<https://www.c-sharpcorner.com/UploadFile/a20beb/unit-testing-in-net/>

Testing : Check whether application is working properly or not

As per the users requirements

Different type of Testing

Static Testing : Review of code

Dynamic Testing : Executing the code

Different levels

1. Unit Testing : Done by Developer , we test single unit (method)

2. Integration Testing : Developer , where we combine all the methods

3. System Testing : Testing complete system : Development Team

4. User Acceptance System : End user

**White Box Testing** : where the tester knows about the internal part of the system (Developer)

**Black Box Testing** : where the tester knows nothing about the internal part of the system (End User)

**Unit Testing Techniques:**

* **Black Box Testing** - Using which the user interface, **input** and output are tested.
* **White Box Testing** - used to test each one of those functions behaviour is tested.
* Gray Box Testing - Used to execute tests, risks and assessment methods.

**Arrange**/**Act**/**Assert** (AAA) is a pattern for **arranging** and formatting code in **Unit Test** methods. ... The idea is to develop a **unit test** by following these 3 simple steps: **Arrange** – setup the **testing** objects and prepare the prerequisites for your **test**. **Act** – perform the actual work of the **test**. **Assert** – verify the result.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace UnitTestProjectDemo

{

public partial class WebForm1 : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void Button1\_Click(object sender, EventArgs e)

{

TextBox3.Text = (add(Convert.ToByte(TextBox1.Text) ,Convert.ToByte(TextBox2.Text))).ToString();

}

public int add(int x, int y)

{

return (x + y);

}

}

}

using System;

using Microsoft.VisualStudio.TestTools.UnitTesting;

using UnitTestProjectDemo;

using System.Web;

namespace TestProject

{

[TestClass]

public class UnitTest1

{

[TestMethod]

public void TestMethod1()

{

// Test has three steps

// Arrange

WebForm1 webForm1 = new WebForm1();

// Act

int result = webForm1.add(10, 20);

// Assert , Verify the output

Assert.AreEqual(30, result); }

}

}