Assignment 2

Graphical user interface, text

Description automatically generated

Unit Test Source Code

public class Tests

{

[SetUp]

public void Setup()

{

}

[Test]

public void Input\_Length()

{

int expected = 5;

int actual = Rectangular.Rectangle.GetRectangleLength(expected);

Assert.AreEqual(expected, actual);

}

[Test]

public void Input\_width()

{

int width = 3;

int actual = Rectangular.Rectangle.GetRectangleWidth(width);

Assert.AreEqual(width, actual);

}

[TestCase(6, 8)]

public void Validate\_Area(int a, int b)

{

int value = 2 \* (a + b);

int actual = Rectangular.Rectangle.GetRectanglePerimeter(a, b);

Assert.AreEqual(value, actual);

}

[TestCase(6, 8)]

public void Validate\_Perimeter(int a, int b)

{

int value = a \* b;

int actual = Rectangular.Rectangle.GetRectangleArea(a, b);

Assert.AreEqual(value, actual);

}

[Test]

public void Calculate\_Perimeter\_Update\_Results()

{

int width = 3;

int length = 5;

int perimeter = 2 \* (width + length);

int result = Rectangular.Rectangle.GetRectanglePerimeter(width, length);

Assert.AreEqual(perimeter, result);

}

[Test]

public void Calculate\_Area\_Update\_Results()

{

int width = 3;

int length = 5;

int Area = (width \* length);

int result = Rectangular.Rectangle.GetRectangleArea(width, length);

Assert.AreEqual(Area, result);

}

[Test]

public async Task Calculate\_Area\_validate\_Results()

{ }

[Test(ExpectedResult = 4)]

public int TeGetRectangleArea()

{

return 2 \* 2;

}

[Test]

public async Task validate\_Perimeter()

{ }

[Test(ExpectedResult = 8)]

public int TeGetRectanglePerimeter()

{

return 2\*(2 + 2);

}

[Test]

public async Task Enter\_Value\_Return\_Length()

{ }

[Test(ExpectedResult = 2)]

public int ChangeRectangleLength()

{

int a = 2;

Console.WriteLine("Your Length is " + a);

return 2;

}

[Test]

public async Task Enter\_Value\_Return\_Width()

{ }

[Test(ExpectedResult = 4)]

public int TChangeRectangleWidth()

{

int a = 4;

Console.WriteLine("Your Width is " + a);

return 4;

}

[TestCase(4)]

public void Input\_Integer\_Value\_Length\_Field(int expected, params int[] numbers)

{

int NewValue = Rectangular.Rectangle.GetRectangleLength(expected);

Assert.AreEqual(expected, NewValue);

}

[TestCase(6)]

public void Input\_Integer\_Value\_Width\_Field(int expected, params int[] numbers)

{

int NewValue = Rectangular.Rectangle.GetRectangleWidth(expected);

Assert.AreEqual(expected, NewValue);

}

[Test]

public void Input\_Required\_Update\_length()

{

}

private void Length(int input, int expected)

{

int num = Rectangular.Rectangle.GetRectangleLength(expected);

Assert.AreEqual(expected, input);

}

[Test]

public void Input\_Required\_Update\_Width()

{

}

private void Width(int input, int expected)

{

int num = Rectangular.Rectangle.GetRectangleWidth(expected);

Assert.AreEqual(expected, input);

}

}

}

Rectangle Source Code

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using NUnit.Framework;

namespace Rectangular

{

public class Program

{

public static void Main(string[] args)

{

bool exitIndicator = false;

int length = 5, width = 10, perimeter = 0, area = 0;

string menu = "";

do

{

Console.WriteLine("1. Get Rectangle Length \n");

Console.WriteLine("2. Change Rectangle Length \n");

Console.WriteLine("3. Get Rectangle Width \n");

Console.WriteLine("4. Change Rectangle Width \n");

Console.WriteLine("5. Get Rectangle Perimeter \n");

Console.WriteLine("6. Get Rectangle Area \n");

Console.WriteLine("7. Exit \n");

menu = Console.ReadLine();

if (menu == "1" || menu == "2" || menu == "3" || menu == "4"

|| menu == "5" || menu == "6" || menu == "7")

{

Console.Clear();

switch (menu)

{

case "1":

length = Rectangle.GetRectangleLength(length);

Console.WriteLine("The current length is " + length);

Console.ReadLine();

break;

case "2":

length = Rectangle.ChangeRectangleLength(length);

Console.WriteLine("The length has been changed to " + length);

Console.ReadLine();

break;

case "3":

width = Rectangle.GetRectangleWidth(width);

Console.WriteLine("The current width is " + width);

Console.ReadLine();

break;

case "4":

width = Rectangle.ChangeRectangleWidth(width);

Console.WriteLine("The width has been changed to " + width);

Console.ReadLine();

break;

case "5":

perimeter = Rectangle.GetRectanglePerimeter(length, width);

Console.WriteLine("The Perimeter of the Rectangle is " + perimeter);

Console.ReadLine();

break;

case "6":

area = Rectangle.GetRectangleArea(length, width);

Console.WriteLine("The Perimeter of the Rectangle is " + area);

Console.ReadLine();

break;

case "7":

exitIndicator = true;

break;

}

}

else

{

Console.WriteLine("Input Error:\n Please enter any one numeric value from 1 to 7\n");

}

} while (exitIndicator == false);

}

}

}

Program Source Code

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using NUnit.Framework;

namespace Rectangular

{

public class Rectangle

{

public static int GetRectangleLength(int input)

{

int length = input;

return length;

}

public static int ChangeRectangleLength(int input)

{

int length = input;

Console.WriteLine("Pleae Enter Length:");

length = Convert.ToInt32(Console.ReadLine());

return length;

}

public static int GetRectangleWidth(int input)

{

int width = input;

return width;

}

public static int ChangeRectangleWidth(int input)

{

int width = input;

Console.WriteLine("Pleae Enter Width:");

width = Convert.ToInt32(Console.ReadLine());

return width;

}

public static int GetRectanglePerimeter(int num1, int num2)

{

int length = num1;

int width = num2;

int perimeter = 2 \* (length + width);

return perimeter;

}

public static int GetRectangleArea(int num1, int num2)

{

int length = num1;

int width = num2;

int area = length \* width;

return area;

}

}

}