Tymur Koltunov

Student ID 8672727

Assignment#1

2019-09-19

Program.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace assignment1\_part1

{

class Program

{

static void Main(string[] args)

{

int input = 0;

int value = 0;

Rectangle r = new Rectangle();

do

{

Console.WriteLine("1. Create default (1 x 1) rectangle");

Console.WriteLine("2. Create custom rectangle");

Console.Write("Enter menu option: ");

if (!int.TryParse(Console.ReadLine(), out input) || (input > 2 || input < 1))

{

Console.WriteLine("Incorrect input");

Console.WriteLine();

}

else

break;

} while (true);

switch (input)

{

default:

Console.WriteLine("Something is wrong");

break;

case 1:

input = 0;

break;

case 2:

int length = 0;

int width = 0;

do

{

Console.Write("Enter length of the rectangle: ");

if (!int.TryParse(Console.ReadLine(), out length) || length <= 0)

{

Console.WriteLine("Incorrect input");

Console.WriteLine();

}

else

break;

} while (true);

do

{

Console.Write("Enter width of the rectangle: ");

if (!int.TryParse(Console.ReadLine(), out width) || width <= 0)

{

Console.WriteLine("Incorrect input");

Console.WriteLine();

}

else

break;

} while (true);

r = new Rectangle(length, width);

input = 0;

break;

}

Console.WriteLine();

do

{

switch (input)

{

default:

Console.WriteLine("Something is wrong");

break;

case 0:

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

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

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

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

Console.WriteLine("5. Get Rectangle Perimiter");

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

Console.WriteLine("7. Exit");

Console.WriteLine();

Console.Write("Enter menu option:");

if (!int.TryParse(Console.ReadLine(), out input) || (input > 7 || input < 1))

{

Console.WriteLine("Incorrect input");

Console.WriteLine();

input = 0;

}

break;

case 1:

Console.WriteLine("Length of the rectangle is: {0}", r.GetLength());

Console.WriteLine();

input = 0;

break;

case 2:

Console.Write("Enter new length:");

if (!int.TryParse(Console.ReadLine(), out value) || value <= 0)

{

Console.WriteLine("Incorrect input");

Console.WriteLine();

input = 2;

}

else

{

Console.WriteLine("Length changed successfully");

Console.WriteLine();

r.SetLength(value);

input = 0;

}

break;

case 3:

Console.WriteLine("Width of the rectangle is: {0} ", r.GetWidth());

Console.WriteLine();

input = 0;

break;

case 4:

Console.Write("Enter new width:");

if (!int.TryParse(Console.ReadLine(), out value) || value <= 0)

{

Console.WriteLine("Incorrect input");

Console.WriteLine();

input = 4;

}

else

{

r.SetWidth(value);

Console.WriteLine("Width changed successfully");

Console.WriteLine();

input = 0;

}

break;

case 5:

Console.WriteLine("Perimiter of the rectangle is: {0}", r.GetPerimiter());

Console.WriteLine();

input = 0;

break;

case 6:

Console.WriteLine("Area of the rectangle is: {0}", r.GetArea());

Console.WriteLine();

input = 0;

break;

case 7:

return;

}

} while (true);

}

}

}

Rectangle.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace assignment1\_part1

{

public class Rectangle

{

private int length;

private int width;

public Rectangle()

{

length = 1;

width = 1;

}

public Rectangle(int length, int width)

{

this.length = length;

this.width = width;

}

public int GetLength()

{

return length;

}

public int GetWidth()

{

return width;

}

public void SetLength(int length)

{

this.length = length;

}

public void SetWidth(int width)

{

this.width = width;

}

public int GetPerimiter()

{

return length \* 2 + width \* 2;

}

public int GetArea()

{

return length \* width;

}

}

}

RecTests.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using NUnit.Framework;

using assignment1\_part1;

namespace assignment1\_part2

{

[TestFixture]

public class Rec\_Tests

{

[Test]

public void GetLength\_input1\_expectedLengthEquals1()

{

//Arrange

int l = 1;

int w = 2;

Rectangle r = new Rectangle(l, w);

//Act

int length = r.GetLength();

//Assert

Assert.AreEqual(l, length);

}

[Test]

public void GetWidth\_input10\_expectedWidthEquals10()

{

//Arrange

int l = 1;

int w = 10;

Rectangle r = new Rectangle(l, w);

//Act

int width = r.GetWidth();

//Assert

Assert.AreEqual(w, width);

}

[Test]

public void SetLength\_input5\_expectedLengthEquals5()

{

//Arrange

int l = 5;

Rectangle r = new Rectangle();

//Act

r.SetLength(l);

//Assert

Assert.AreEqual(l, r.GetLength());

}

[Test]

public void SetWidth\_input50\_expectedWidthEquals50()

{

//Arrange

int w = 50;

Rectangle r = new Rectangle();

//Act

r.SetWidth(w);

//Assert

Assert.AreEqual(w, r.GetWidth());

}

[Test]

public void GetPerimiter\_length8width5\_expectedPerimiter26()

{

//Arrange

int l = 8;

int w = 5;

int expected = l \* 2 + w \* 2;

Rectangle r = new Rectangle(l, w);

//Assert

Assert.AreEqual(expected, r.GetPerimiter());

}

[Test]

public void GetArea\_length45width32\_expectedArea1440()

{

//Arrange

int l = 45;

int w = 32;

int expected = l \* w;

Rectangle r = new Rectangle(l, w);

//Assert

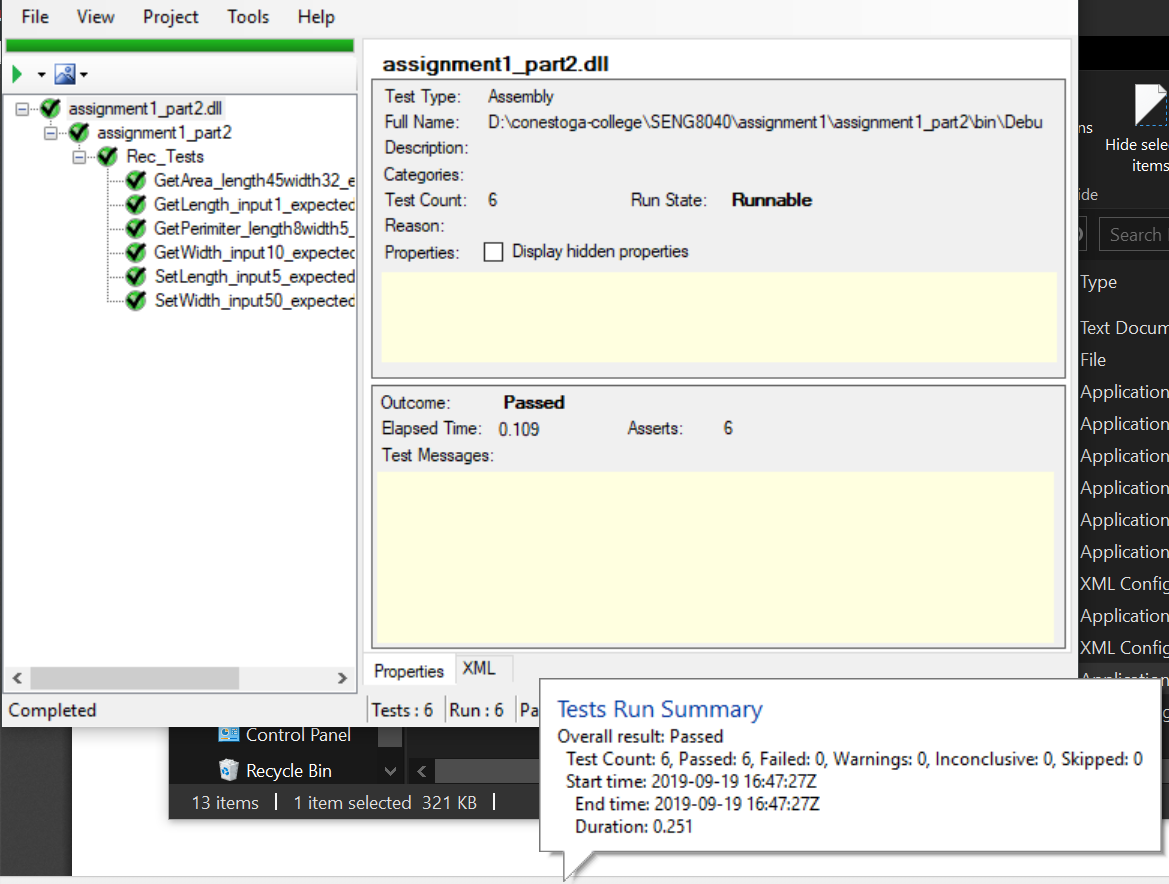
Assert.AreEqual(expected, r.GetArea());

}

}

}

NUnit



Git

