Variables, Expressions, and Statements Assignment (Calculator)

Due date: ____________________

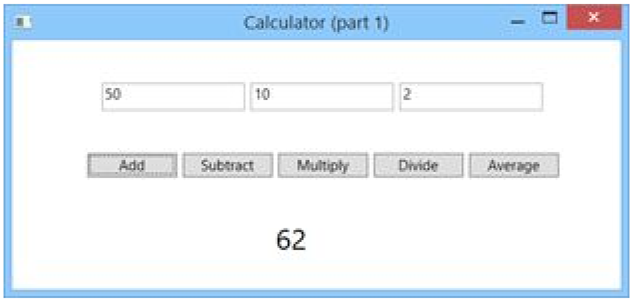
Objectives

The purpose of this assignment is to assess your understanding of the following concepts:

- Understand WPF (Windows Presentation Foundation) programming with some basic controls
- Event-driven programming
- Understand variables of string and integer types
- Understand variable scope
- Understand expressions and statements

Remember, if you're spending a lot of time on something that isn't part of the assignment objectives, then get some help. Search the internet first, then ask a neighbor. If you still can't figure it out, then ask Mr. Brooks.

Assignment Part 1



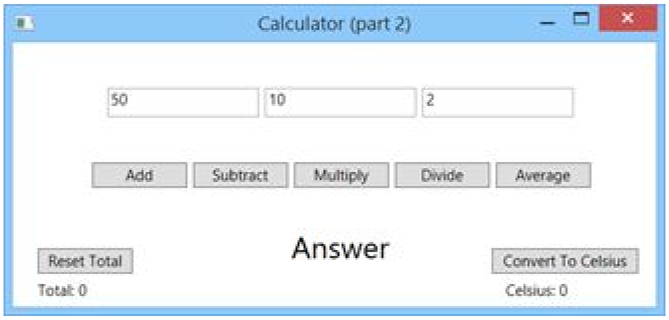
50 10 2

62

Write a WPF program that has three text boxes for numbers (with an initial value of 50, 10, and 2), and five buttons: add, subtract, multiply, divide, and average. Below that is a TextBlock that displays the answer of the calculation using the operation the user clicked on. Something like the picture at right.

For example, suppose the values of the three text boxes are 10, 5, and 2, respectively. The Add button adds the three values (the example answer would be 17). The Subtract button subtracts the last two values from the first value (the example answer would be 3). The Multiply button would multiply the three values (the example answer would be 100). The Divide button would divide the first value by the second value and that answer by the third value (the example answer would be 1). The Average button would calculate the average of the three values (the example answer would be 5).

Assignment Part 2



50 10 2

Answer

Total: 0

Celsius: 0

Once that is working, add two more buttons and TextBlocks, as pictured at right. On the left side is a running total (starting at zero when the program starts). Each time an operation (add, subtract, multiply, divide, or average) is performed, the running total is increased by the answer value. Clicking on the "Reset Total" button resets the total back to zero.

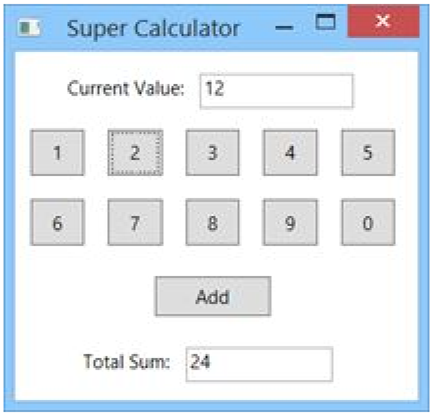
On the right side is a button ("Convert to Celsius") that converts the answer to the last operation from Fahrenheit to Celsius. (The formula for conversion is C = (F – 32) × 5 ÷ 9.)

Points Breakdown

- 10 points: your window has all of the necessary controls.
- 20 points: all of the variables in your program (including the controls) have meaningful names and use the "camelCase" naming convention.
- 30 points: 6 points for each of the Add, Subtract, Multiply, Divide, and Average buttons generating the correct answer for the input values.
- 20 points: the program keeps a running total on the left-hand side with a "Reset Total" button that resets the total back to 0. The total must be displayed in a single TextBlock control with the text "Total:" preceding the value. The value for total must be stored in an integer variable (i.e. not in the TextBlock text).
- 20 points: the program has a button that converts the current answer value from Fahrenheit to Celsius. The total must be displayed in a single TextBlock control with the text "Celsius:" preceding the value.

Additional Learning

- (1 ticket/5 bonus points) Change your calculator program to use **`Slider`** controls as input instead of TextBoxes.
- (up to 2 tickets/10 bonus points) Create a new calculator program that looks like the picture at right. The "current value" and "total sum" text boxes should have the **`IsReadOnly`** property set to True (i.e. box is checked). Clicking the numbered buttons creates a number the "current value" text box. For example, from a current value of 0, clicking the 1 button should change the current value to 1; clicking the 4 button should change the current value to 14; clicking the 8 button should change the current value to 148. Once the desired current value is created, clicking the Add button adds the current value to the total sum (which keeps a running total) then clears the current value (resetting it back to zero). This program will earn one ticket if it functions correctly and another ticket if the code never calls **`Convert.ToInt32`**, or in other words uses the text boxes purely for displaying values, not for storing values.



Current Value: 12

Total Sum: 24