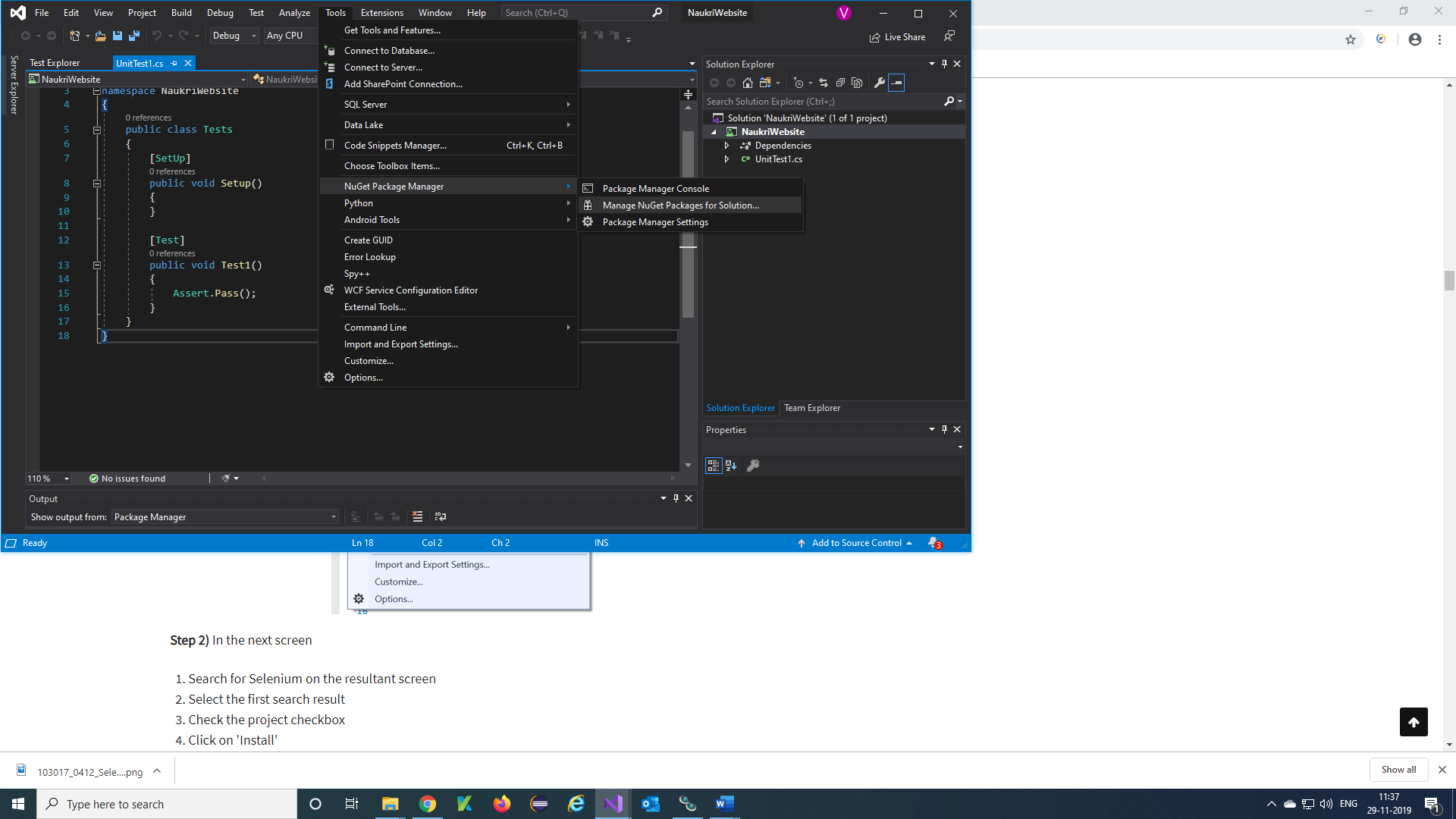
***Set Up Visual Studio with Selenium Web Driver***

***Index:***

1. ***Selenium WebDriver Download***
2. ***Nunit Plugin download***
3. ***Create a testcase***
4. ***Writing Example testcase in visual studio***
5. ***How to Build the testcase***
6. ***How to Run testcase in Visual studio***

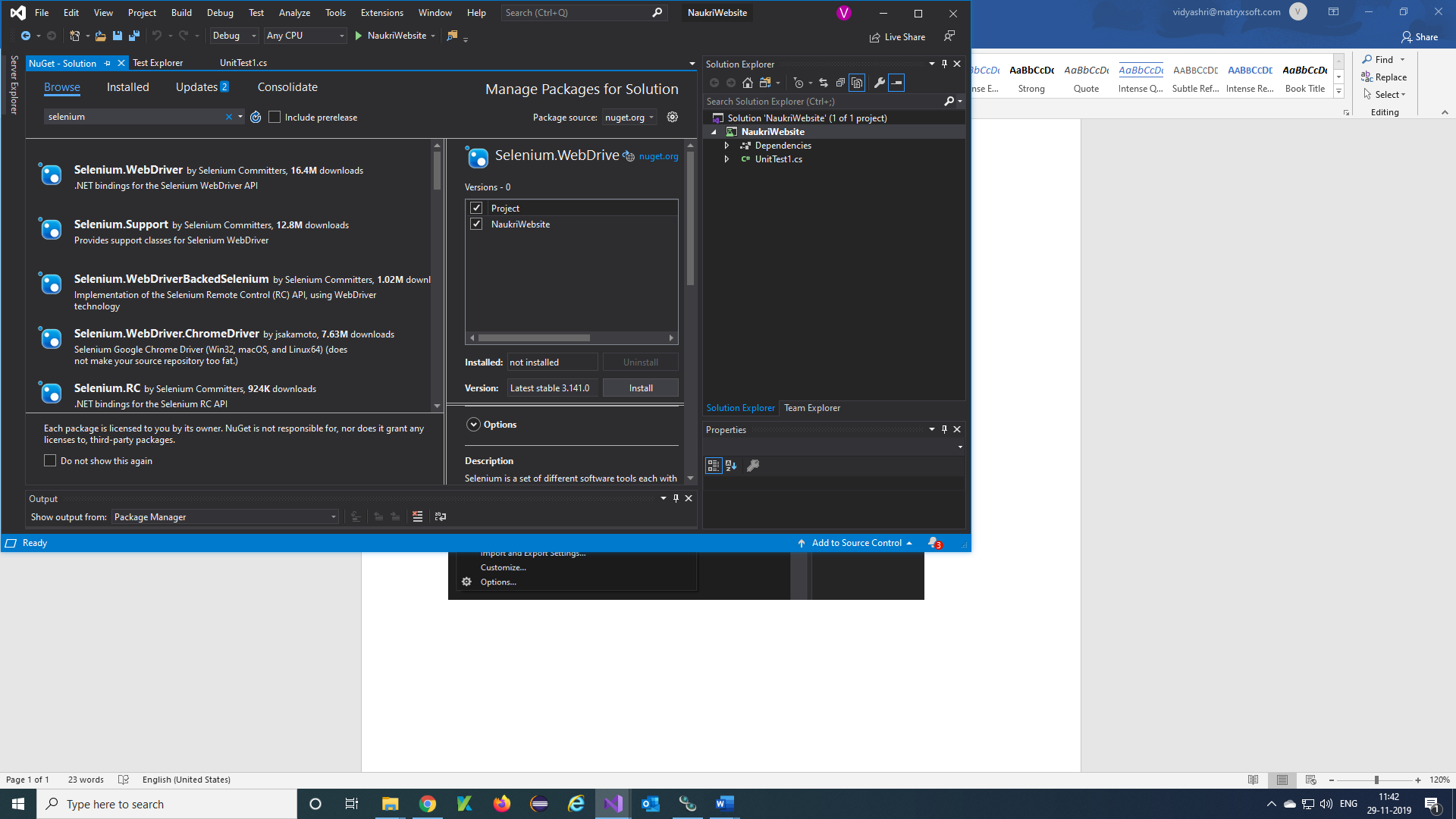
***Selenium Set Up:***

**Step 1)** Navigate to Tools -> NuGet Package Manager -> Manage NuGet Packages for Solution.

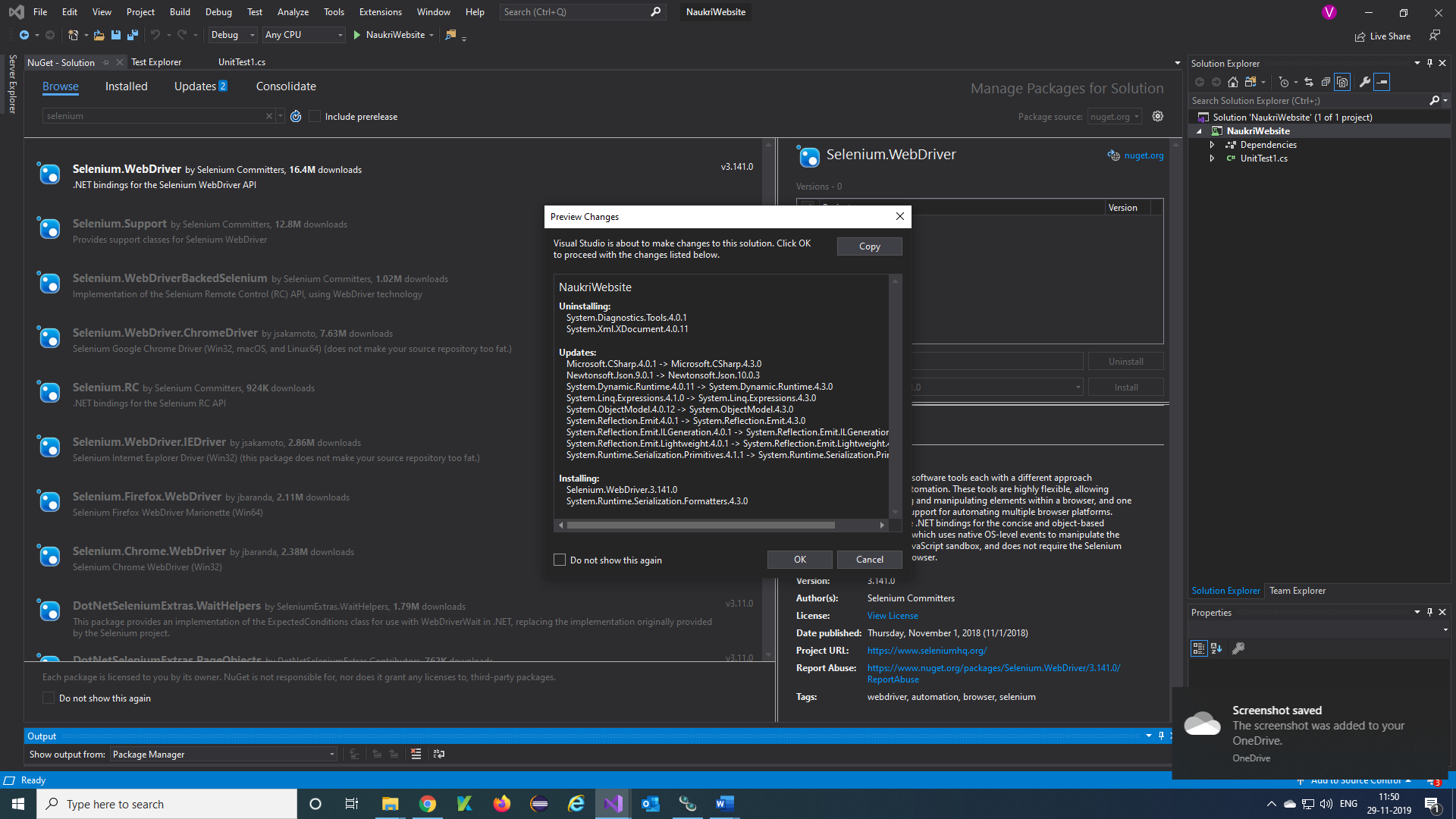


**Step 2)** NuGet-Solution Will be display

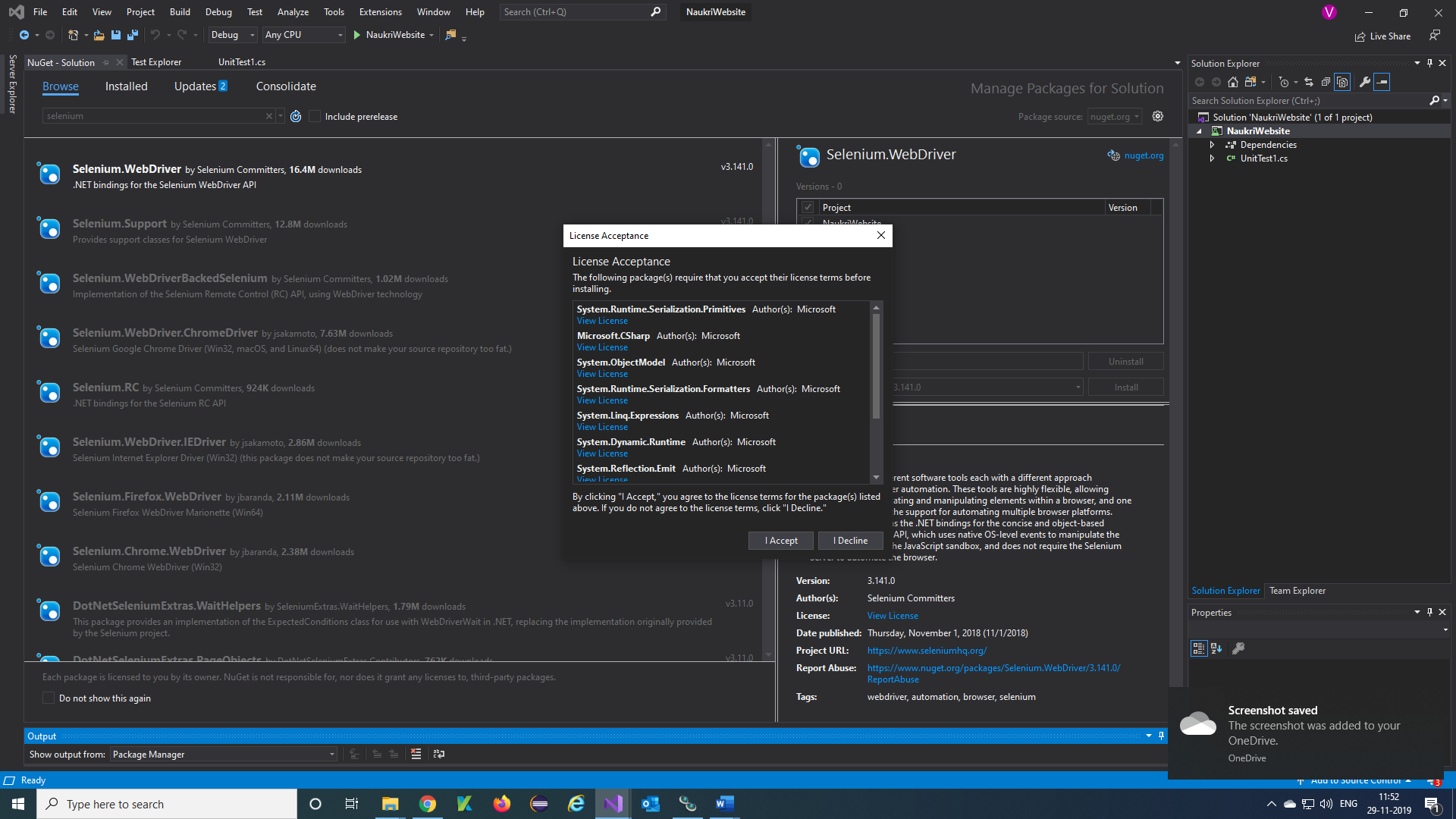
1. Select Browse and click on search and enter ‘selenium’
2. Click on Selenium.Webdriver and select your project name & click on Install



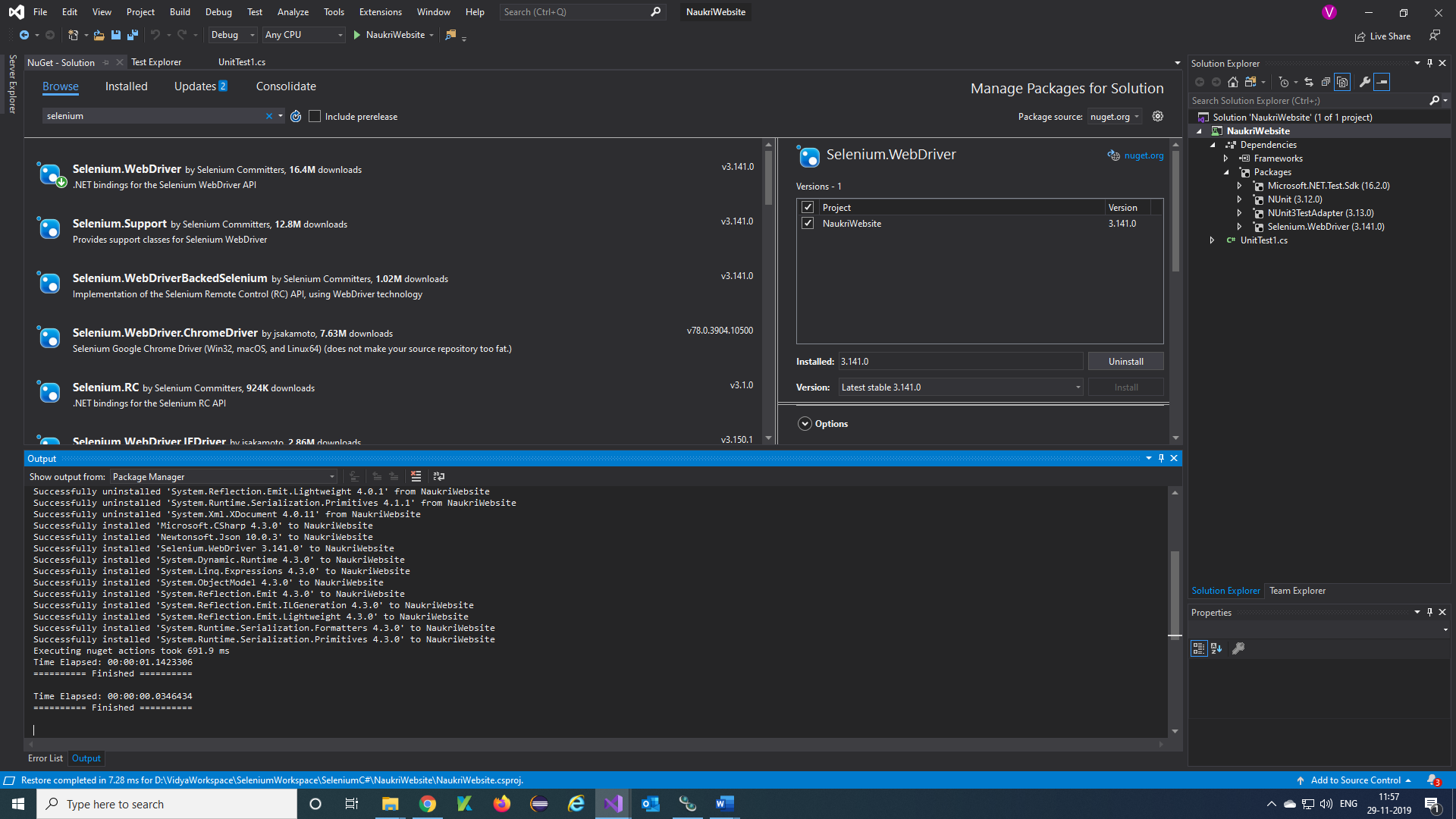
**Step 3)** Click on 'OK' button in the pop-up screen



**Step 4)** Click on ‘I Accept’ in the pop-up screen



**Step 5)** The below message will be displayed once the package is successfully installed.



Note: Follow the above steps to download Selenium.Support, Firefox Driver, Chrome Driver and Required Plugins to your project .

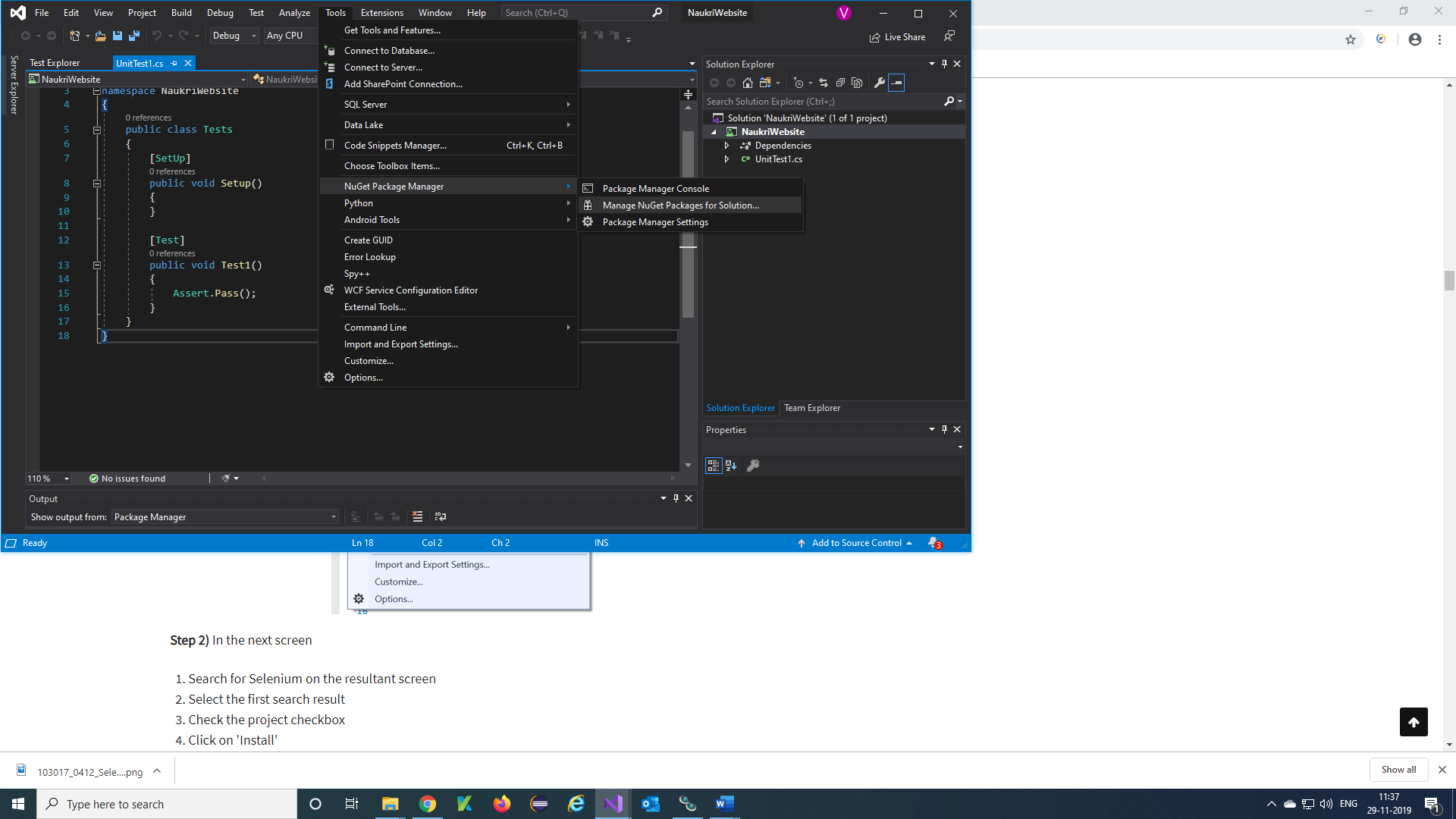
***NUnit Framework: Overview***

NUnit is the [Unit Testing](https://www.guru99.com/unit-testing-guide.html) framework supported by Visual Studio and Selenium WebDriver. NUnit is the most widely used Unit Testing framework for .Net applications. NUnit presents the test results in a readable format and allows a tester to debug the automated tests.

We need to install NUnit Framework and NUnit Test Adapter onto Visual Studio inorder to use it.

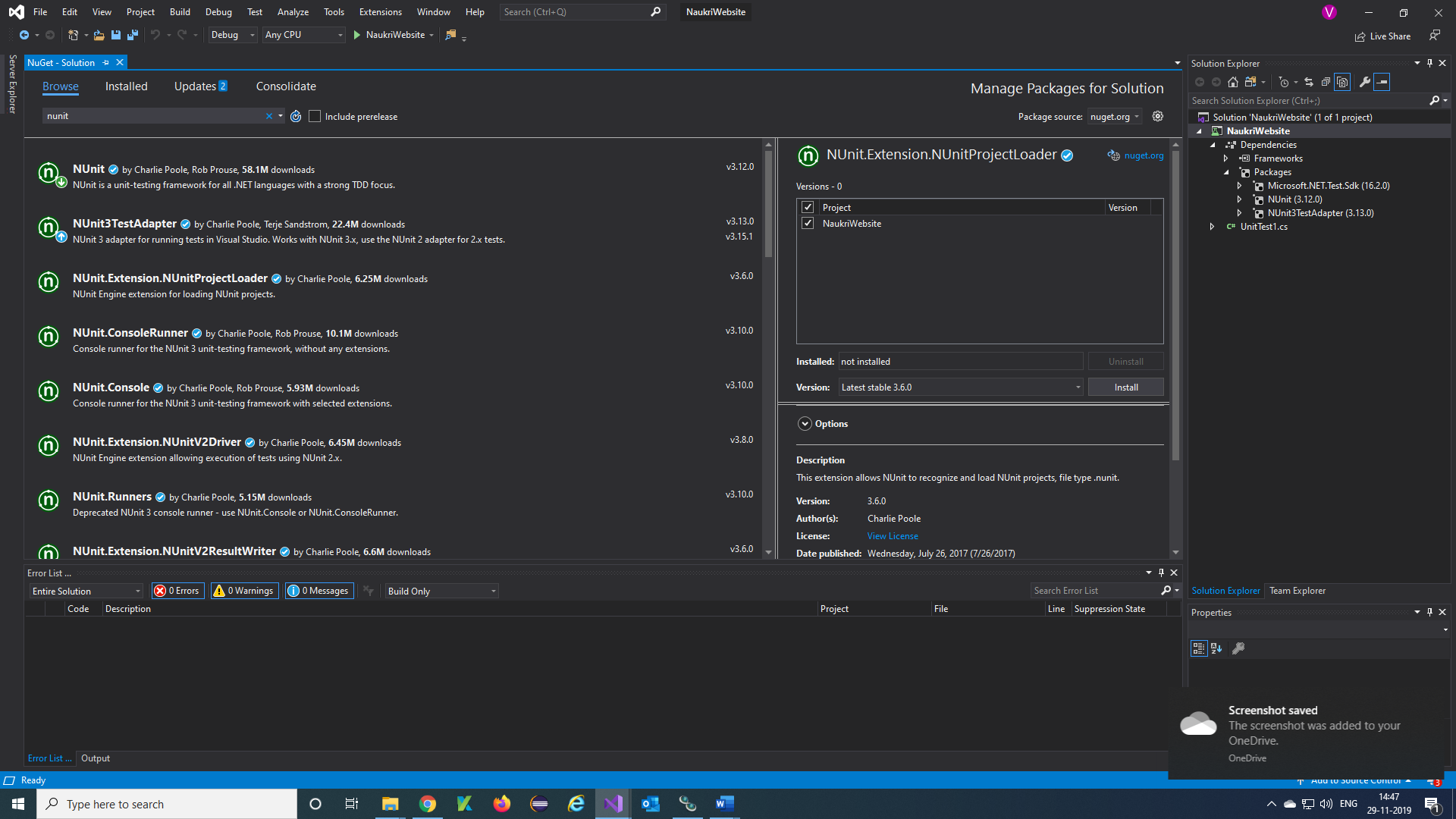
***Steps to install NUnit Framework:***

**Step 1)** Navigate to Tools -> NuGet Package Manager -> Manage NuGet Packages for Solution

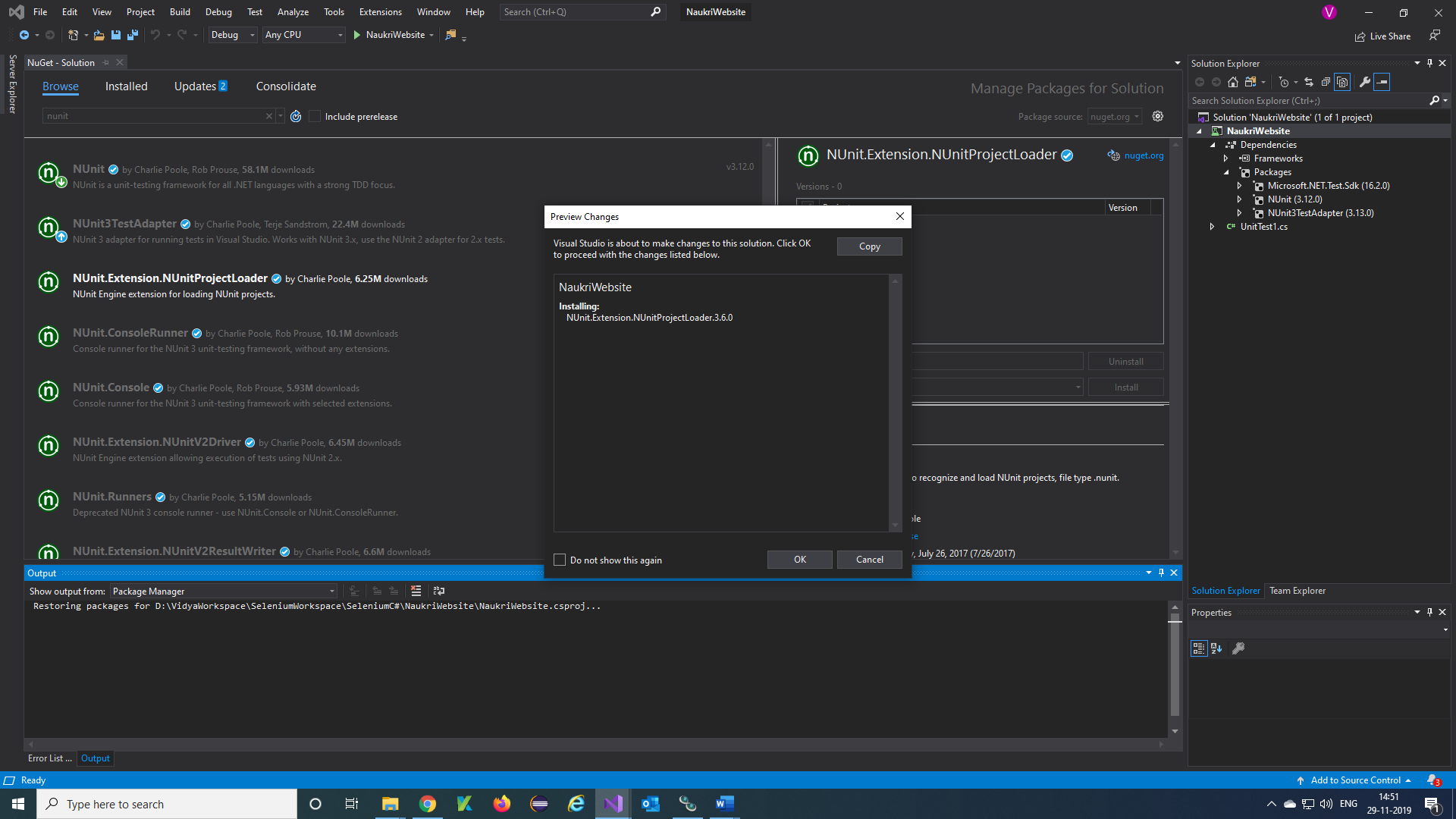


**Step 2)** NuGet-Solution Will be display

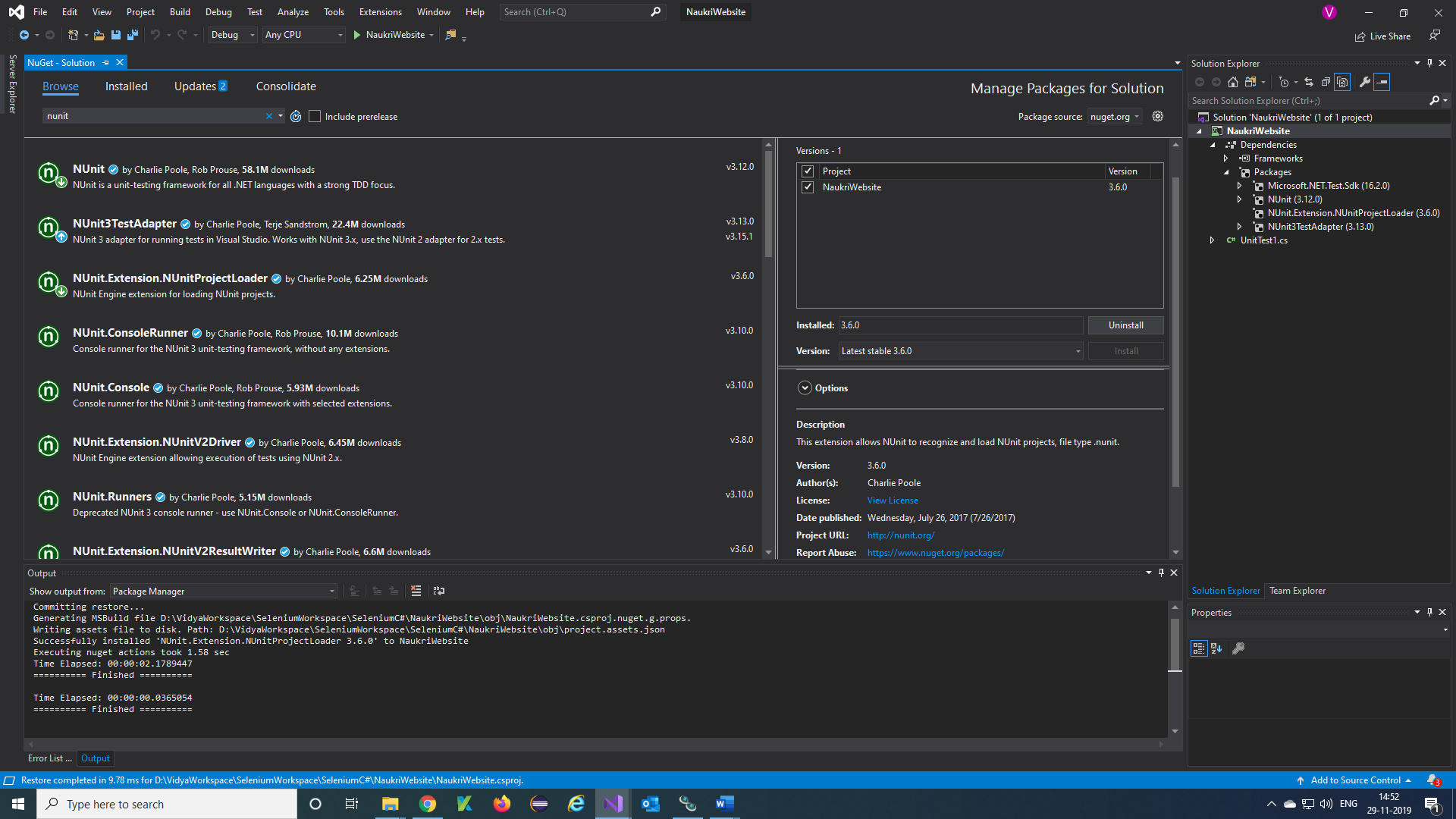
1. Select Browse and click on search and enter ‘NUnit
2. Click on NUnit and select your project name & click on Install



**Step 3)** Click on 'OK' button in the pop-up screen



**Step 4)** The below message will be displayed once the package is successfully installed.



**Note**: Follow the same above steps to download NUnit3TestAdapter package.

***Selenium and NUnit Framework:***

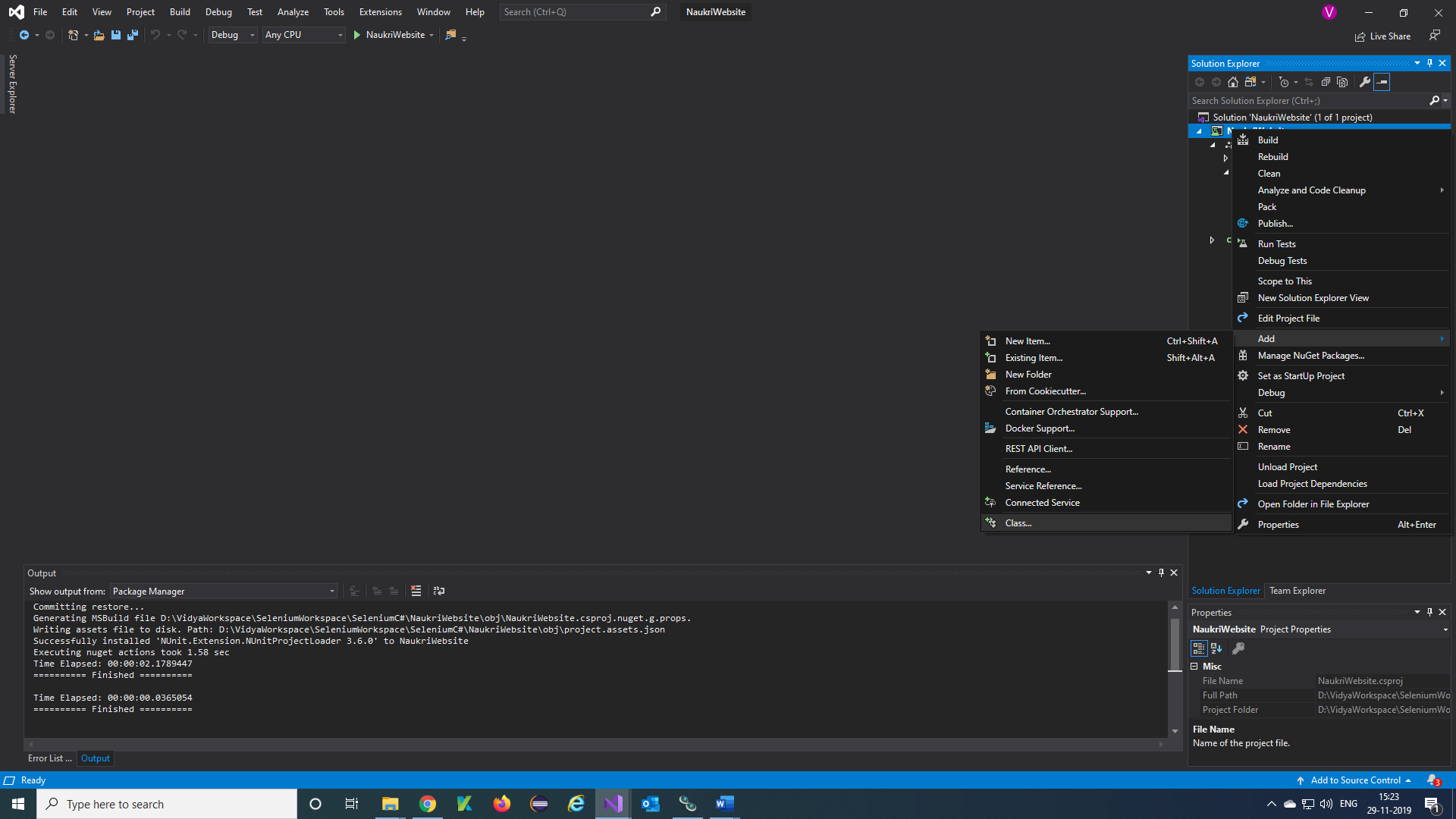
Integration of selenium with NUnit framework allows a tester to differentiate between various test classes. NUnit also allows testers to use annotations such as SetUp, Test, and TearDown to perform actions before and after running the test.

NUnit framework can be integrated with Selenium by creating a NUnit test class and running the test class using NUnit framework.

The below are the steps needed to create and run a test class using NUnit framework.

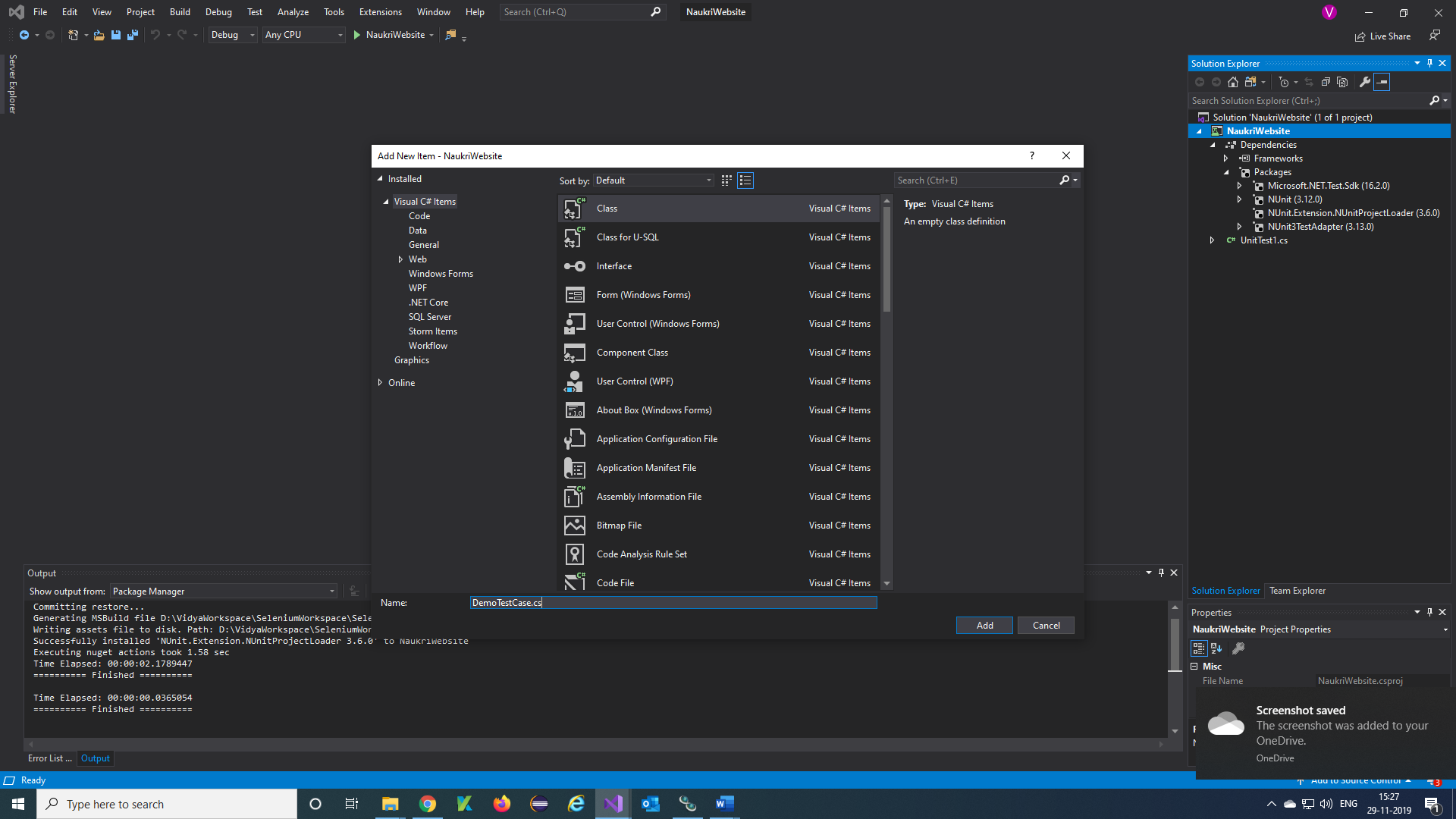
***Steps to create a NUnit Test class in Selenium:***

**Step 1)** In the Solution Explorer, Right click on project > Add > Class

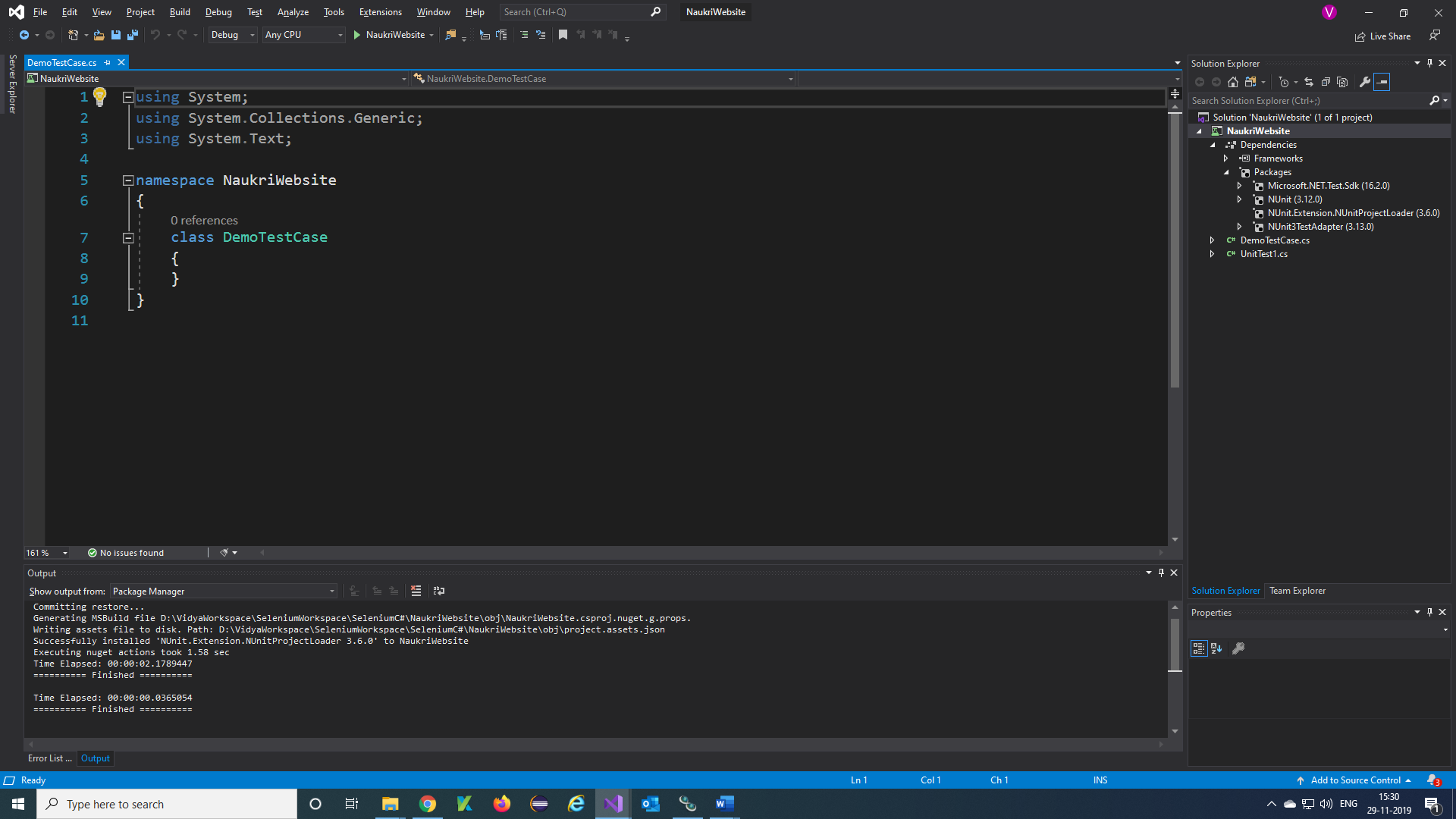


**Step 2)**Class creation window will appear.

1. Provide a name to the class
2. Click on Add button



**Step 3)** The below screen will appear.



**Step 4)** Add the following code to the created class. Please note that you need to download the Chrome/Firefox drivers to your project to chrome/Firefox driver initialization.

using NUnit.Framework;

using OpenQA.Selenium;

using OpenQA.Selenium.Firefox;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

namespace NaukriWebsite

{

class DemoTestCase

{

[Test]

public void TestMethod()

{

IWebDriver driver = new FirefoxDriver();

//Go to Google

driver.Navigate().GoToUrl("https://www.naukri.com");

driver.Manage().Window.Maximize();

driver.FindElement(By.Id("login\_Layer")).Click();

Console.WriteLine("Login Page shall be display");

//Login to Naukri website

driver.FindElement(By.Name("email")).SendKeys("vidyaclk12@gmail.com");

driver.FindElement(By.Name("PASSWORD")).SendKeys("vidya123");

driver.FindElement(By.XPath("//button[contains(text(),'Login')]")).Click();

Thread.Sleep(6000);

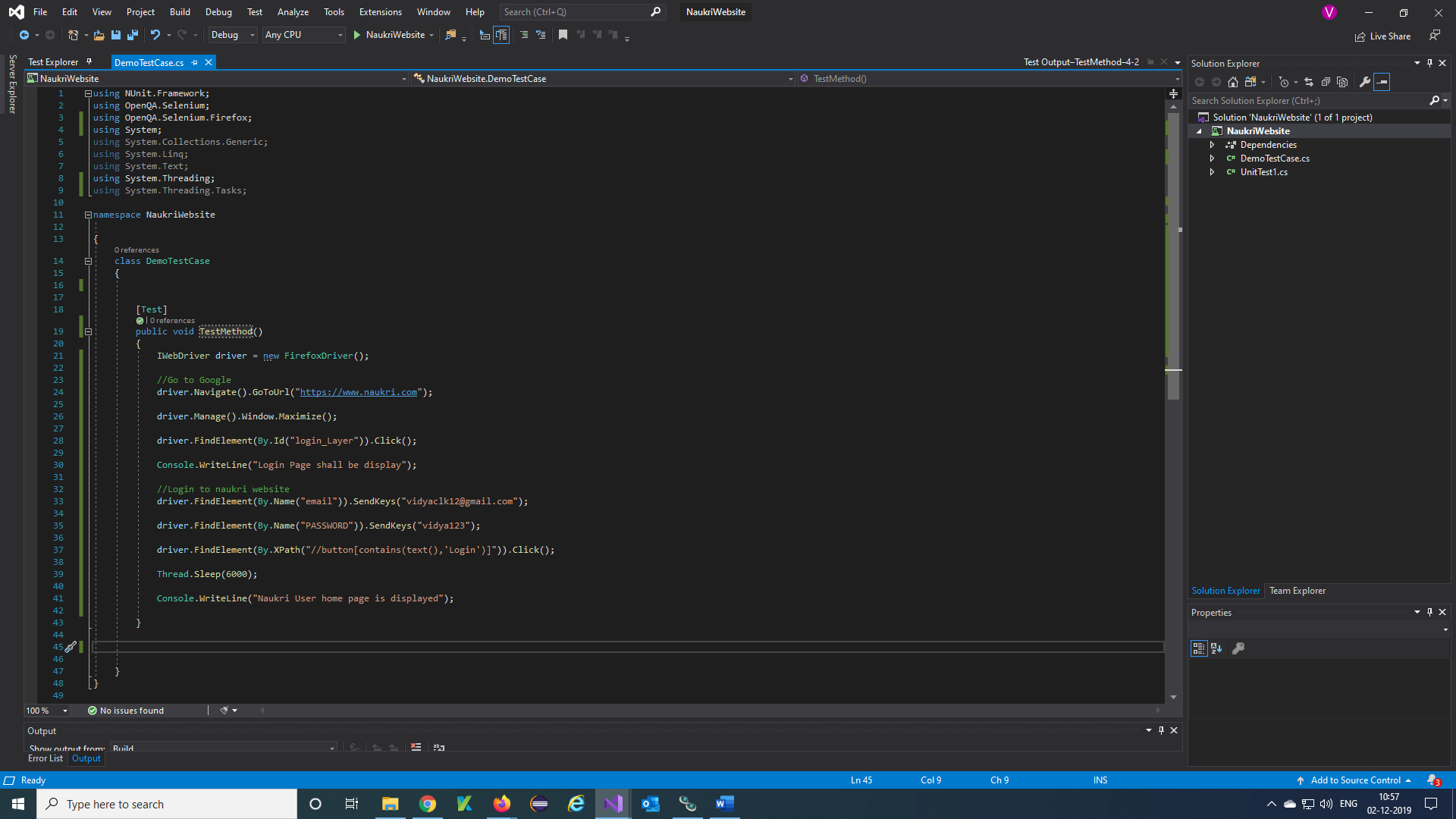
Console.WriteLine("Naukri User home page is displayed");

}

}

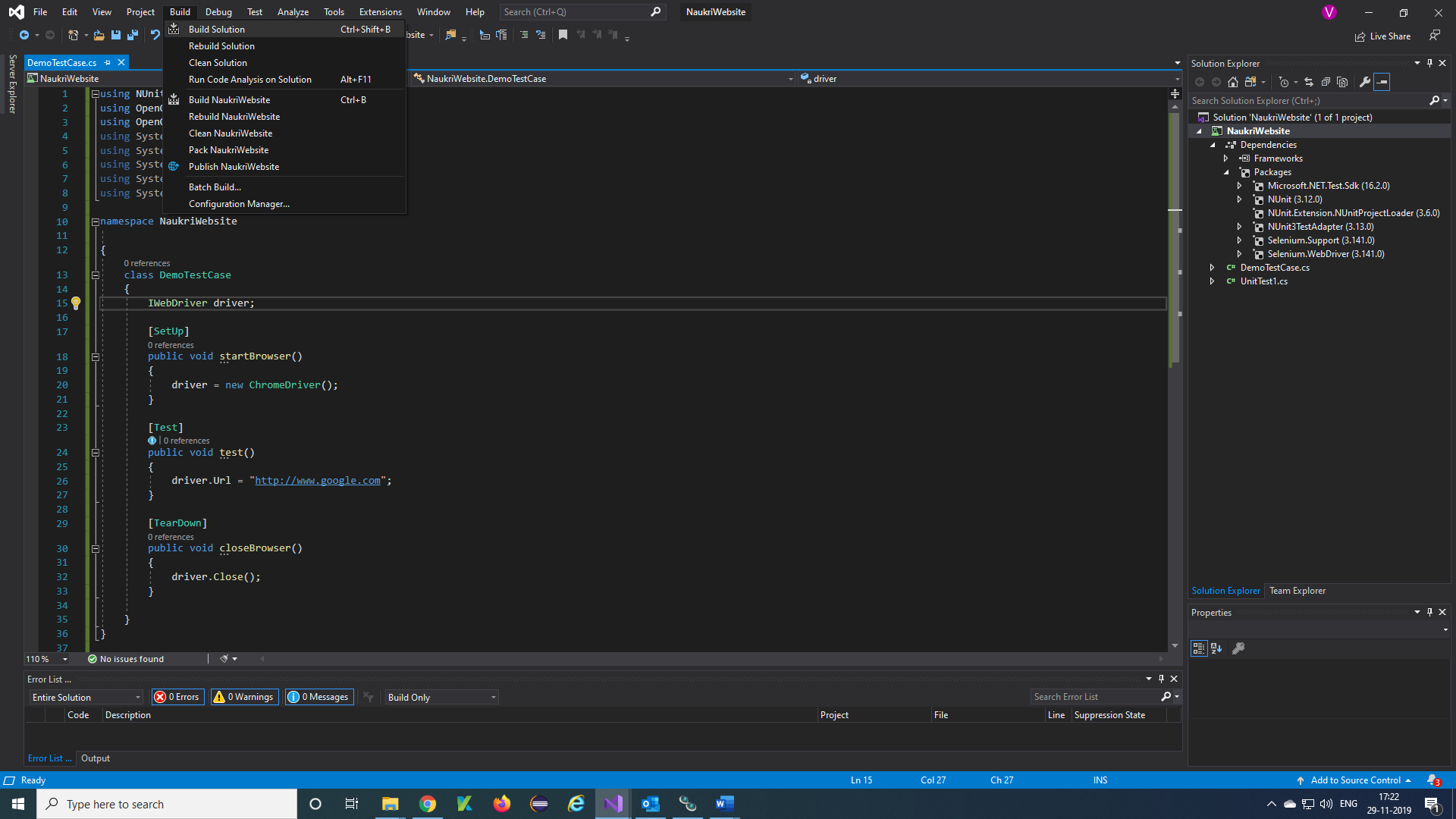
}

**Step 5)** The below screen will appear.

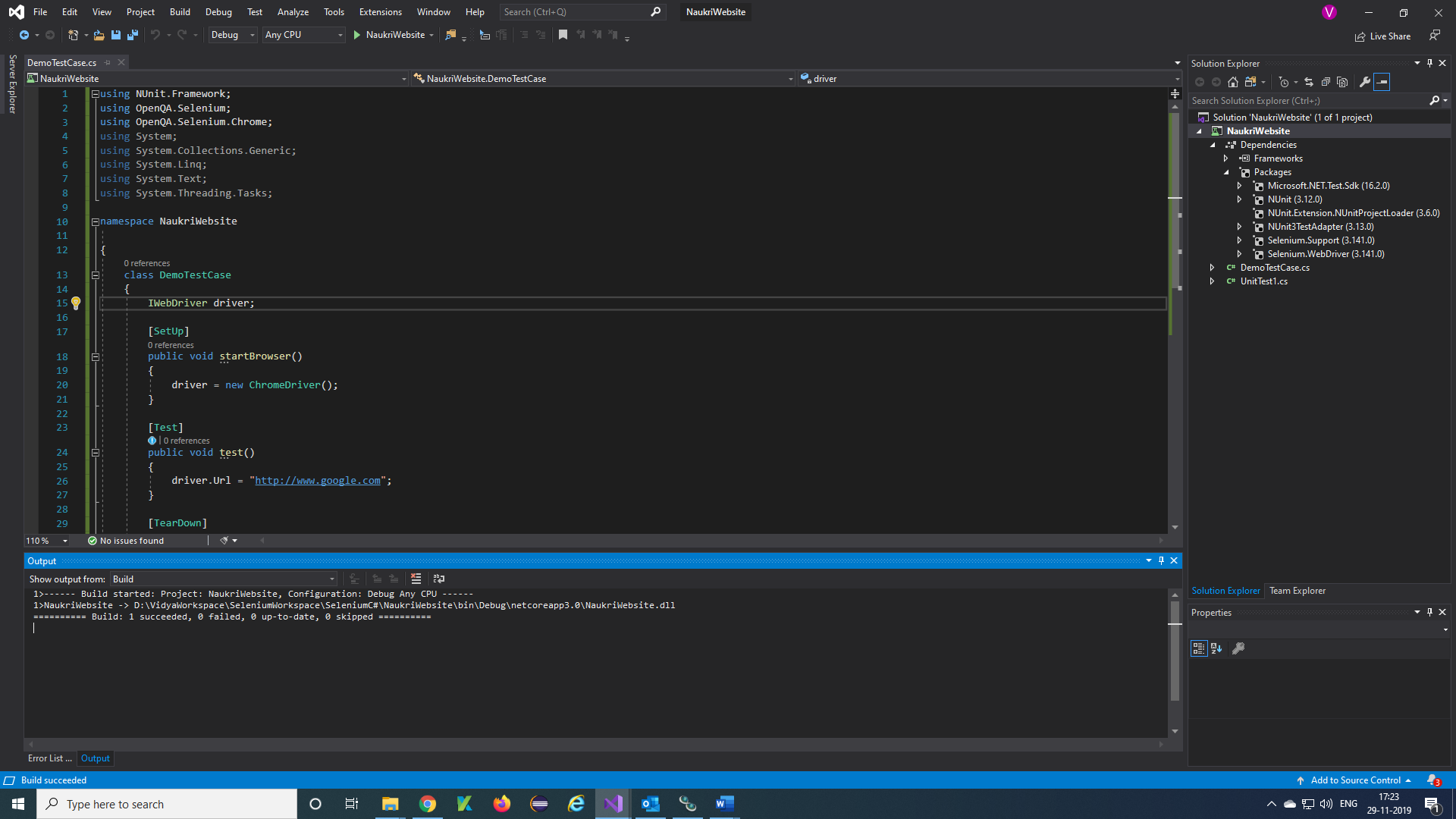


***Run the testcases:***

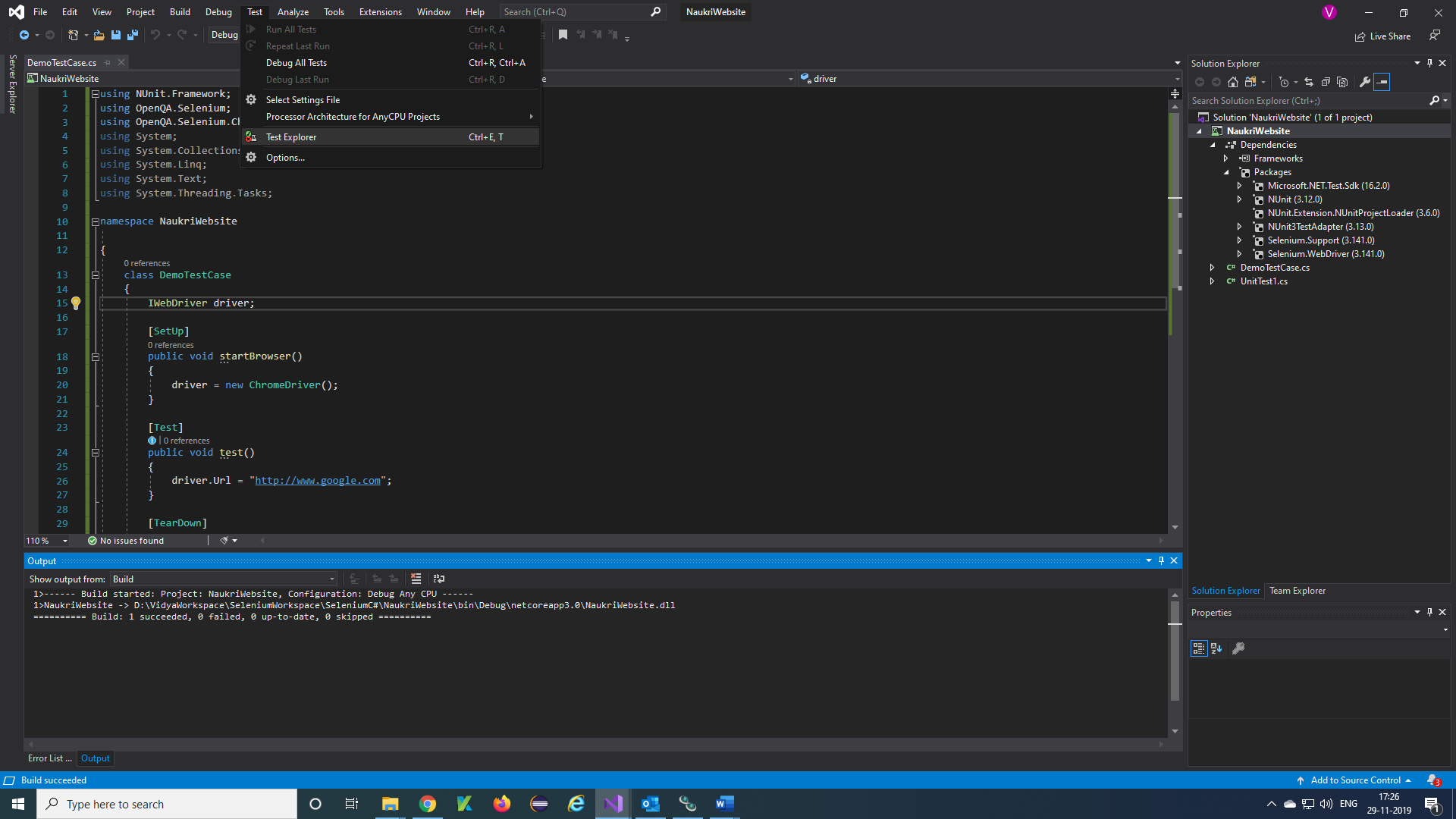
**Step 1)** Click on 'Build' -> 'Build Solution'



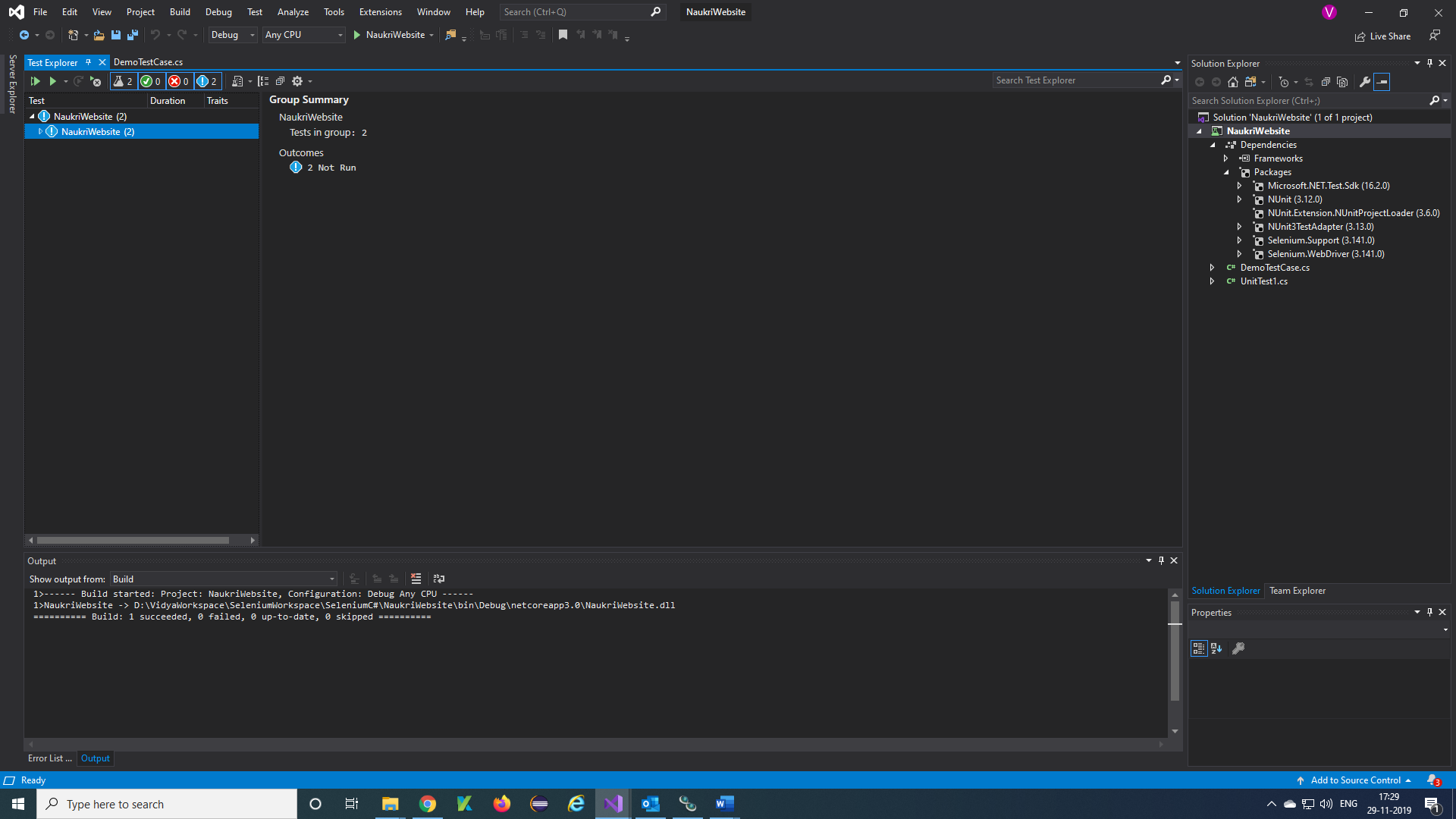
**Step 2)** The build is successful



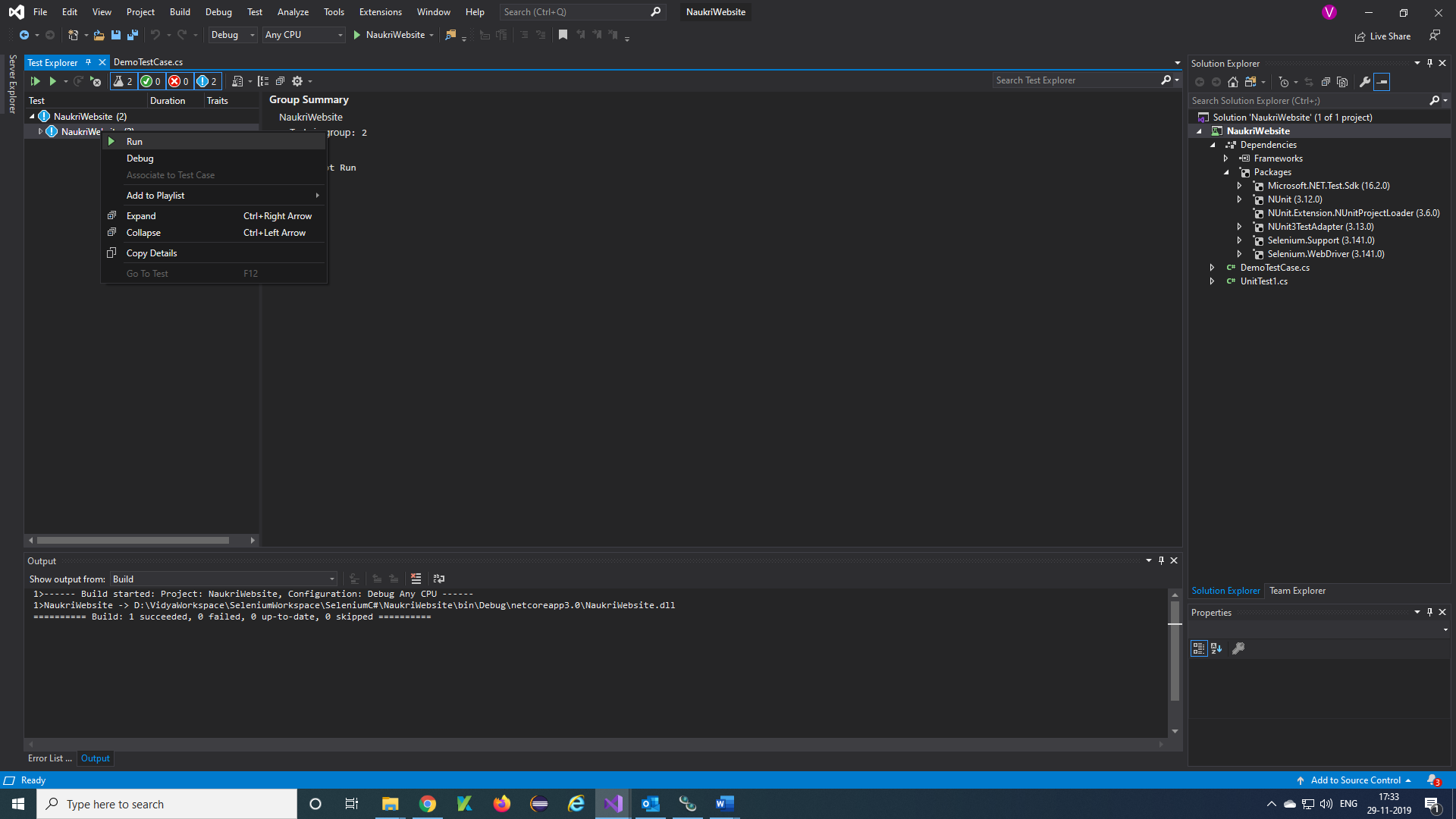
**Step 3)** Once the build is successful, we need to open the Test Explorer window. Click on Test -> Windows -> Test Explorer



**Step 4)** Select your testcase

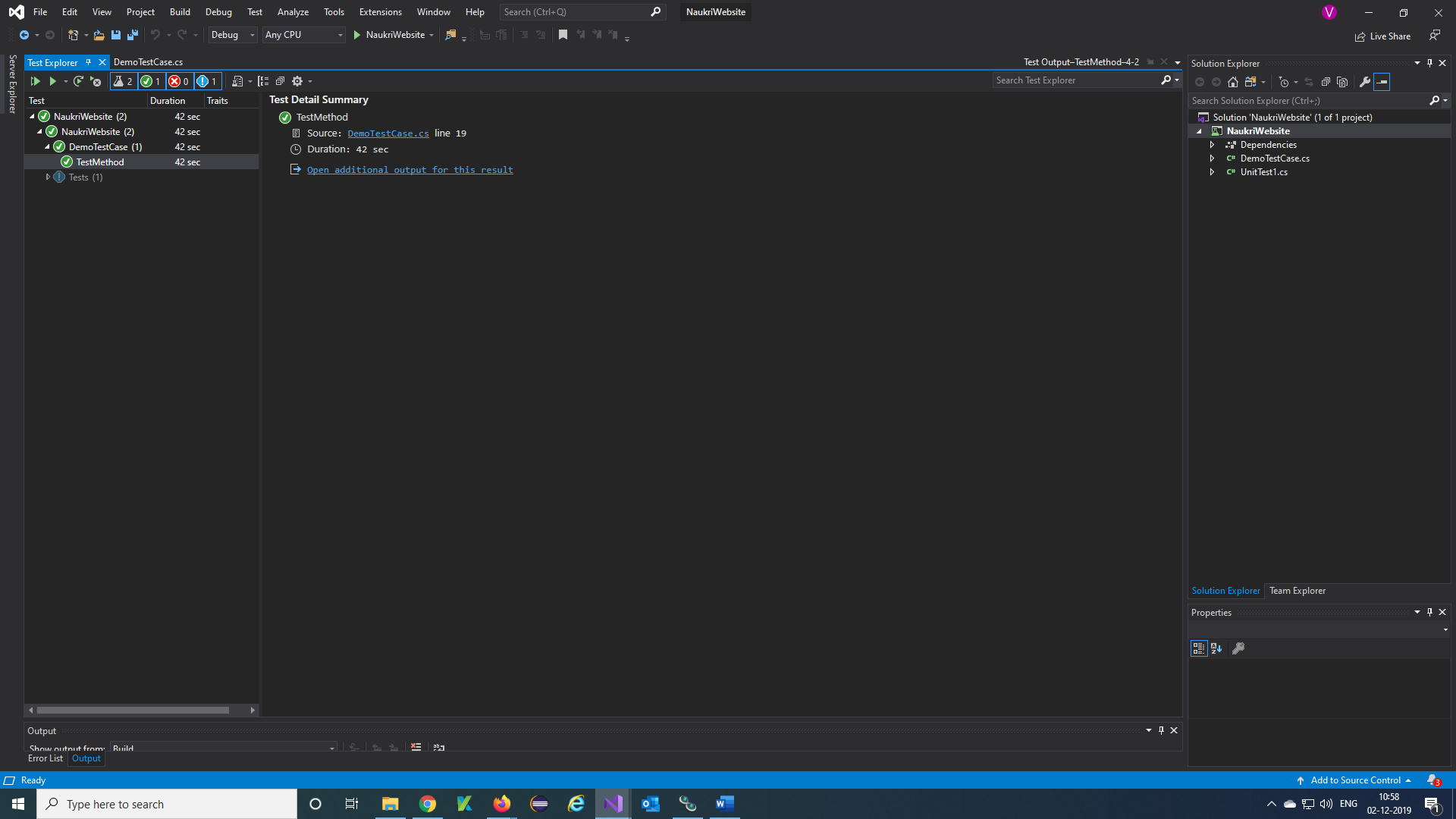


**Step 5)** Right click on your testcase or test method and click on Run.



**Note**: Here you can run particular test and also, we can give Run All

**Step 6)** Once the testcase ran successfully this will appear and also, we can check here our console output result.



**Step 7)** This is console output result of our testcase.

