**Vector**

Add a constructor to the Vector class.

Make it so that:

* the fields of this class were initialized in the constructor.
* the Length field (vector length) has become a computed property.

public static void Check()

{

Vector vector = new Vector(3, 4);

Console.WriteLine(vector.ToString());

vector.X = 0;

vector.Y = -1;

Console.WriteLine(vector.ToString());

vector = new Vector(9, 40);

Console.WriteLine(vector.ToString());

Console.WriteLine(new Vector(0, 0).ToString());

}

public class Vector

{

public double X;

public double Y;

public double Length;

// Add constructor!

public override string ToString()

{

return string.Format("({0}, {1}) with length: {2}", X, Y, Length);

}

}

**Code:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace umop14o10zVector

{

class Program

{

static void Main(string[] args)

{

Check();

Console.ReadKey();

}

public static void Check()

{

Vector vector = new Vector(3, 4);

Console.WriteLine(vector.ToString());

vector.X = 0;

vector.Y = -1;

Console.WriteLine(vector.ToString());

vector = new Vector(9, 40);

Console.WriteLine(vector.ToString());

Console.WriteLine(new Vector(0, 0).ToString());

}

public class Vector

{

public double X;

public double Y;

public double Length

{

get { return Math.Sqrt(X \* X + Y \* Y); }

}

public Vector(double x, double y)

{

X = x;

Y = y;

}

public override string ToString()

{

return string.Format("({0}, {1}) with length: {2}", X, Y, Length);

}

}

}

}