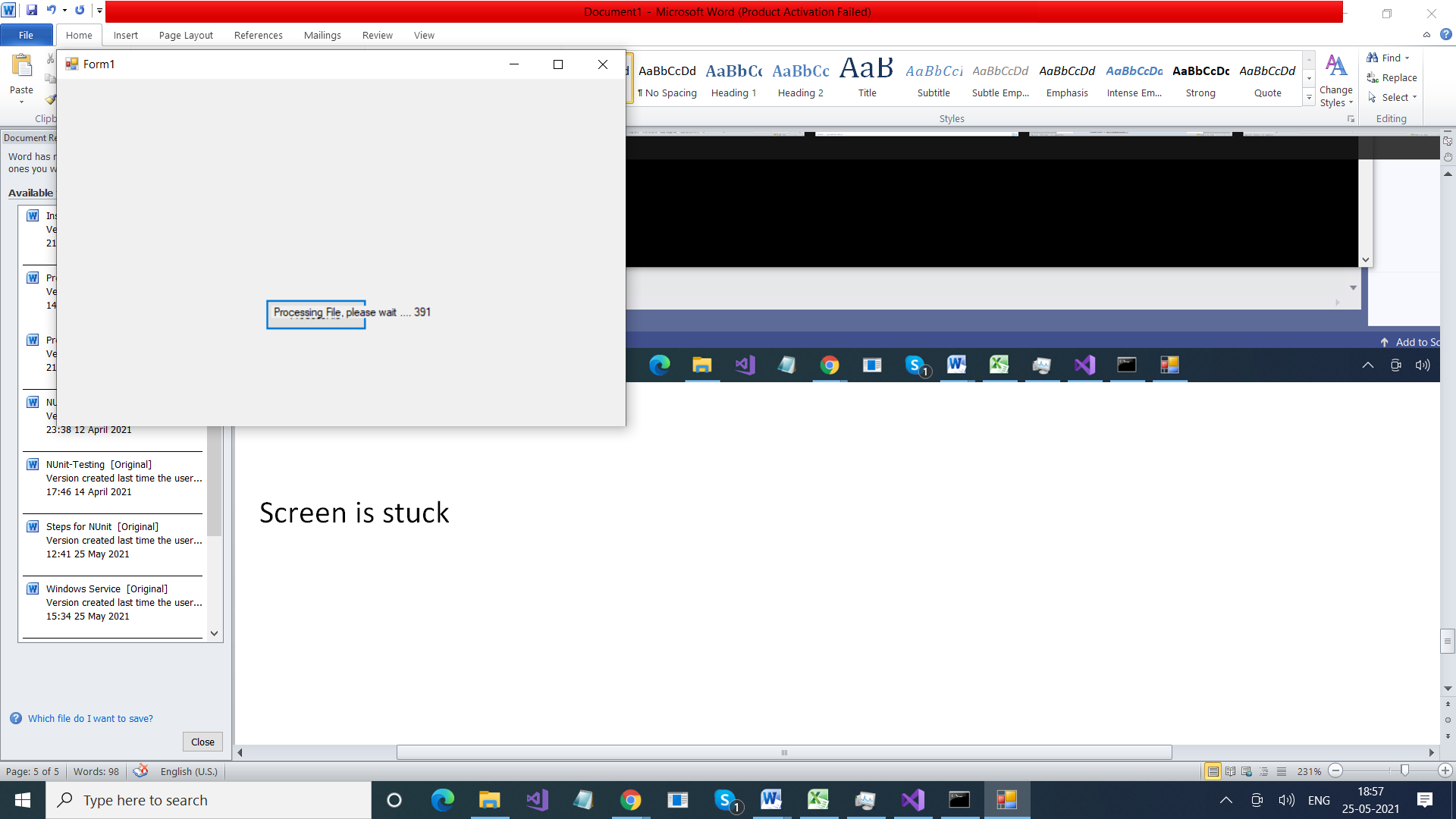
}

}

}



Screen is stuck



private async void btnProcess\_Click(object sender, EventArgs e)

{

Task<int> task = new Task<int>(CountCharacters);

task.Start();

lblCount.Text = "Processing File, please wait .... ";

int count = await task;

lblCount.Text += count.ToString();

}