**Assignment 2**

PROG8170

Software Quality Assurance

School of Engineering and Information Technology

Conestoga College Institute of Technology and Advanced Learning

Submitted to Baljeet Bilkhu

Submitted by:

Chitmanjeet Singh Grewal (8250979)

Date:09/10/2018



**Code of Program.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace SQAAssignment2

{

class Program

{

static void Main(string[] args)

{

int a, b, c;

string side1, side2, side3;

string value = string.Empty;

bool validMenuSelection = false;

while (validMenuSelection == false)

{

Console.WriteLine("Press 1 Enter triangle dimensions");

Console.WriteLine("Press 2 Exit");

value = Console.ReadLine();

switch (value)

{

case "1":

Console.WriteLine("Please enter three values to know the type of triangle");

Console.WriteLine("");

bool isNum;

do

{

side1 = Console.ReadLine();

isNum = int.TryParse(side1, out a);

} while (!(isNum));

do

{

side2 = Console.ReadLine();

isNum = int.TryParse(side2, out b);

} while (!(isNum));

do

{

side3 = Console.ReadLine();

isNum = int.TryParse(side3, out c);

} while (!(isNum));

if ((a <= 0) && (b <= 0) && (c <= 0) && (float.IsNaN(a) && (float.IsNaN(b) && (float.IsNaN(c)))))

{

Console.WriteLine("Unvalid input, Enter valid input");

}

else

{

Console.WriteLine($"Entered input is: {TriangleSolver.Analyze(int.Parse(side1), int.Parse(side2), int.Parse(side3))}");

}

break;

case "2":

{

System.Environment.Exit(0);

break;

}

default:

Console.WriteLine("Enter the valid option");

break;

}

}

}

}

}

**Screen Shots**

**Code of Triangle.cs**

**Source code:**

**Screen Shots**

**SQATAssignment1\_Tests.cs**

**Source code:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Prog8170\_Assignment1;

using NUnit.Framework;

namespace SQATAssignment1\_Tests

{

[TestFixture]

class SQATAssignment1\_Tests

{

[Test]

public void GetLength\_input2\_expectLenghtEquals2()

{

//Arrange

int l = 2;

int w = 6;

Rectangle testRectangle = new Rectangle(l, w);

//Act

int length = testRectangle.GetLength();

Assert.AreEqual(length, l);

}

[Test]

public void SetLength\_input4\_expectLenghtEquals4()

{

//Arrange

int l = 4;

int w = 5;

Rectangle testRectangle = new Rectangle(l, w);

//Act

int length = testRectangle.SetLength(4);

Assert.AreEqual(length, l);

}

[Test]

public void GetWidth\_input5\_expectWidthEquals5()

{

//Arrange

int l = 3;

int w = 5;

Rectangle testRectangle = new Rectangle(l, w);

//Act

int width = testRectangle.GetWidth();

Assert.AreEqual(width, w);

}

[Test]

public void SetWidth\_input6\_expectWidthEquals3()

{

//Arrange

int l = 6;

int w = 3;

Rectangle testRectangle = new Rectangle(l, w);

//Act

int width = testRectangle.SetWidth(3);

Assert.AreEqual(width, w);

}

[Test]

public void GetPerimeter\_inputLength1\_inputWidth1\_expectPerimeterEquals3()

{

//Arrange

int l = 1;

int w = 1;

Rectangle testRectangle = new Rectangle(l, w);

//Act

int Perimeter = testRectangle.GetPerimeter();

Assert.AreEqual(Perimeter, 4 );

}

public void GetArea\_inputLength6\_inputWidth3\_expectAreaEquals18()

{

//Arrange

int l = 6;

int w = 3;

Rectangle testRectangle = new Rectangle(l, w);

//Act

int Area = testRectangle.GetArea();

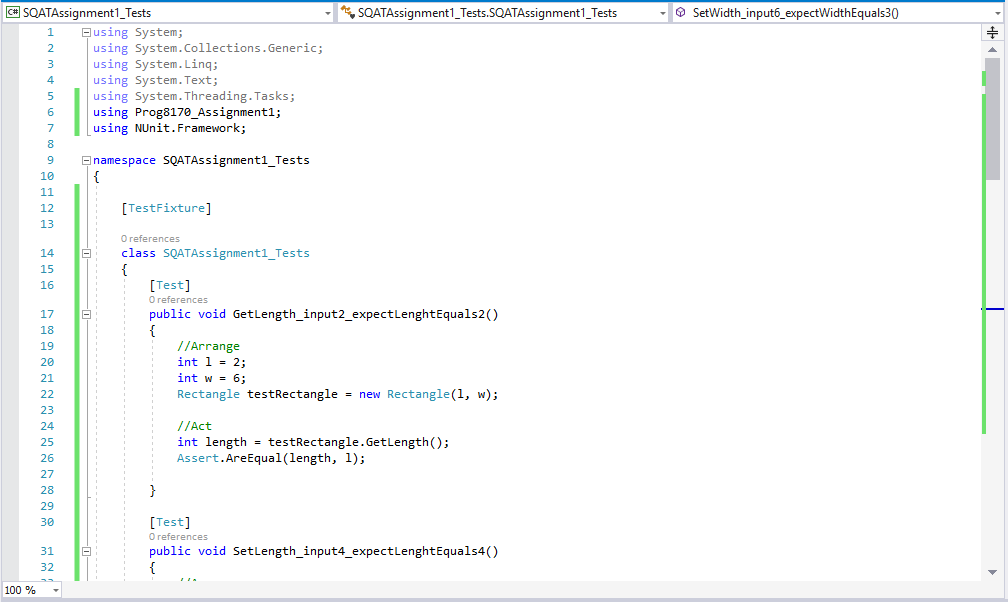
Assert.AreEqual(Area, 18 );

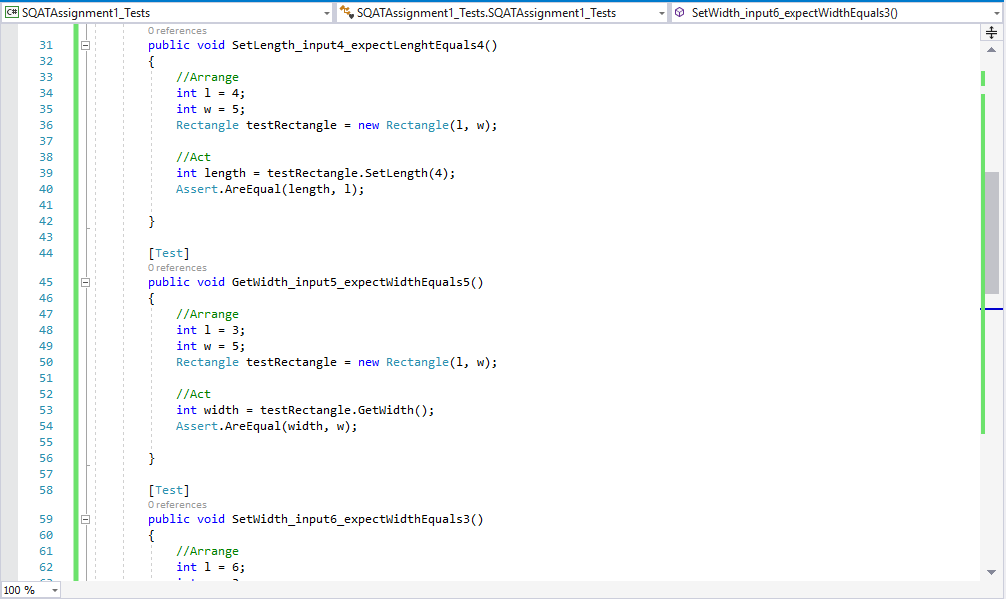
}

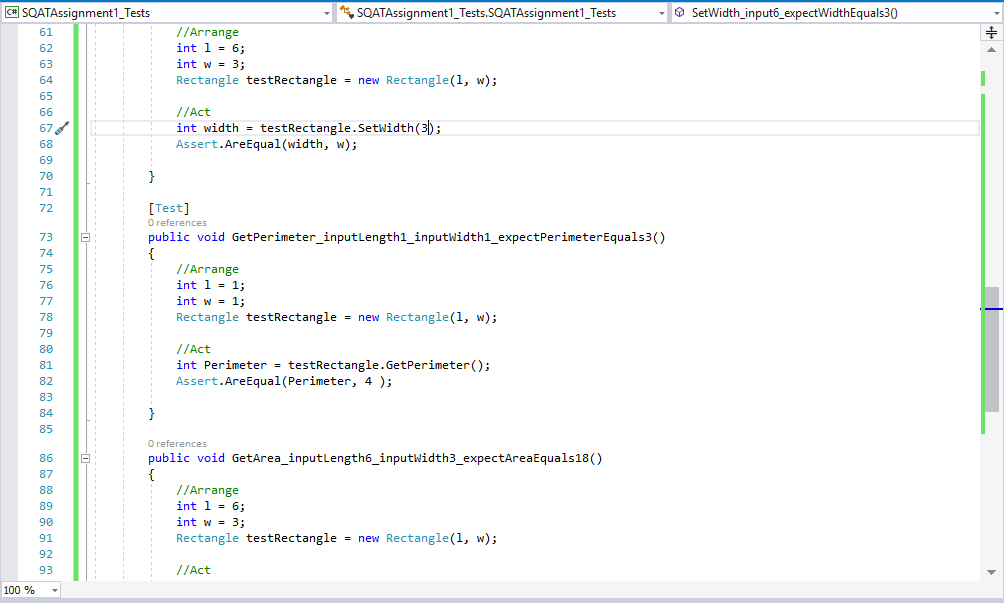
}

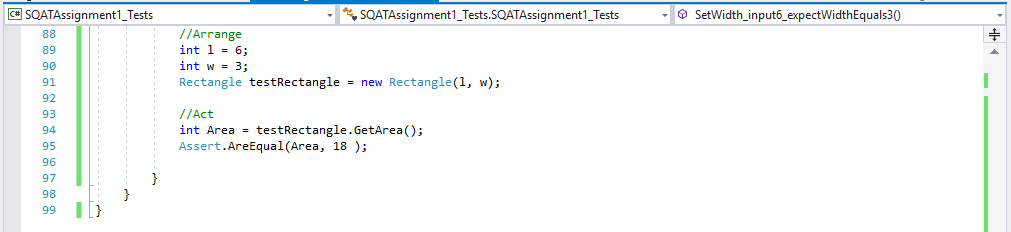
}

**Screen Shots**

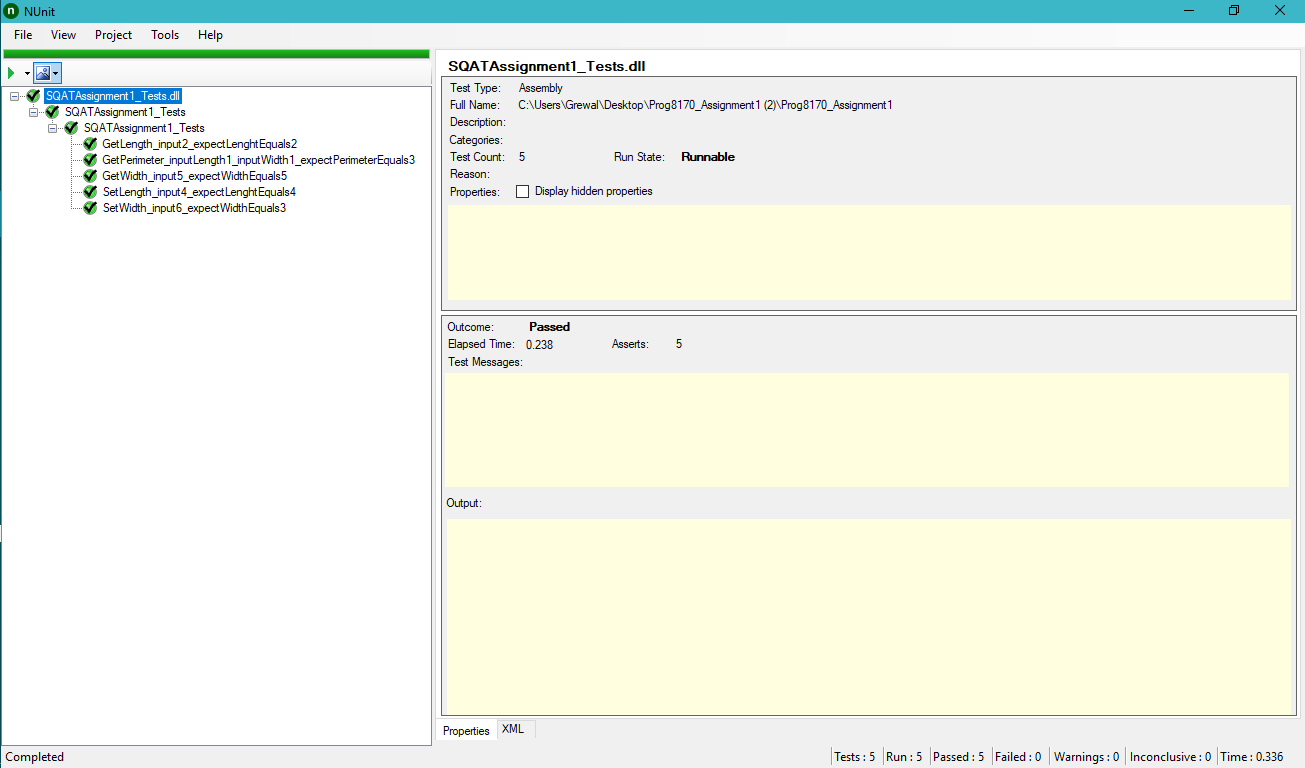




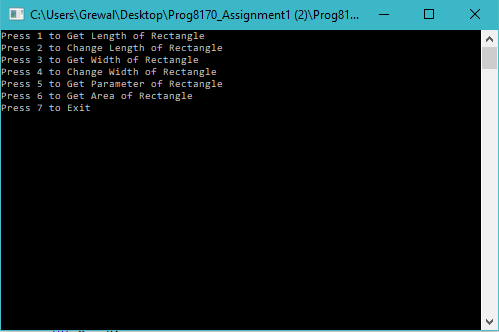




**Screen Shot of NUnit test**



**Screen Shot of Output**



**Screen Shot of Github**

