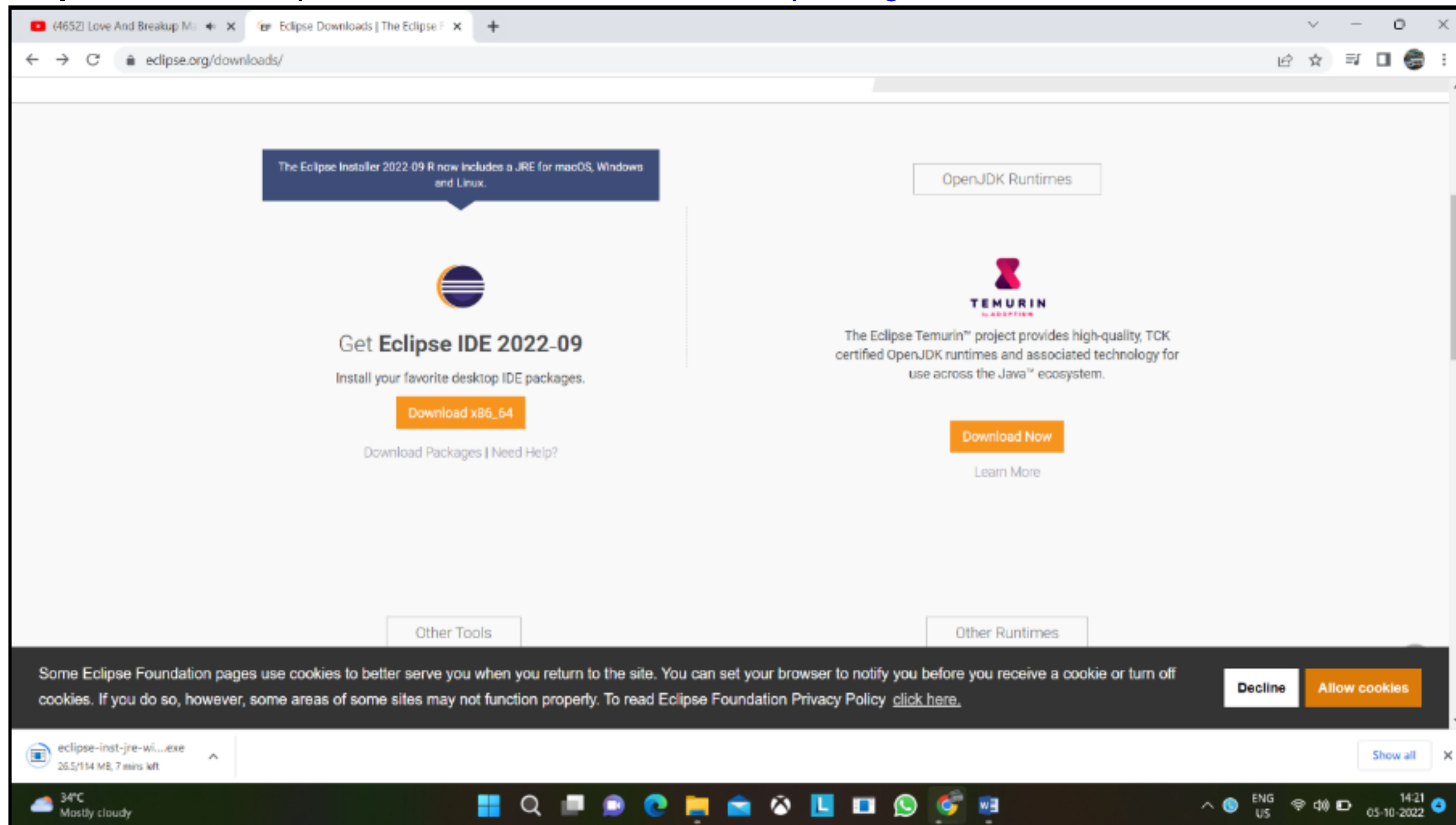
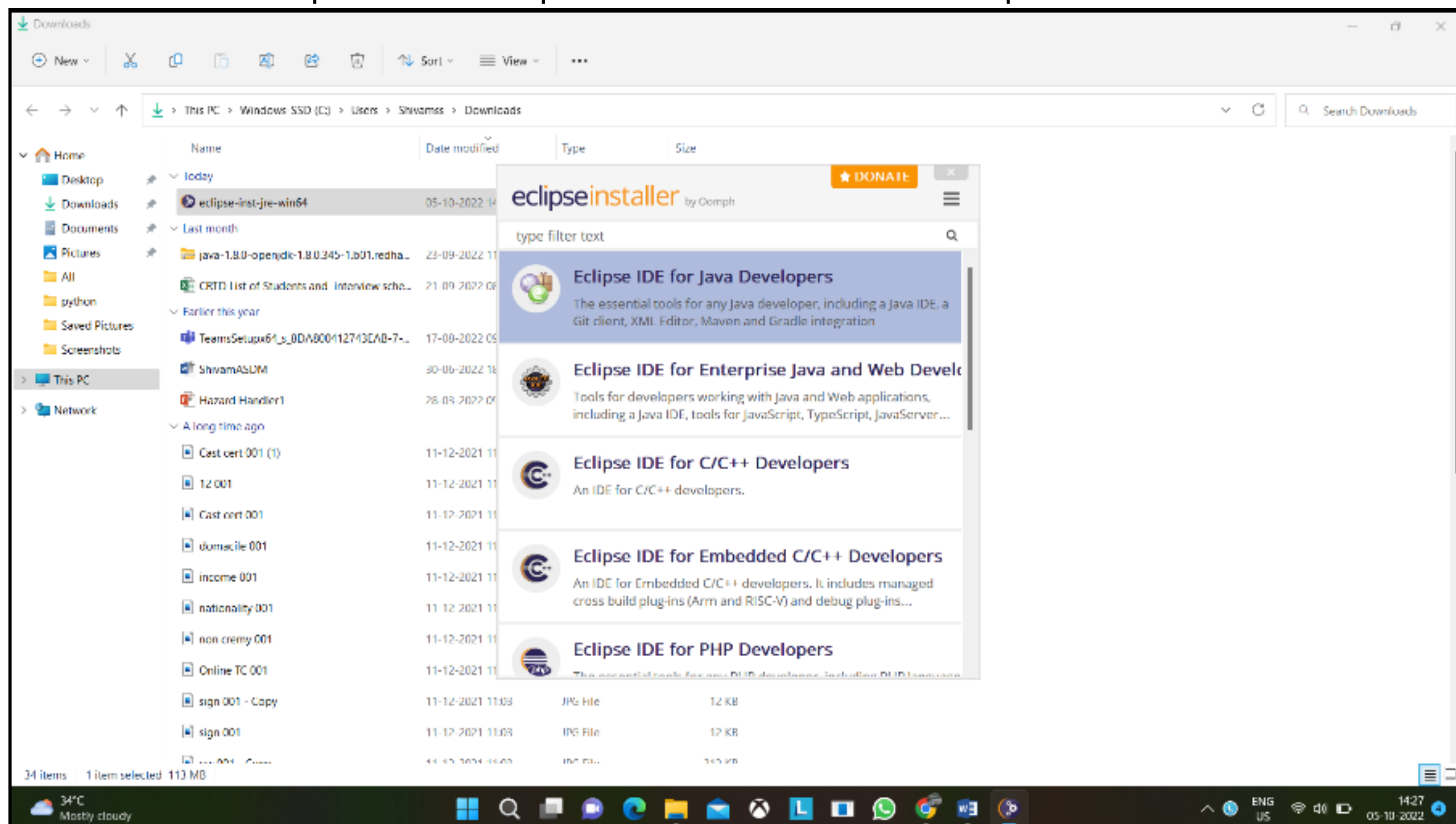


Assignment: Study and do the Configuration of cloudsim. Also execute & check the performance of existing algorithms.

Step 1: To Install Eclipse IDE, Go to Website → www.eclipse.org → Then Click Download file for x86_64

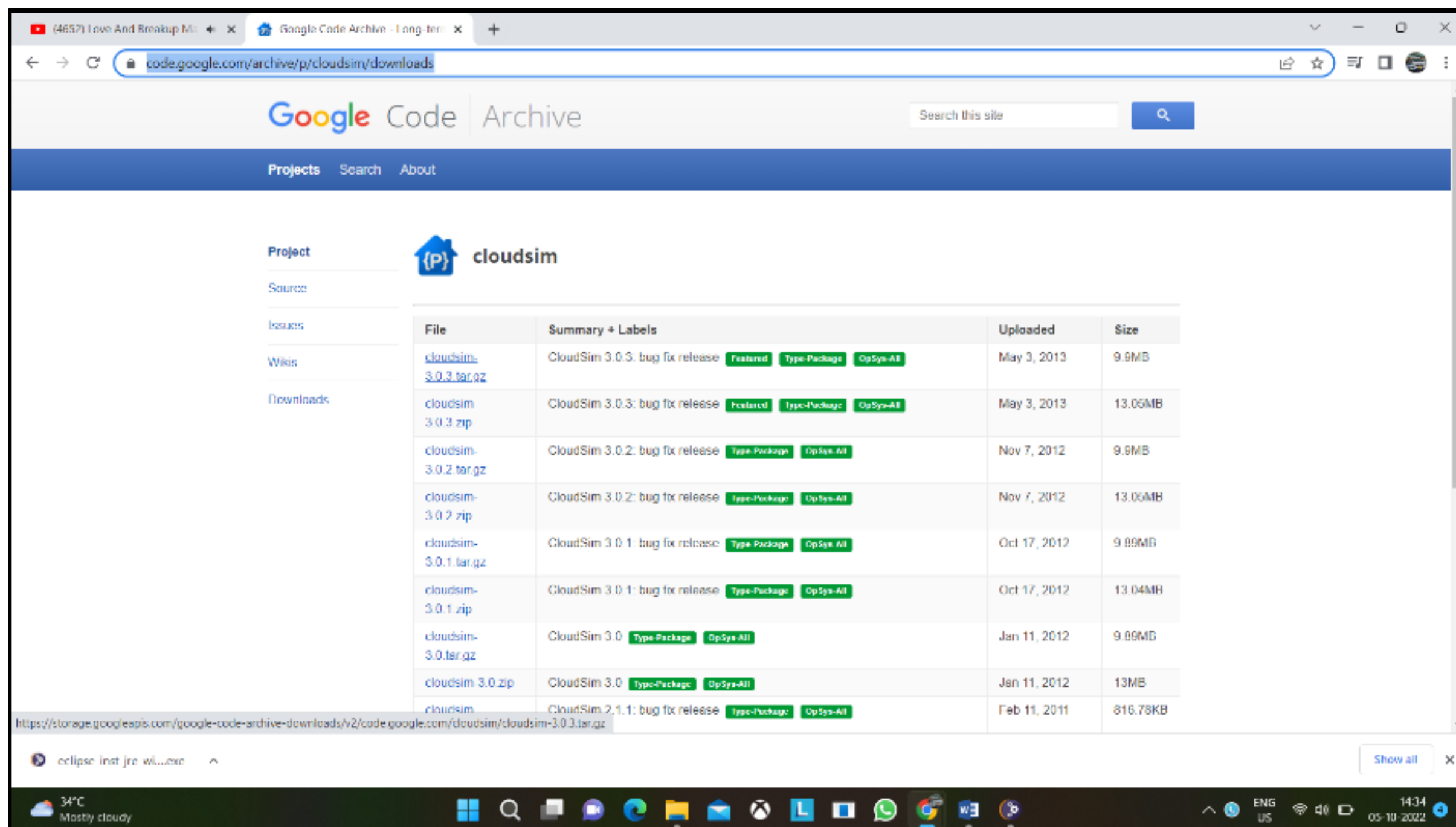


→ Extract the file → open to Install Eclipse IDE → Select for Java Developer to install



Now we in install jar files of cloudsim

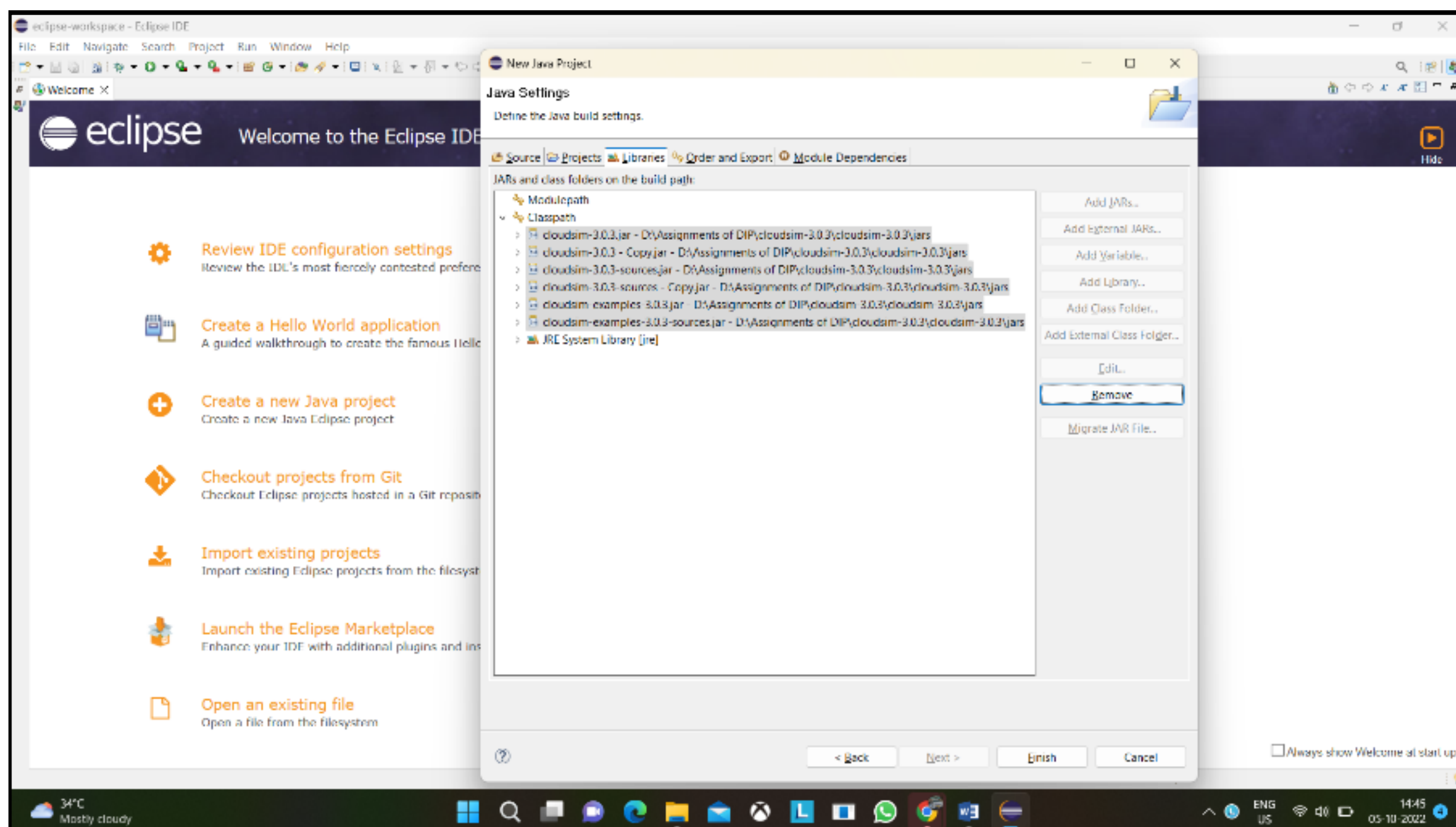
Step 2: Go To <https://code.google.com/archive/p/cloudsim/downloads>



Select cloudsim 3.0.3 tar.gz file & download it and extract it.

Step 3: Before importing jar files place jar files in same project folder

Open Eclipse IDE → Click on File → New → Project → Java Project → Name the project & click on next button → Select Libraries → Click on Add Jars → Select cloudsim folder and go to jar folder select jar files
Done



Step 4: To add jar files in existing project do as follows Right Click on Module Name → Select Build Path Option or go to properties → Go to Libraries → Click on classpath → do as step 3

Assignment: Creating a Warehouse Application in Salesforce.com.

Step 1: Log into your Sandbox or Developers Organization.

<https://developer.salesforce.com/signup>

Click on setup → create → objects → new custom objects.

Label: MySale

Plural Label: MySales

Object Name: MySale

Record Name: MySale Description

Data Type: Text

Click Save.

Step 2: Under MySale Go to Custom Field and Relationships → Click on New Custom Field

Creating 1st Field:

Select Data type as Auto Number → Next →

Enter the detail Field Label: PROD_ID → Next → Next → Save & New
Starting Number: 1001

Creating 2nd Field:

Select Data type as Auto Date → Next →

Enter the detail Field Label: Date of Sale → Next → Next → Save & New

Creating 3rd Field:

Select Data type as Auto Number → Next →

Enter the detail Field Label: Quantity Sold Length: 3 Decimal Places: 0 → Next →
Next → Save & New

Creating 4th Field:

Select Data type as Auto Currency → Next →

Enter the detail Field Label: Rate Length: 4 Decimal Places: 2
Default Value: 10 → Next → Next → Save & New

Creating 5th Field:

Select Data type as Auto Currency → Next →

Enter the detail Field Label: Quntity__c*Rate__c → Next → Next → Save

Step 3:

Now create a Tab

Click on setup → create → tabs → new custom tab → choose MySale → Next → Next → Save.

Step 4:

Now create an App

Click on setup → create → Apps → new → custom app → next → Enter My Shop for the App Label → Next
→ visible to all → click on save.

On the top in the tab bar you can see the tab which has been created by you click on the tab you can see your object is opened just click on new button and provide the details mentioned.

Assignment: Creating an Application in Salesforce.com using Apex programming Language.

Step 1: Log into your Sandbox or Developers Organization.

<https://developer.salesforce.com/signup>

Click on setup → create → objects → new custom objects.

Enter Book for Label.

Enter Books for Plural Label.

Click Save.

Step 2: Now let's create a custom field.

In the custom field & relationship section of the Book Object click new.

Select Number for the datatype & next.

Enter Price for the field Label.

Enter 16 in the length text box.

Enter 2 in the decimal places & Next → Next → Save.

Step 3: Click on setup → Develop → Apex classes & click new

In the class Editor enter this class

(Book__c is object API Name & Price__c is custom field API Name)

```
public class MyHelloWorld
{
    public static void applyDiscount(Book__c[] Books)
    {
        for(Book__c b:Books)
        {b.Price__c*=0.9;}
    }
}
```

Step 4: Add a Trigger

A trigger is a piece of code that can execute objects before or after specific data manipulation language events occurred.

Click on setup → create → objects → click the object you have created ex:

Book Scroll down you can see Trigger Click on New

In the trigger Editor enter this class

```
trigger HelloWorldTrigger on Book__c(before insert)
{
    Book__c[] Books=Trigger.new;
    MyHelloWorld.applyDiscount(Books);
}
```

Step 5: Now create a Tab

Click on setup → create → tabs → new custom tab → choose Book → Next → Next → Save.

Step 6: Now create an App

Click on setup → create → Apps → new → custom app → next → Enter My Book Shop for the App Label →

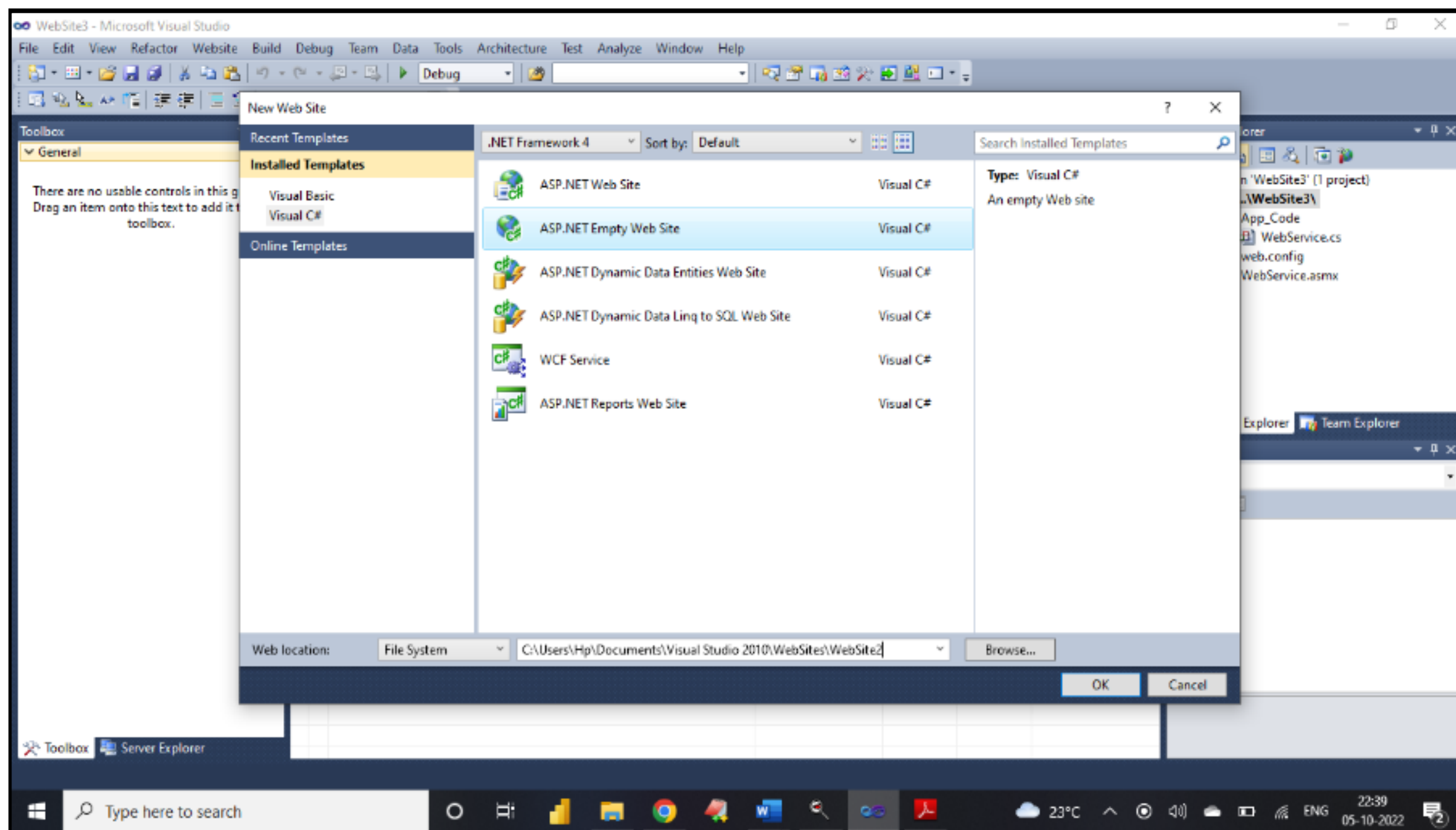
Next → visible to all → click on save.

Step 7: Now Insert a Book

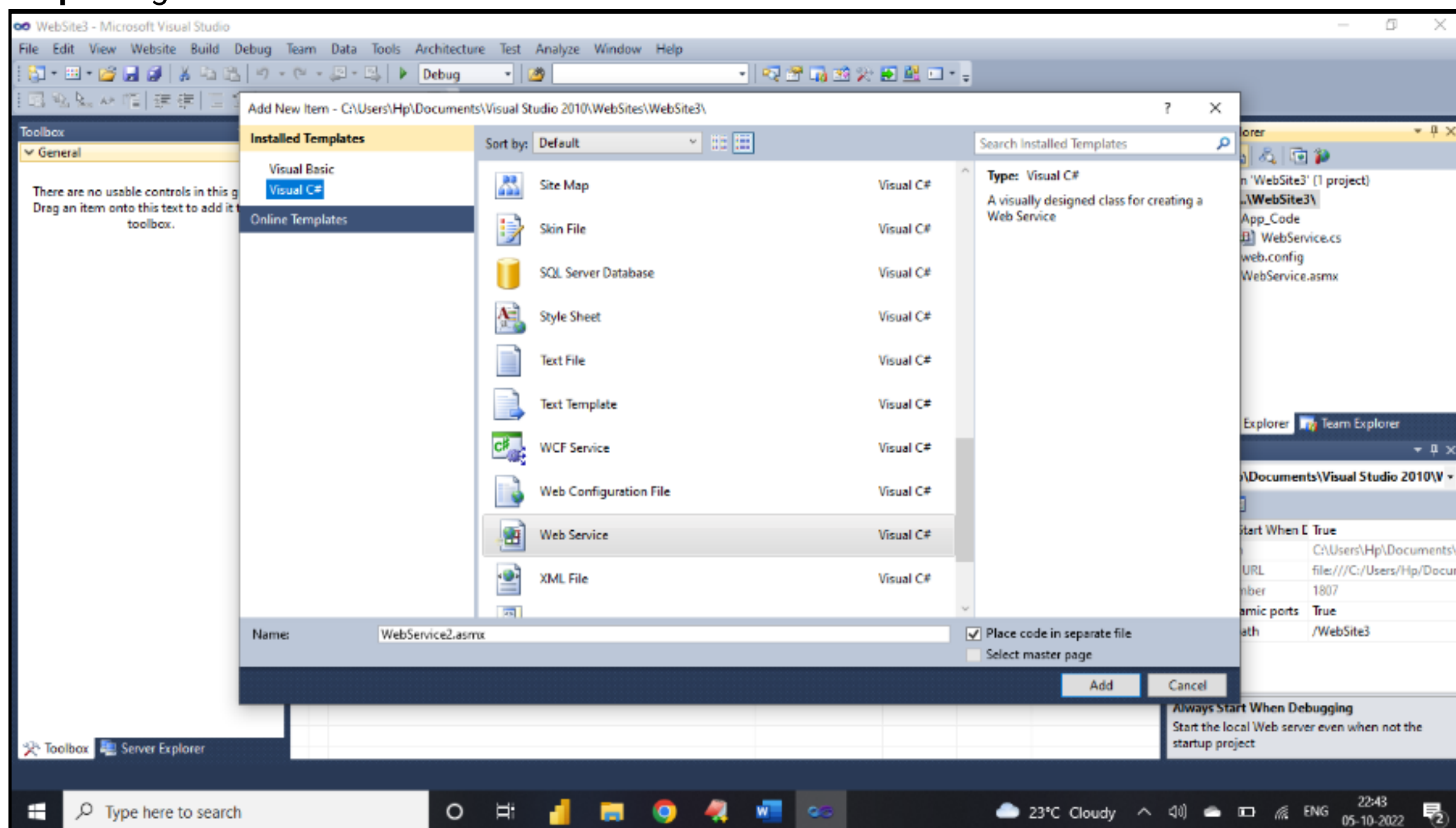
Click on My Book Shop → Books → new → insert a name for Book → insert price for that book → click on save.

Assignment: Implementation of SOAP Web services in C#

Step 1: Open Visual Studio → Go to File → Select New → Web site → ASP.NET Empty Web Site

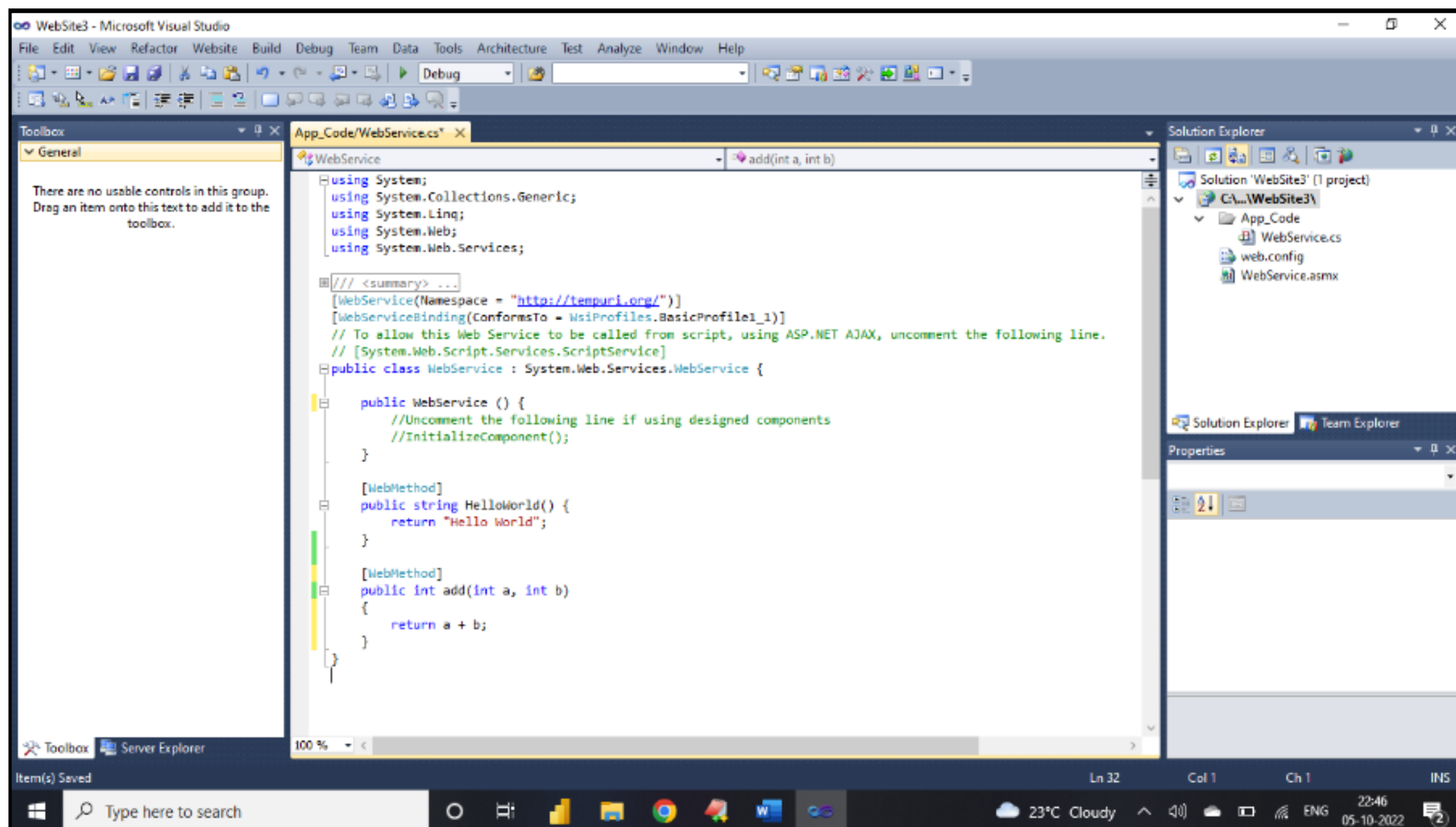


Step 2: Right Click on website Menu → Add New Item → Web Service →

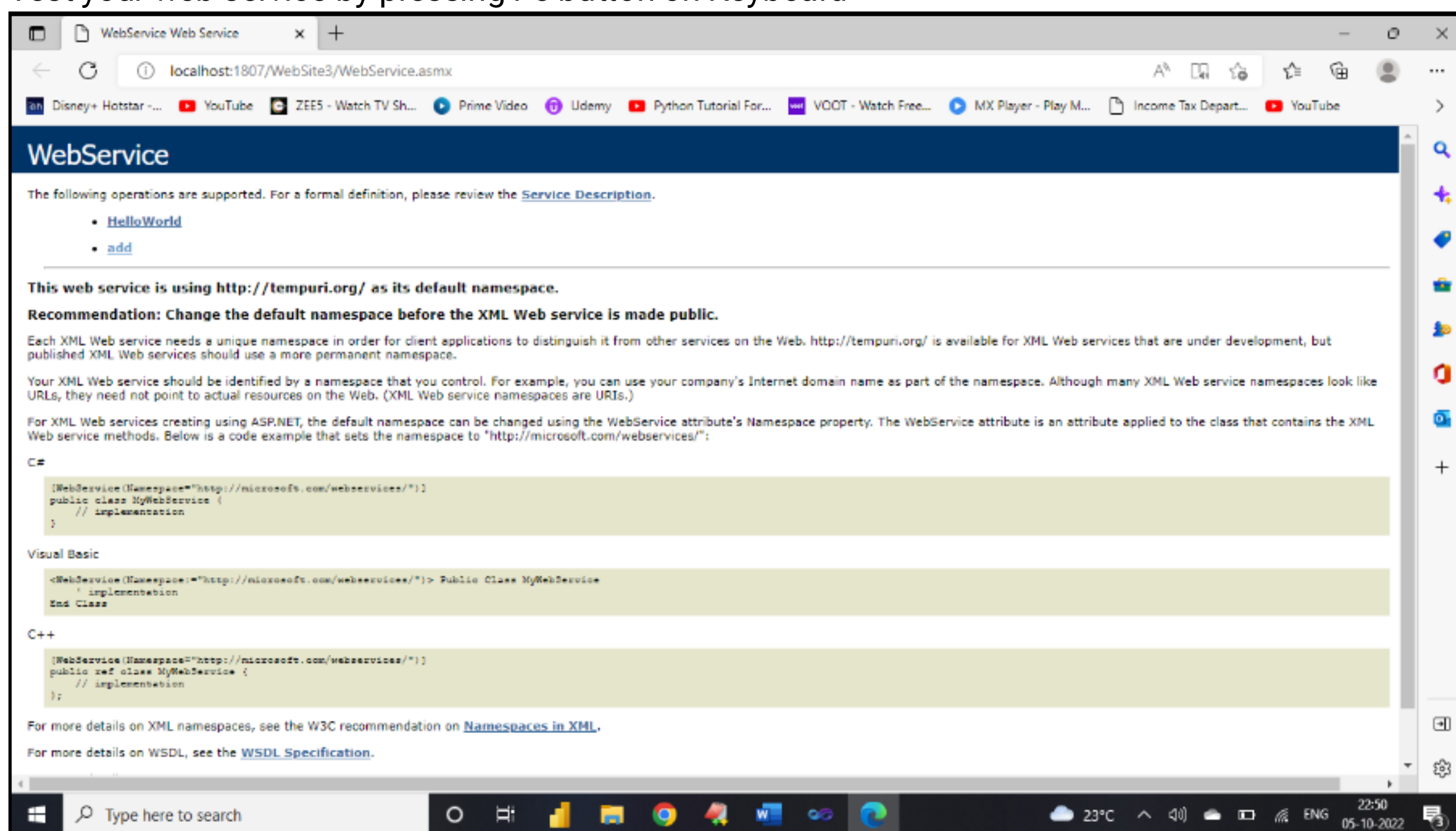


Add following code

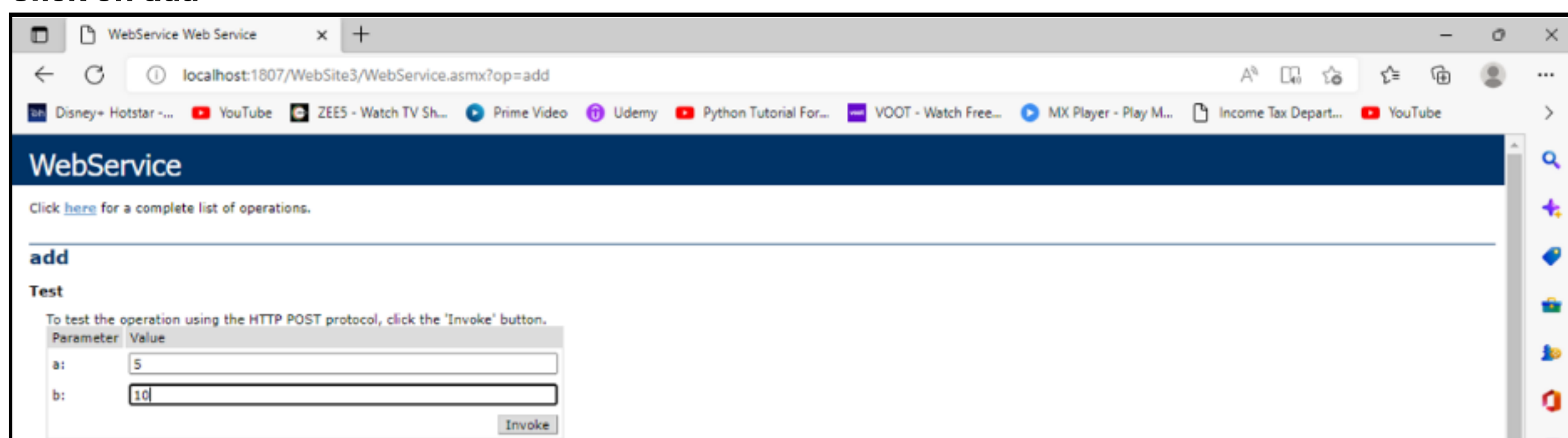
```
[WebMethod]
public int add(int a, int b)
{
    return a + b;
}
```

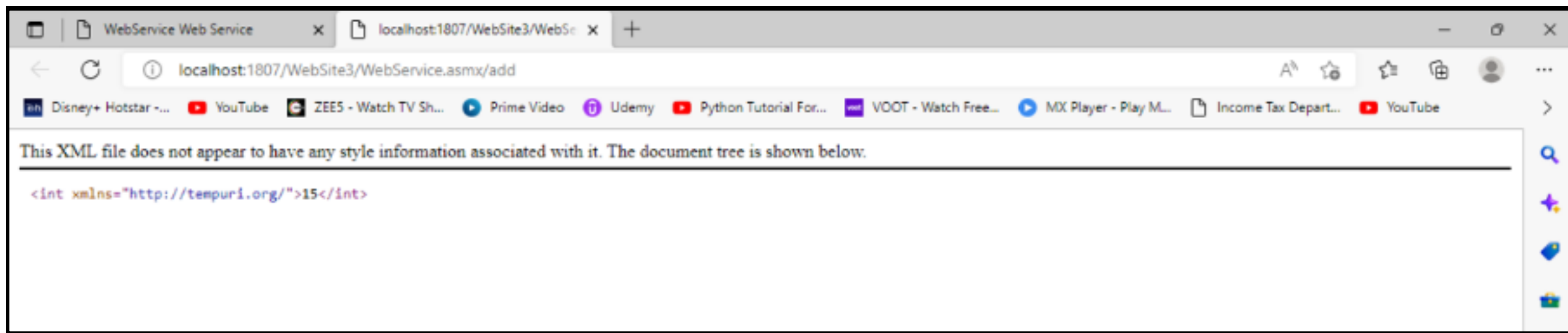


Test your web service by pressing F5 button on Keyboard



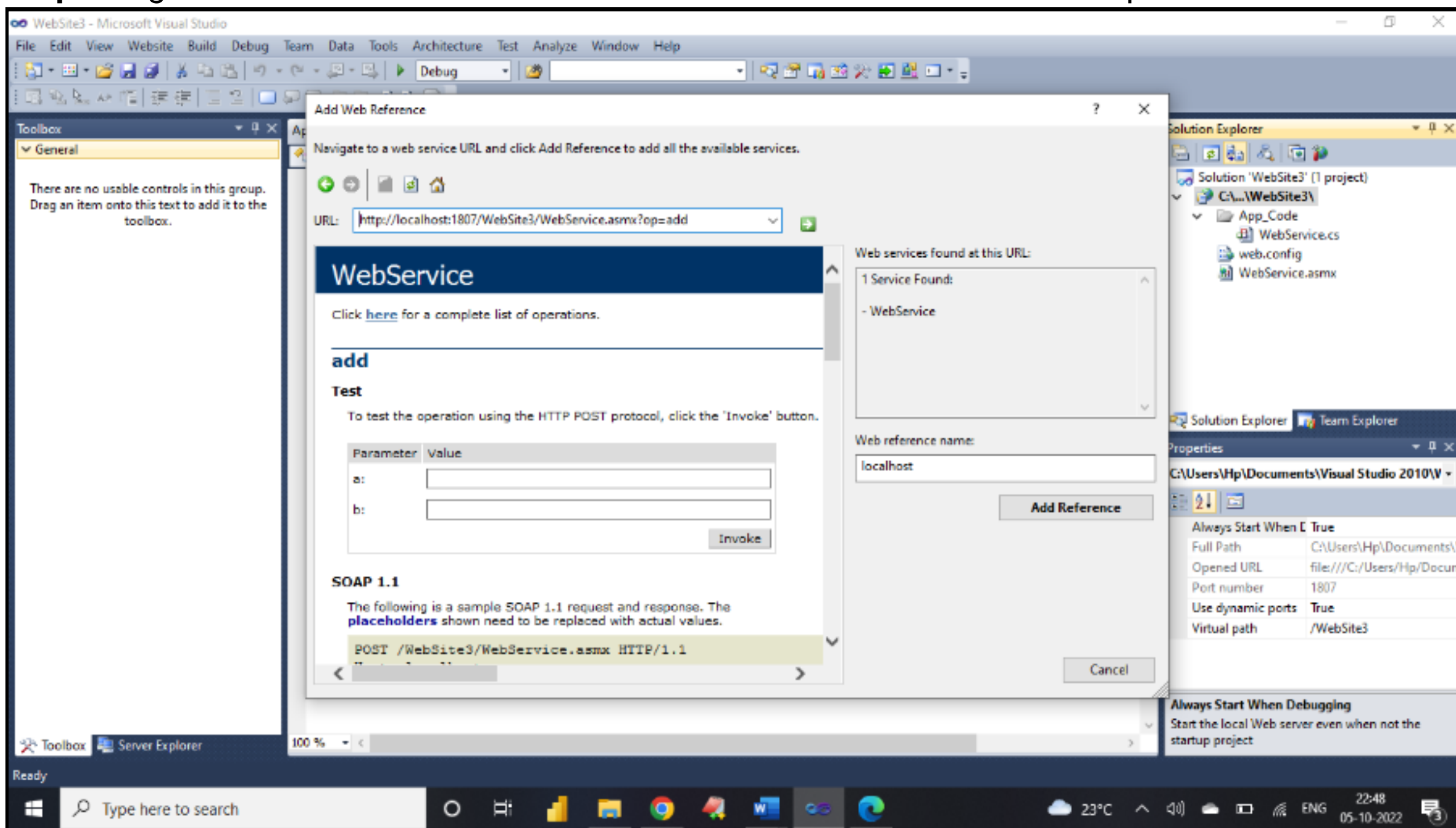
Click on add





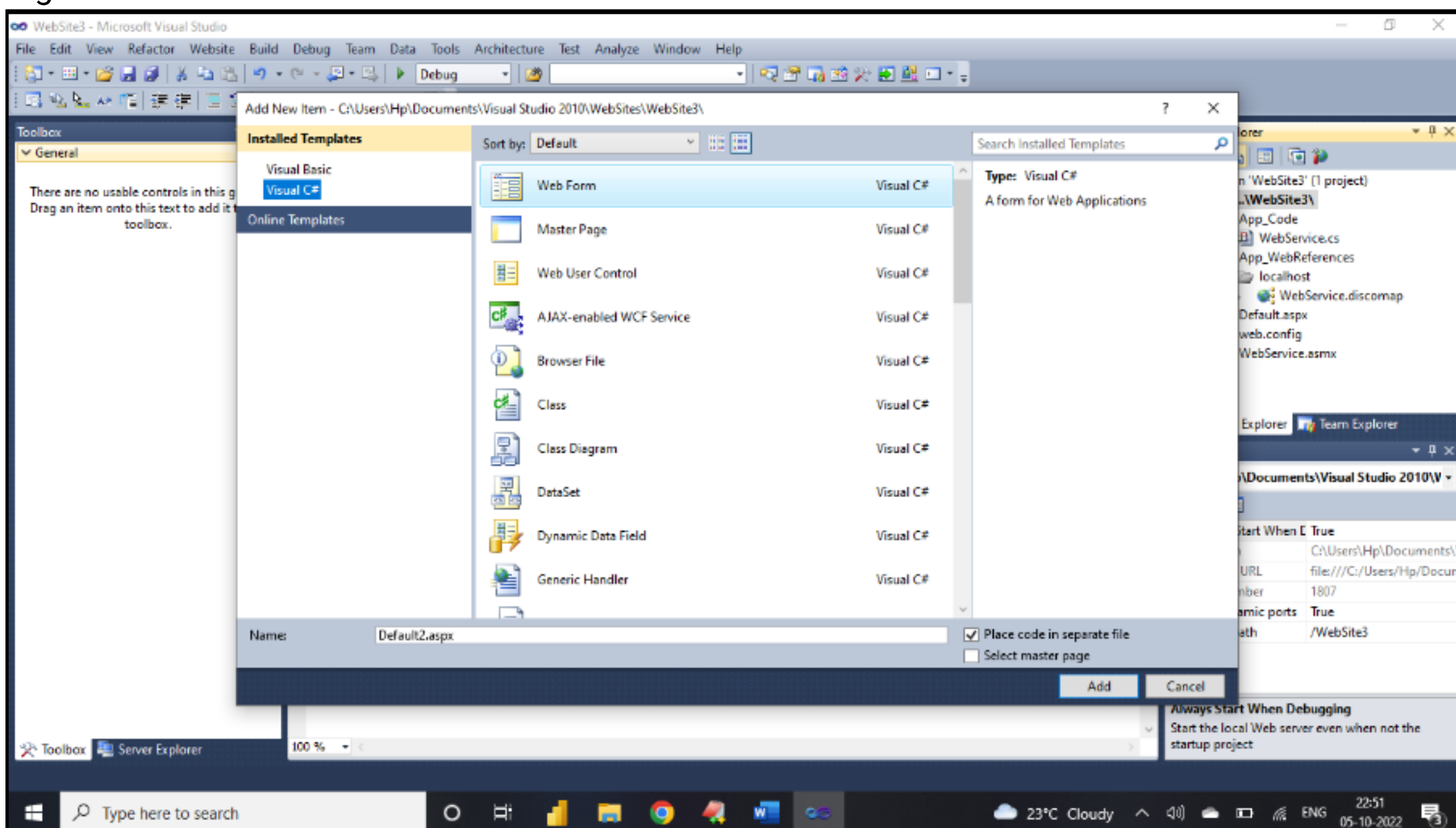
Copy URL of Web service (having extension .asmx)

Step 3: Right Click on website Menu → Add Web Reference → Paste the URL copied earlier

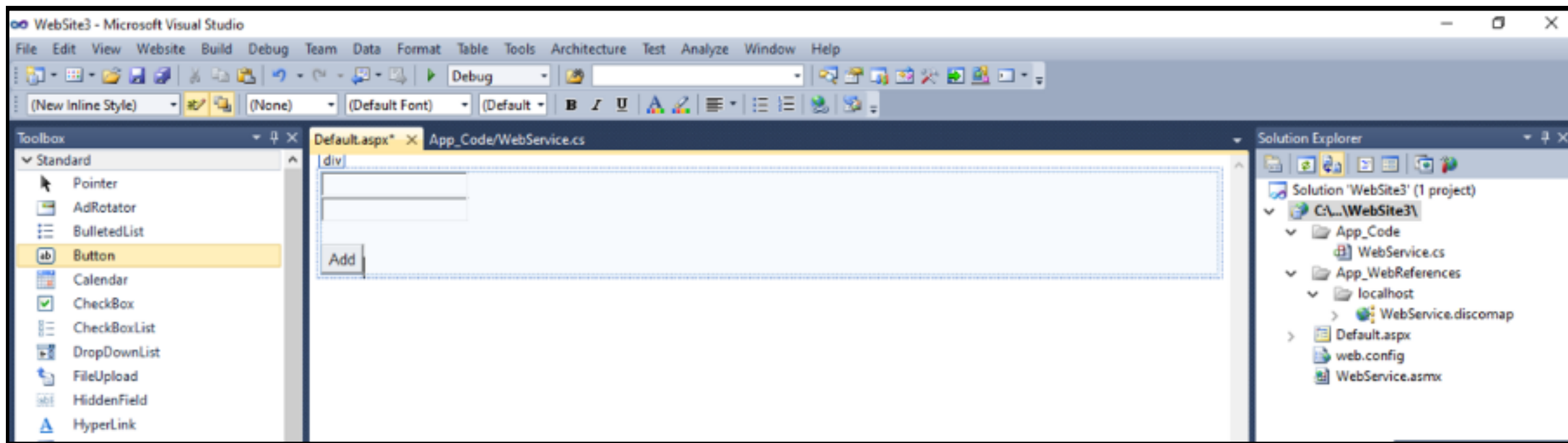


Step 4: Now create application for addition and add following code

Right Click on website Menu → Add New Item → Web Form



Now Double Click on Add Button Type This Code



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

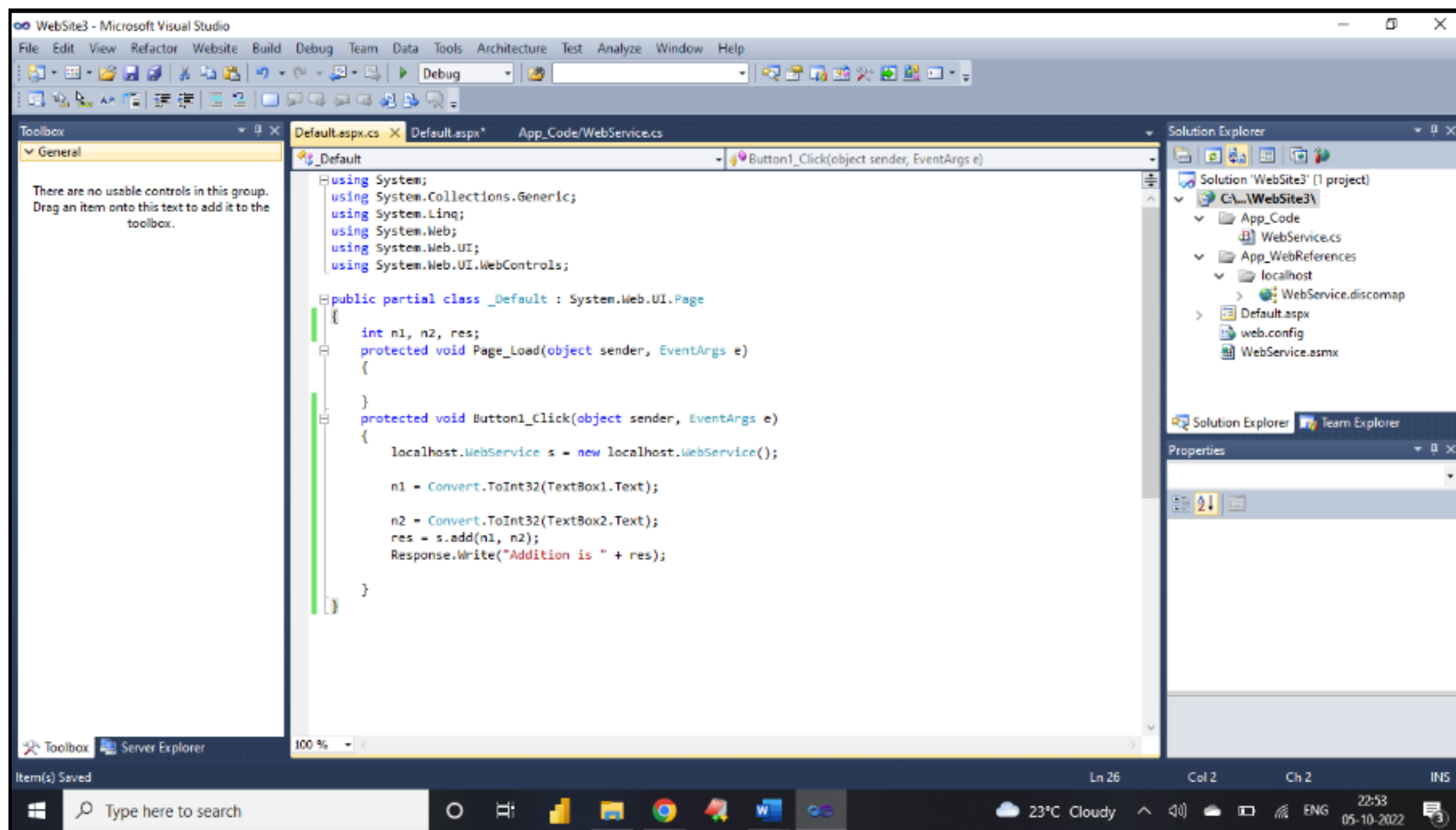
public partial class _Default : System.Web.UI.Page
{
    int n1, n2, res;
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        localhost.WebService s = new localhost.WebService();

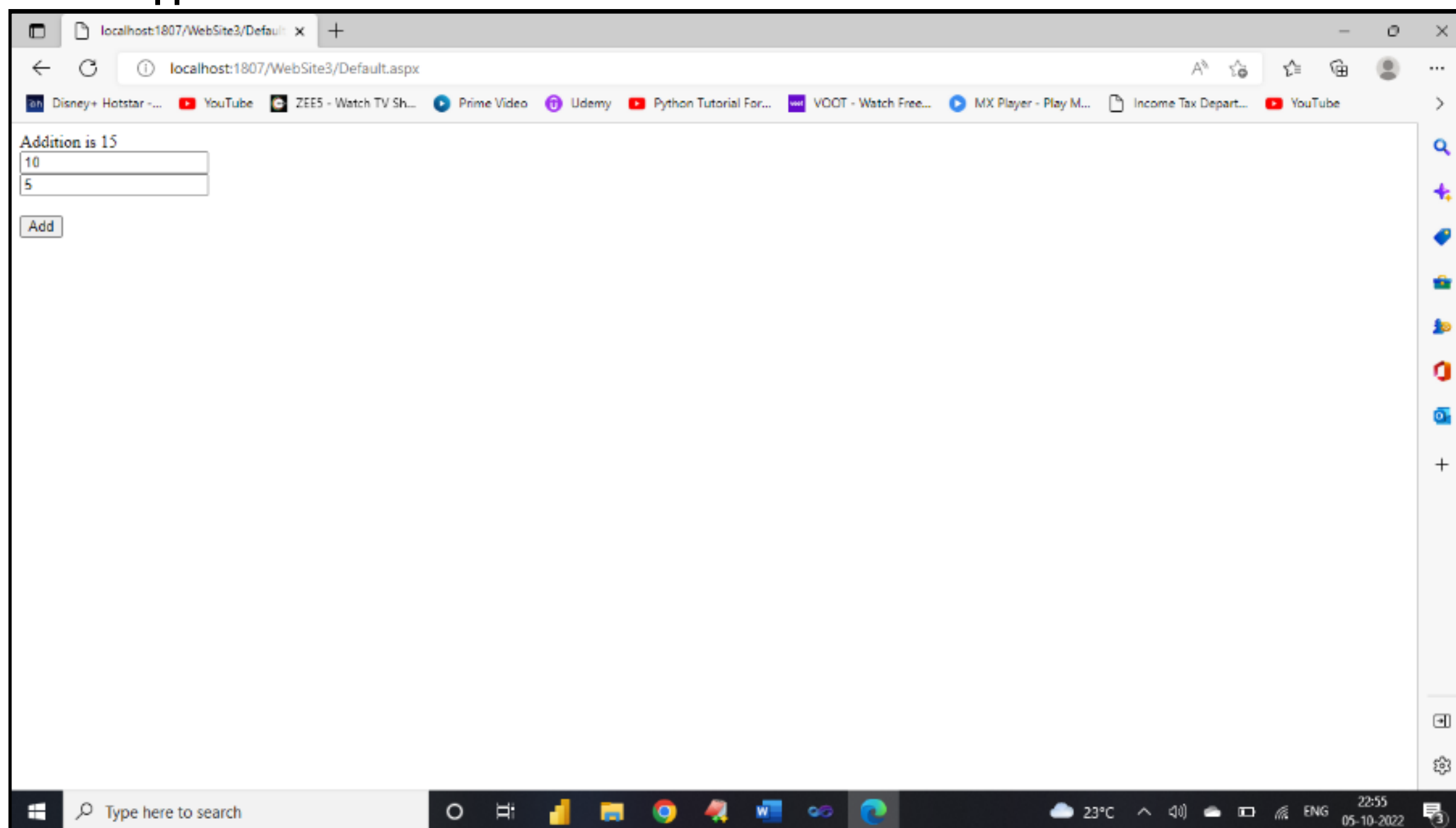
        n1 = Convert.ToInt32(TextBox1.Text);

        n2 = Convert.ToInt32(TextBox2.Text);
        res = s.add(n1, n2);
        Response.Write("Addition is " + res);

    }
}
```



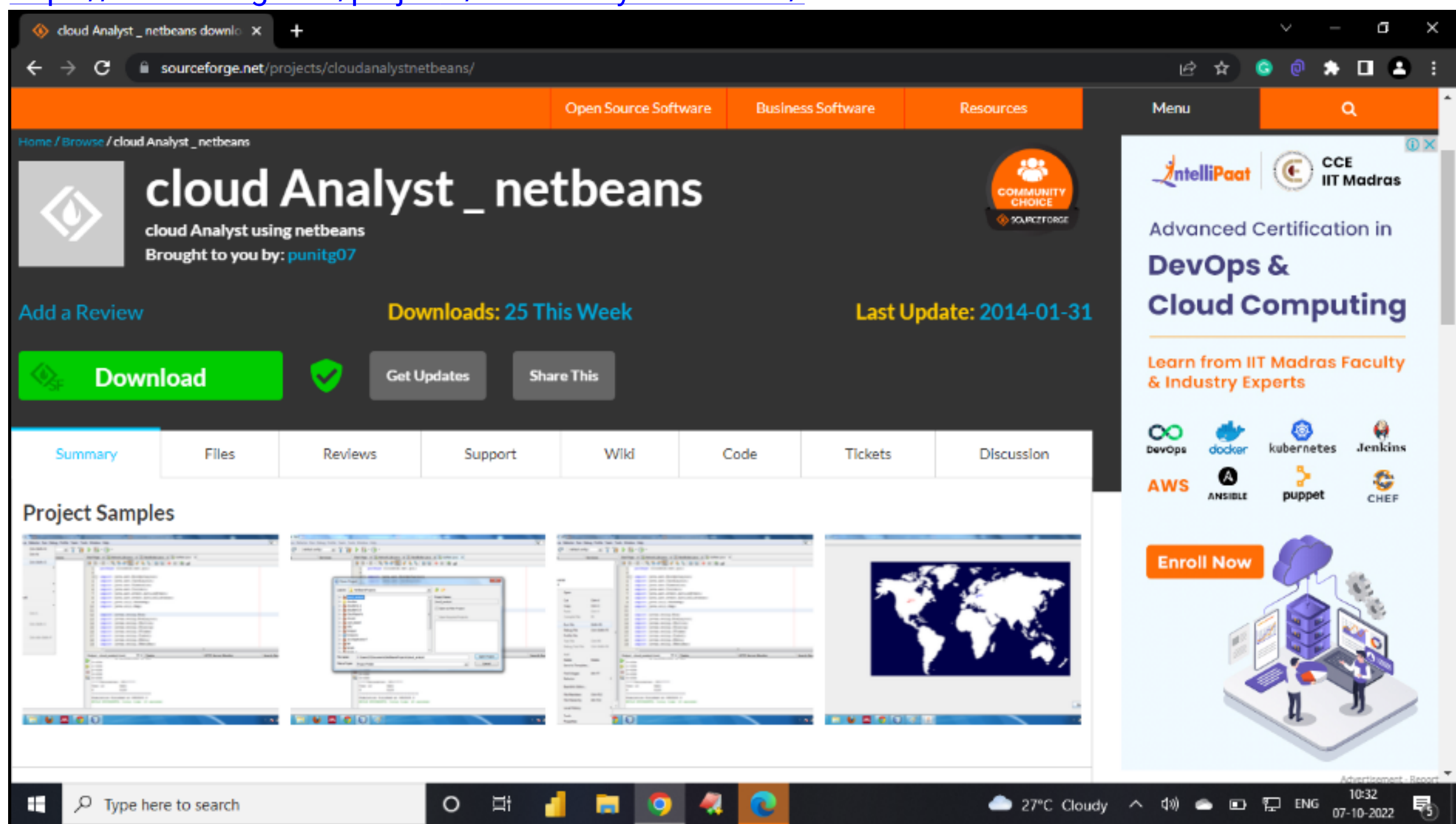
Run the Application



Assignment: Downloading and Installing Cloud Analyst

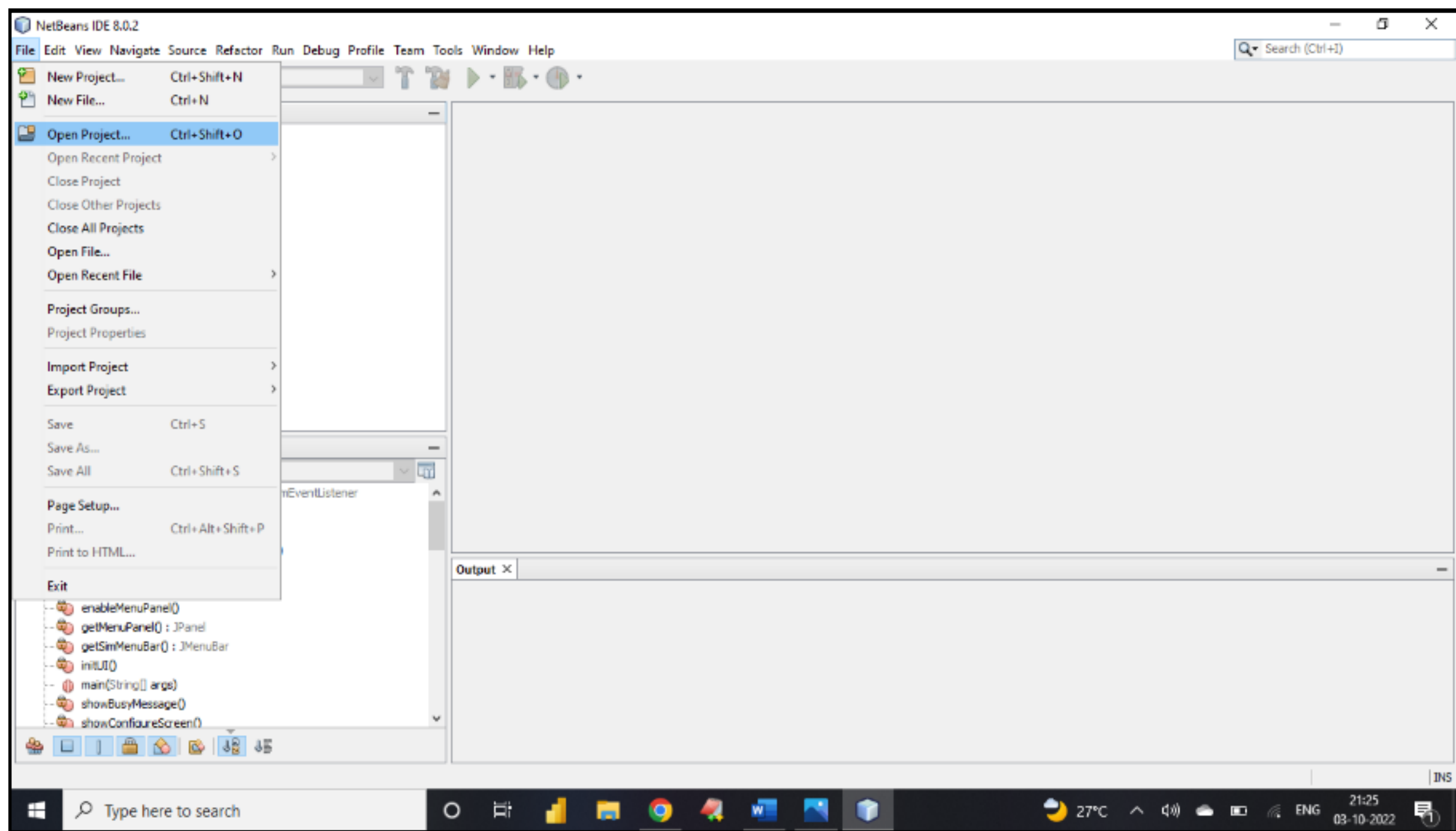
Step 1:

Download Cloud Analyst NetBeans project from the below mentioned link
<https://sourceforge.net/projects/cloudanalystnetbeans/>



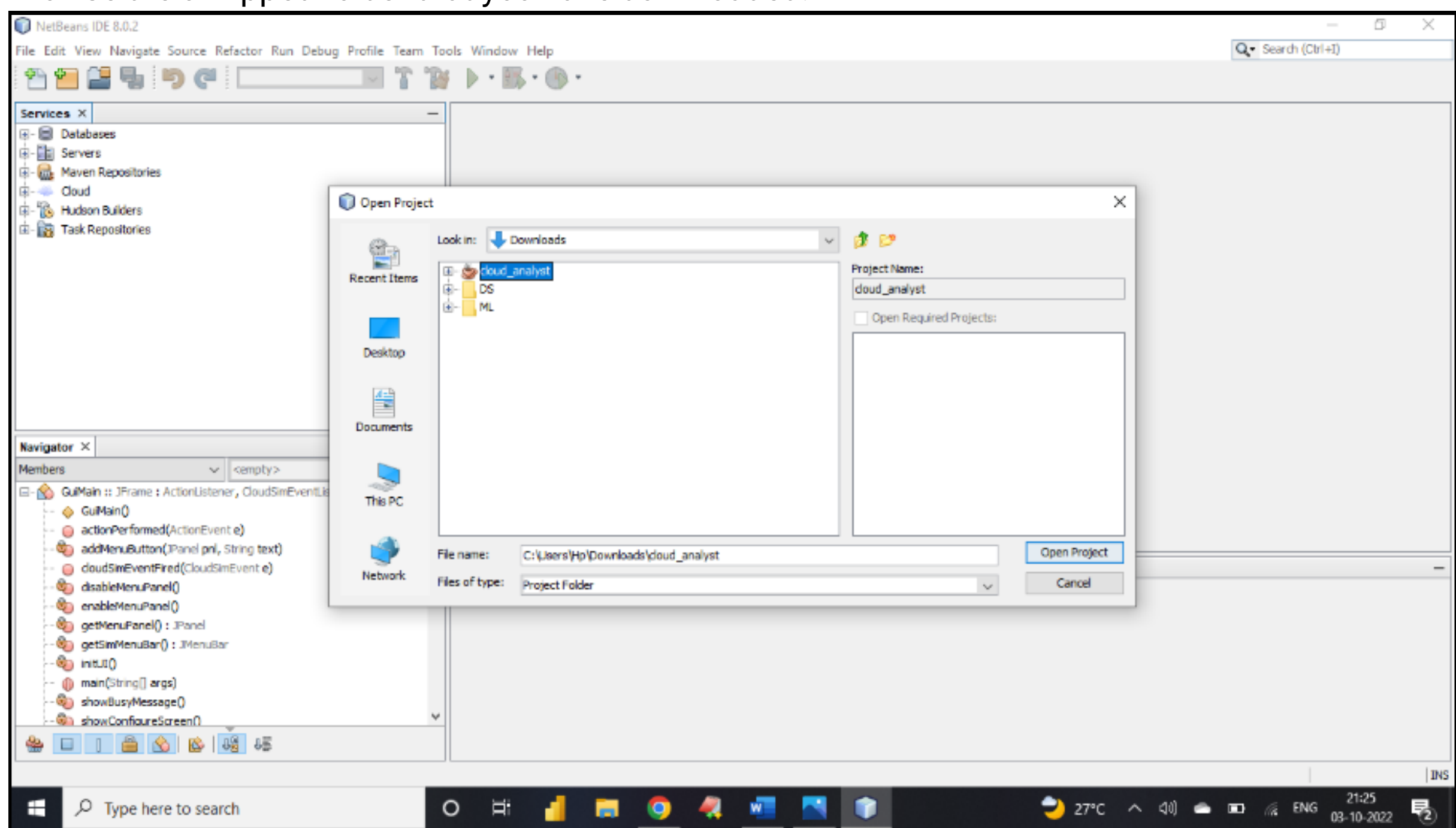
Step 2:

To run the cloud analyst got to NetBeans File → open project → browse the unzipped folder



Step 3:

Browse the unzipped folder that you have downloaded.



Step 4:

Open source package folder inside which open cloudsims.ext.gui
right click on the gui.main.java and click run

