

Exercise (1)

Create a class named "**Table**". It must have a constructor, indicating the width and height of the board. It will have a method "ShowData" which will write on the screen the width and that height of the table. Create an array containing 10 tables, with random sizes between 50 and 200 cm, and display all the data.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        class Table
        {
            private float width, height;

            public Table()
            {
            }
            public Table(float width, float height)
            {
                this.width = width;
                this.height = height;
            }
        }
    }
}
```

```
public float Width
{
    set { width = value; }
    get { return width; }
}
public float Height
{
    set { height = value; }
    get { return height; }
}

public void ShowData()
{
    MessageBox.Show("Width: "+width.ToString()+"\nHeight: "+
        height.ToString());
}
} // end of class table
private void button1_Click(object sender, EventArgs e)
{
    Table[] myTables = new Table[10];
    Random rnd = new Random();

    for (int i = 0; i < 10; i++)
    {
        myTables[i] = new Table(rnd.Next(50, 201), rnd.Next(50, 201));
        myTables[i].ShowData();
    }
}
}
```

Exercise (2)

A complex number has two parts: the real part and the imaginary part.

In a number such as $a+bi$ ($2-3i$, for example) the real part would be "a" (2) and the imaginary part would be "b" (-3).

Create a class ComplexNumber with:

A constructor to set the values for the real part and the imaginary part.

Setters and getters for both.

A method "ToString", which would return "(2,-3)"

A method "GetMagnitude" to return the magnitude of the complex number (square root of a^2+b^2)

A method "Add", to sum two complex numbers (the real part will be the sum of both real parts, and the imaginary part will be the sum of both imaginary parts)

Create a test program, to try these capabilities.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace complex
```

```
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        class ComplexNumber
        {
            protected double a, b;

            public ComplexNumber(double realPart, double imaginaryPart)
            {
                a = realPart;
                b = imaginaryPart;
            }
        }
    }
}
```

```
public double GetReal()
{
    return a;
}
```

```
public void SetReal(double a)
{
    this.a = a;
}
```

```
public double GetImaginary()
{
    return b;
}
```

```
public void SetImaginary(double b)
{
    this.b = b;
}
```

```
public new string ToString()
{
    return "(" + a + "," + b + ")";
}
```

```
public double GetMagnitude()
{
    return Math.Sqrt((a * a) + (b * b));
}
```

```
public void Add(ComplexNumber c2)
```

```
    {
        a += c2.GetReal();
        b += c2.GetImaginary();
    }

}

private void button1_Click(object sender, EventArgs e)
{

    ComplexNumber number = new ComplexNumber(5, 2);
    MessageBox.Show("Number is: " + number.ToString());

    number.SetImaginary(-3);
    MessageBox.Show("Number is: " + number.ToString());

    MessageBox.Show("Magnitude is: ");
    MessageBox.Show(number.GetMagnitude().ToString());

    ComplexNumber number2 = new ComplexNumber(-1, 1);
    number.Add(number2);
    MessageBox.Show("After adding: ");
    MessageBox.Show(number.ToString());
}
}
```