Sofia University
Department of Mathematics and Informatics

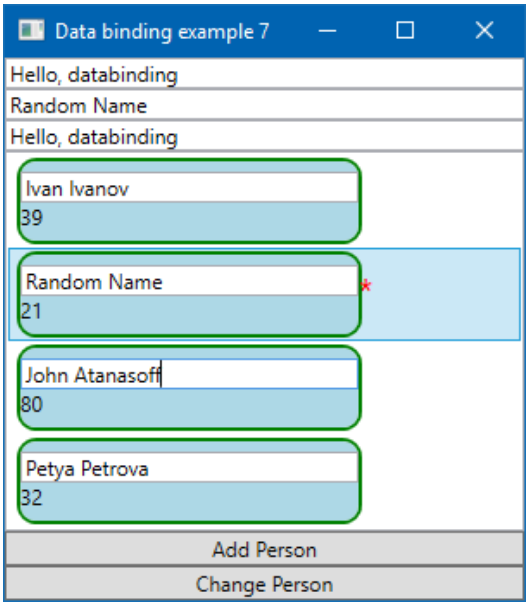
Course : OO Programming C#.NET
Date: December 10, 2018
Student Name:

Lab No. 11

Submit the all C# .NET files developed to solve the problems listed below. Use comments and Modified-Hungarian notation.

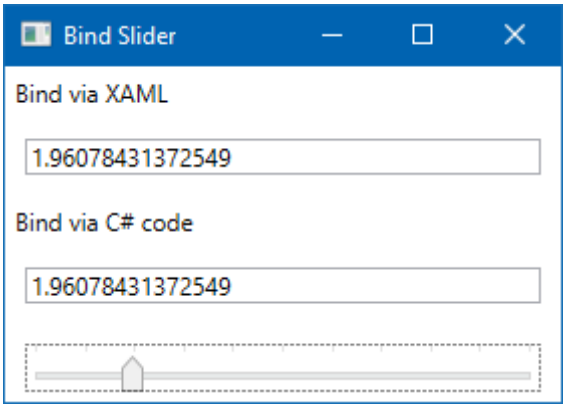
Problem No.1

Make use of the attached sample code **DataBindingExample7.rar** to **add a Validation rule** for the **Age** property of a **Person** to be between **25** and **55** years. Provide the minimum and the maximum year as **parameters to the Validation rule**. Add a **Button** to **delete elements from the list** of **people** so that the databinding with the **ListBox** is preserved.



Problem No.2

Consider a WPF application that contains two elements- a **TextBox** at the top and a **Slider** control beneath it. You'd like to have the **TextBox** represent the value of the **Slider** such that when you move the **Slider** back and forth, the value in the **TextBox** changes to show the Slider's current value. Use databinding to implement this task.

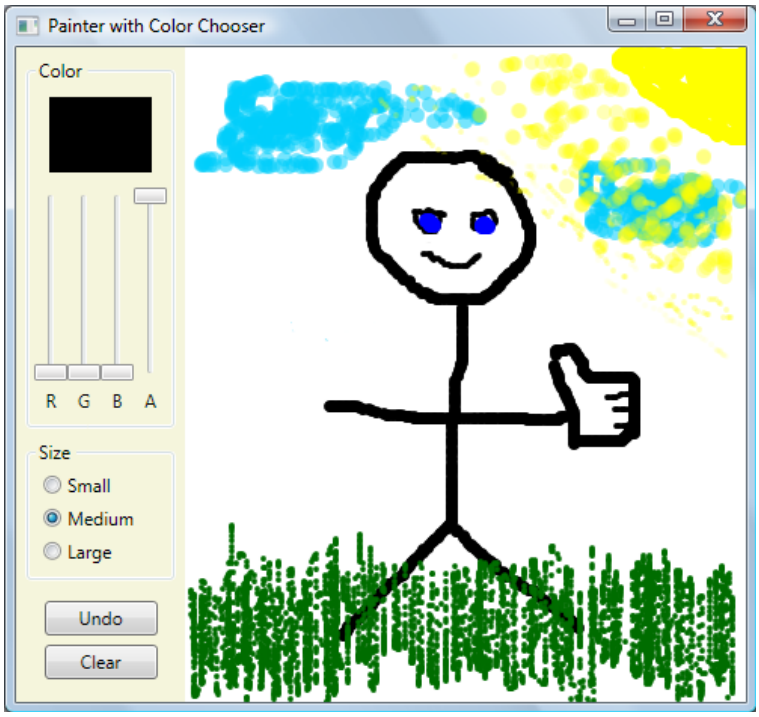


Implement a **Converter** class and use it display the current slider value with 2 digits after the decimal point

How can you implement the same feature using the **StringFormat** property of **Binding**?

Problem No.3

Incorporate an RGBA color chooser into the Painter example (*attached* to the Lab as `Painter.rar`) to look like the figure displayed below m**aking use of user- defined styles that target respectively the Slider and the Button types**. Apply the styles to the sliders and the buttons in the required GUI design style to make all the sliders and buttons (with **rounded corners**) look the same. Let the user select the brush color using the color chooser instead of the group of RadioButtons. Use **Expression Blend** for this purpose.



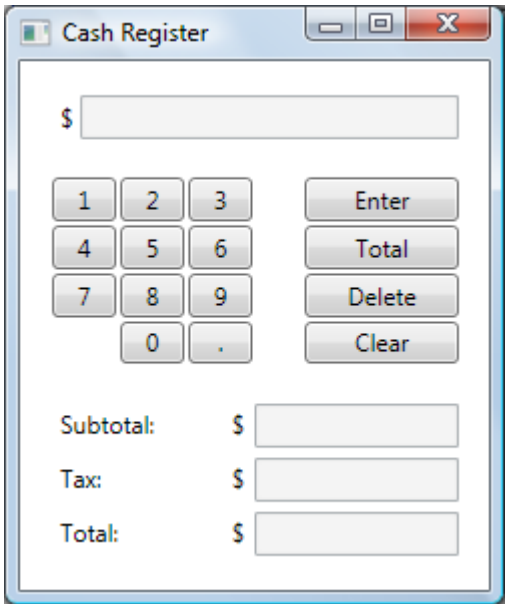
Additionally, **modify the drawing procedure** to **use line segments** connecting consecutive positions of the mouse cursor, while dragging it over the Canvas. The free **line thickness and color**, **Undo** and **Clear** must be **managed by means of the tools on the left side** of the Canvas as shown on the above figure

Problem No. 4

Make use of the attached code **for CommandReverse.rar** to create another user- defined Command that changes the Background of the TextBox to Green when the Length of the Text in it is longer than 8 characters and makes the Background Red when the Length is less than 8 characters

Problem No.5

Create a cash-register application modeled after the one presented below. It should allow users to enter a series of prices, then obtain the total. The **Delete** button should clear the current entry, and the **Clear** button should reset the application .**Use appropriately defined styles for the buttons**.



Implement the task as a WPF UserControl and test it in a separated WPF application.

Problem No.6

Create a C#.NET calculator as a WPF ***UserControl*** that allows the user to input numbers in a textbox and choose an operation to perform on them (addition, multiplication, division, subtraction) as it is done with a usual calculator (*see the design of the Calculator application in the Accessories Program group in the MS Windows environment*) . Design buttons to execute these operations, as well as, buttons:

a) to remember the currently displayed number (**M** operation)
b) to add the currently displayed number with the number stored in memory and display the result (**M+** operation)
c) to subtract the currently displayed number with the number stored in memory and display the result (**M-** operation)
d) to clear the memory (**MC-** operation

The methods performing the Calculator operations must be **public.** There should be also **two *`public set`* properties** for the user numeric input, necessary to complete the calculator operations**.** There should be a ***`public get`* property** for the Calculator result.

Catch division by zero exceptions, by canceling the division operation and displaying an error message in the textbox. **Allow only legal numeric user input** in the textbox**.**

Write a C#.Net WPF application to test this **user contro**l. (*submit the source code of the user control, its DLL file and the full set of files of the C#.Net Windows application*)

Problem No.7

In order to get basic skills in Expression Blend complete the tasks displayed in the following presentations:

https://www.youtube.com/watch?v=tbulRaoc7ow
https://www.youtube.com/watch?v=2cfFQvGejRA

https://www.youtube.com/watch?v=sVcbjbYvmzM
https://www.youtube.com/watch?v=2cfFQvGejRA
https://www.youtube.com/watch?v=LS2vxIDn2vI