This week I continued down the trail of learning more about Unit Testing in Visual Studio. Last week I mentioned that I found a video series on Channel 9 and this week I was able to finish the series and actually follow along in the code and get to write my own Units Tests in Visual Studio.

Here is my first very basic Unit Test for a class called Calculator.cs

using System;

using Microsoft.VisualStudio.TestTools.UnitTesting;

using MyCalculator;

namespace MyCalculator.Tests

{

[TestClass]

public class CalculatorTests

{

[TestMethod]

[TestCategory("Add")]

[Priority(0)] //adds a priority to the tests

public void AddSimple()

{

var calculator = new Calculator(); //Arrange

int sum = calculator.Add(1, 2); //Act

Assert.AreEqual(3, sum); //Assert

}

[TestMethod]

[TestCategory("Divide")]

//write code FIRST

public void DivideSimple()

{

var calculator = new Calculator();

int quotient = calculator.Divide(10, 5);

Assert.AreEqual(2, quotient);

}

[TestMethod]

[Ignore] //ignores this test

[ExpectedException(typeof(DivideByZeroException))] //we are telling visual studio that we are EXPECTING a DivideByZero error

//check for divide by 0

public void DivideByZero()

{

Calculator calculator = new Calculator();

calculator.Divide(10, 0);

}

}

}

There are a couple of different methods that I picked up from the videos, but the biggest one was what they refer to as, AAA.

Arrange, Act and Assert

The first step is to set up the test classes, then call the methods that you want to call in and finally asses to see if the results are correct.

As you can see from my code I was able to check some very simple math applications. In my second [TestMethod] DivideSimple(), I wrote the test BEFORE I wrote the code in the class. This is Test Driven Development. You are to write the tests before you write the business logic.

The second part of the video series was about how you can use Unit Tests to actually test some of the UI. The presenter was using the Unit Tests to actually perform a click event and pass in a value on a visual calculator application. Be testing this you could programmatically be able test hundreds or thousands of different use cases while on lunch break.

Besides the video series I also found a tutorial on visualstudio.com. This tutorial also had other elements including how to set up and install Visual Studio, but the one I focused on was the Unit Testing tutorial.

Create and Run Unit Test Tutorial

<http://www.visualstudio.com/en-us/get-started/create-and-run-unit-tests-vs.aspx>