**WEEK 2**

**NUnit and Moq**

**-Syed Ateeb Ul Hasan**

**Superset ID: 6362359**

**Exercise – 1 (NUnit)**

**Code -   
File Name – CalculatorTests.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using NUnit.Framework;

using CalcLibrary;

namespace CalcLibrary.Tests

{

[TestFixture]

public class CalculatorTests

{

private Calculator calculator;

[SetUp]

public void Setup()

{

calculator = new Calculator();

}

[TearDown]

public void Teardown()

{

calculator = null;

}

[TestCase(7, 2, 4)]

[TestCase(0, 2, -1)]

[TestCase(0, 0, 0)]

[TestCase(1, 7, 3)]

[TestCase(0, 3, -1)]

[TestCase(10, 4, 0)]

[TestCase(1, 2, 8)]

[TestCase(5, 2, -2)]

[TestCase(8, 0, -4)]

[TestCase(6, 8, 1)]

[TestCase(9, 2, -4)]

[TestCase(3, 1, 0)]

[TestCase(7, 6, 1)]

[TestCase(8, 0, -2)]

[TestCase(5, 7, 8)]

[TestCase(6, 0, 2)]

[TestCase(1, 2, -1)]

[TestCase(3, 6, 0)]]

public void Add\_WhenCalled\_ReturnsCorrectSum(int a, int b, int expected)

{

int result = calculator.Add(a, b);

Assert.That(result, Is.EqualTo(expected));

}

}

}

**File Name – Calculator.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace CalcLibrary

{

public class Calculator

{

public int Add(int a, int b)

{

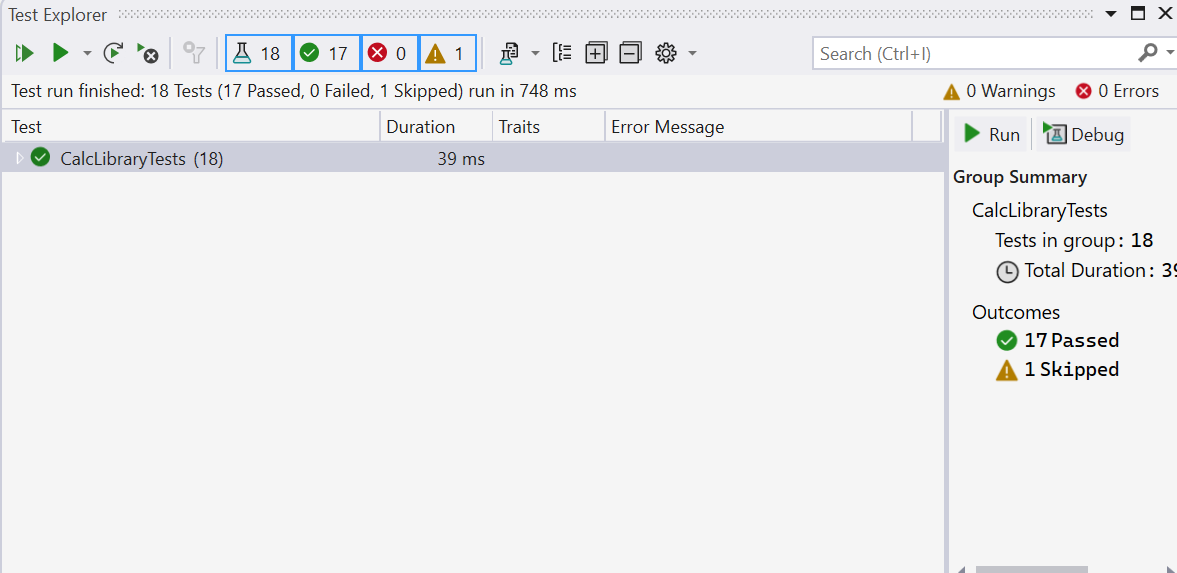
return a + b;

}

}

}

**Output –**

****

**Exercise – 1 (Moq)**

**Code –**

**Under Folder - CustomerCommLib**

**File Name – CustomerComm.cs**

namespace CustomerCommLib

{

public class CustomerComm

{

IMailSender \_mailSender;

public CustomerComm(IMailSender mailSender)

{

\_mailSender = mailSender;

}

public bool SendMailToCustomer()

{

// Imagine real customer email and message

return \_mailSender.SendMail("cust123@abc.com", "Some Message");

}

}

}

**File Name – MailSender.cs**

using System.Net;

using System.Net.Mail;

namespace CustomerCommLib

{

public interface IMailSender

{

bool SendMail(string toAddress, string message);

}

public class MailSender : IMailSender

{

public bool SendMail(string toAddress, string message)

{

MailMessage mail = new MailMessage();

SmtpClient SmtpServer = new SmtpClient("smtp.gmail.com");

mail.From = new MailAddress("your\_email\_address@gmail.com");

mail.To.Add(toAddress);

mail.Subject = "Test Mail";

mail.Body = message;

SmtpServer.Port = 587;

SmtpServer.Credentials = new NetworkCredential("username", "password");

SmtpServer.EnableSsl = true;

SmtpServer.Send(mail);

return true;

}

}

}

**Under Folder – CustomerComm.Tests**

**File Name – CustomerCommTests.cs**

using NUnit.Framework;

using Moq;

using CustomerCommLib;

namespace CustomerComm.Tests

{

[TestFixture]

public class CustomerCommTests

{

private Mock<IMailSender>? \_mockMailSender;

private CustomerCommLib.CustomerComm? \_customerComm;

[OneTimeSetUp]

public void Init()

{

\_mockMailSender = new Mock<IMailSender>();

\_mockMailSender

.Setup(x => x.SendMail(It.IsAny<string>(), It.IsAny<string>()))

.Returns(true);

\_customerComm = new CustomerCommLib.CustomerComm(\_mockMailSender.Object);

}

[Test]

public void SendMailToCustomer\_ReturnsTrue()

{

bool result = \_customerComm!.SendMailToCustomer();

Assert.That(result, Is.True);

}

}

}

**Output –**

