Programming II

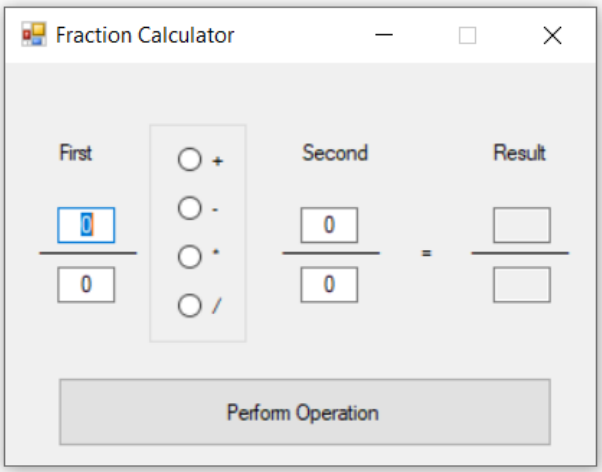
Exercise

You will create a new project for this application called **`FractionCalculator`**. To get a head start copy all the controls from your previous project **`IntroCalculator`**.

1. Open the previous project (if you were not present at the last class, then a completed project is available at the instructor's p drive), go to the form designer tab, select all the controls, copy them and switch to the next project, go to the form designer and then paste all the controls.
2. Also copy the relevant code from your previous project to the current one.
3. Remember none of the copied event handler will be connected; you will have to wire-up them manually.

> You must exhibit the same behavior as **`IntroCalculator`** i.e. the computation is done and the results display when the button is click or the operation is changed.

The Fraction Calculator GUI



Enhancements over the previous application

Filter the inputs to the TextBoxes:

Insert the following lines in the **`KeyPress`** event handler for all of the input textboxes so that only digits are accepted.

```
if (char.IsDigit(e.KeyChar))
```

```
  return;
else
  e.Handled = true;          //discard the non-digit entries
```

Specify the tab order:

Set the **`TabIndex`** property for all the interactable components so that the user is able to use the application without the support of a mouse.

Result TextBoxes shouldn't be changeable:

Set the property **`ReadOnly`** of the Result textboxes to **`True`** so the user cannot change the results.

Fraction Class:

Additionally, you will need to add code to create the **multiply** and **divide** static methods. It is good practice to define classes in separate files.

```
    public class Fraction
    {
        public int Numerator { get; set; }
        public int Denominator { get; set; }

        public Fraction(int numerator = 0, int denominator = 1)
        {
            Numerator = numerator;
            Denominator = denominator;
        }

        public static Fraction Add(Fraction left, Fraction right)
        {
            int numerator = (left.Numerator * right.Denominator) + (right.Numerator *
left.Denominator);
            int denominator = left.Denominator * right.Denominator;

            return new Fraction(numerator, denominator);
        }

        public static Fraction Subtract(Fraction left, Fraction right)
        {
            int numerator = (left.Numerator * right.Denominator) - (right.Numerator *
left.Denominator);
            int denominator = left.Denominator * right.Denominator;

            return new Fraction(numerator, denominator);
        }

        public override string ToString()
        {
            return $"[{Numerator}, {Denominator}]";
        }
    }
```

Modify the Operation Method:

You will need to modify your code to work with fractions instead of ints.

> A useful debugging tool is the `MessageBox.Show()` method. It creates a pop-up window and displays a message.
>
> It is valuable as a debugging tool, where you may display the values of variables.