Department of Instrumentation and

Automation

Faculty of Technology

University of Colombo

IA 2018 Rapid Application Development

Group Assignment

Scientific Calculator

(Group 5)

Due Date of Submission

[26 / 03 / 2020]

Table of Content

1.Software Functionalities

This application consists with 3 major parts

1. Standard Calculator
2. Scientific Calculator
3. Temperature Converter

and all of these components are integrated into a single MDI application. This application is developed using the Visual Studio IDE, and the UI elements are designed using Windows Forms. Aside from that, all logic and components are written in C#.

When the.exe file is run, the user will see the splash screen (**Figure 1**) that was created in Adobe Photoshop and inserted into the application. After the application is launched, the full calculator will be displayed by default (**Figure 2**), and the user can change the mode by clicking the stripe menu.

When comes to functionalities in each component A Standard calculator is performs arithmetic operations on numbers. The simplest calculators can do only addition, subtraction, multiplication, and division. Scientific calculators include exponents, log, natural log (ln), trig functions, and memory. These functions are vital when you're working with scientific notation or any formula with a geometry component. Finally, the temperature can be converted between Celsius Fahrenheit and Kelvin using the unit converter.

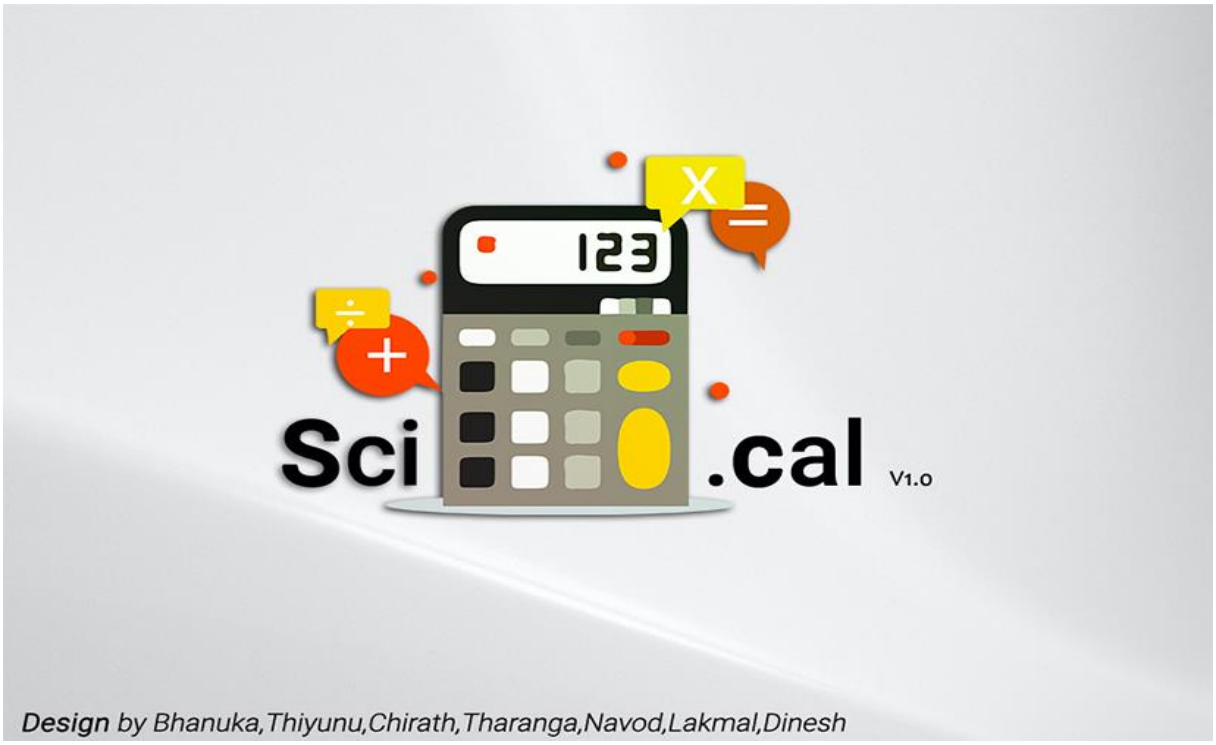

Figure 1

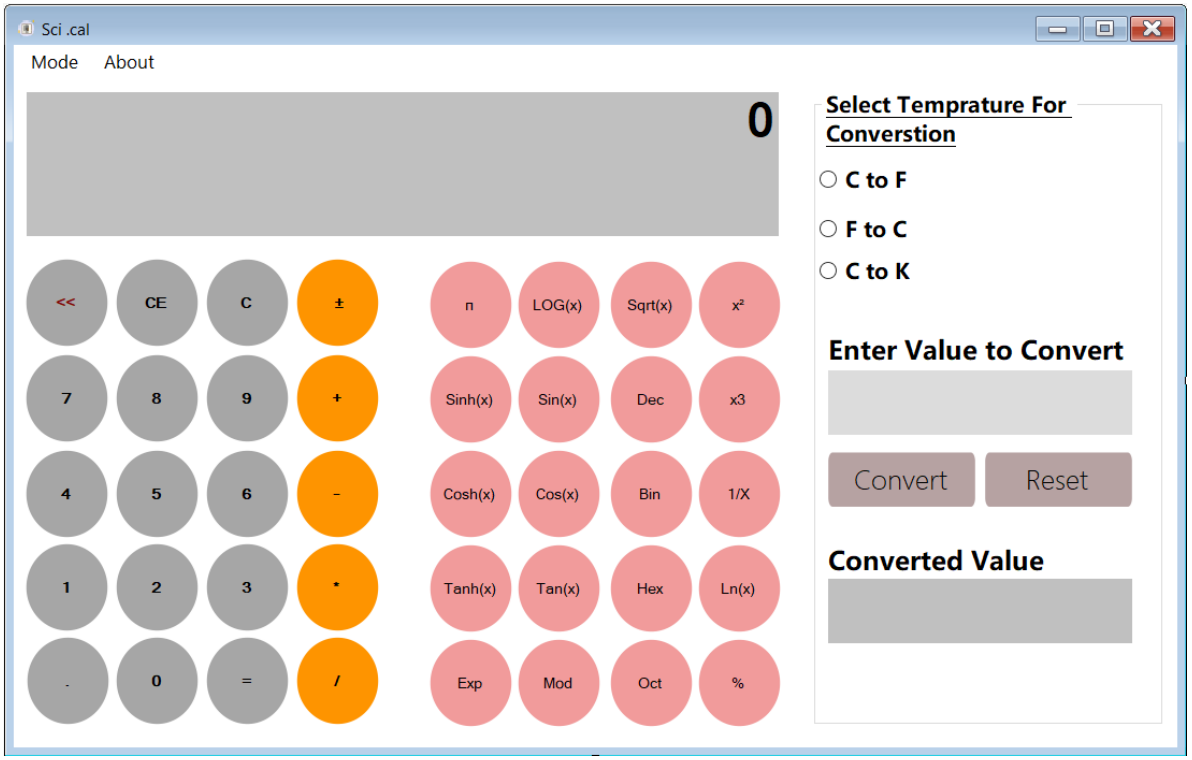

Figure 2

2.Software Structure

2.1 Splash Screen

Adobe Photoshop was used to create the main design for the splash page (**Figure 3**). That image was used as the form 1 background image, and for the animation, we used a loading timer and two panels, the first of which is a loading panel and the second of which is a progress bar; when the application starts, the progress bar will fill up the loading panel.

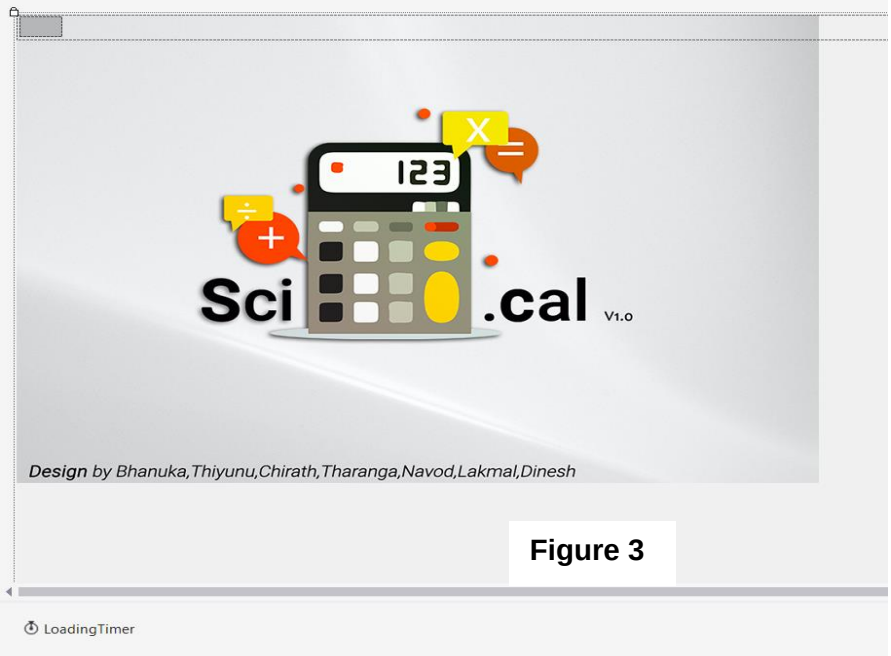

Figure 3

For the Loading time we used C# code animate splash screen progress bar.in there we initialized component (**Figure 4**) and `LoadingTimer.(stop)` when `private void LoadingTimer_Tick()` IF Logic True.

```
using System;
using System.Windows.Forms;

namespace windows_calculator
{
    3 references
    public partial class Form2 : Form
    {
        1 reference
        public Form2()
        {
            InitializeComponent();
        }

        0 references
        private void pictureBox1_Click(object sender, EventArgs e)...

        1 reference
        private void LoadingTimer_Tick(object sender, EventArgs e)
        {
            panel2.Width += 3;
            if (panel2.Width >= 1099)
            {
                LoadingTimer.Stop();
                Form1 f1 = new Form1();
                f1.Show();
                this.Hide();
            }
        }

        1 reference
        private void Form2_Load(object sender, EventArgs e)...
    }
}
```

Figure 4

2.2 Standard Calculator

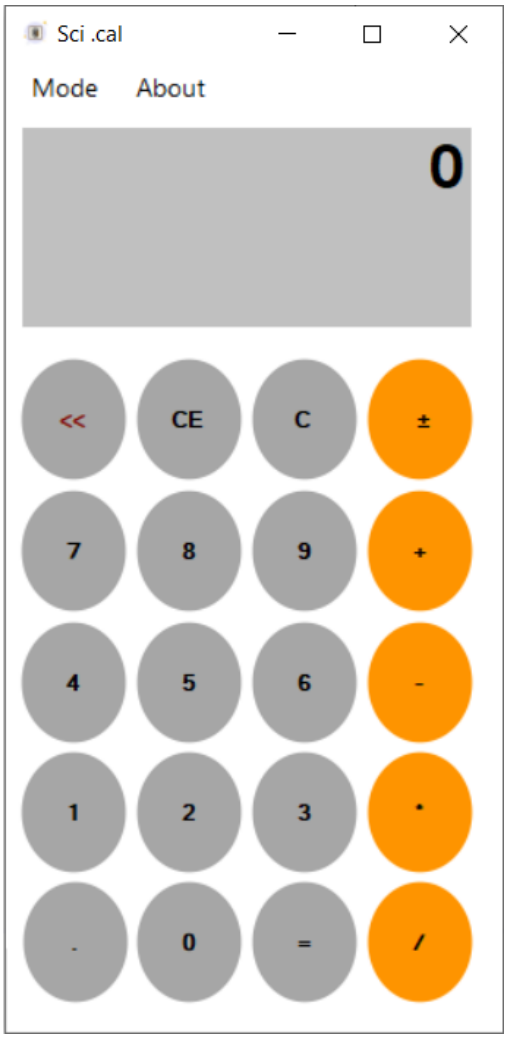

Figure 5

For the Standard calculator (**Figure 5**), the design is done on Form 1, and buttons are added into the form and aligned in a simple manner, as we can see at the top of the application strip Tool menu added in there are two options Mode and About. In the mode section, you can select calculator mode.

You will be able to perform simple arithmetic operations such as addition, division, multiplication, and subtraction in the Standard Mode.

when talking about algorithm all buttonclick event begin set to event called as button_sender. Before button_click we initialized `Globle Variables` into our C# code. (**Figure 6**)

```
1   using System;
2   using System.Windows.Forms;
3
4   namespace windows_calculator
5   {
        4 references
6       public partial class Form1 : Form
7       {
8           // Globle Variables
9           private Double results = 0;
10
11          private String operation = "";
12
13          private bool enter_value = false;
14
15          private float iCelsius, iFahrenheit, ikelvin;
16          private char iOperation;
17
            1 reference
18          public Form1()...
22
            0 references
23          private void sciToolStripMenuItem_Click(object sender, EventArgs e)...
27
            1 reference
28          private void button1_Click(object sender, EventArgs e)
29          {
30              if (txtDisplay.Text.Length > 0)
31              {
32                  txtDisplay.Text = txtDisplay.Text.Remove(txtDisplay.Text.Length - 1, 1);
33              }
34
35              if (txtDisplay.Text == "")
36              {
37                  txtDisplay.Text = "0";
38              }
39          }
40
```

Figure 6

Then after display label named as `txtDisplay` when we are pressing any button it will shows by using following. (**Figure 7**)

```
83          }
84
            11 references
85          private void button_Click(object sender, EventArgs e)
86          {
87              //grab number input and display
88
89              if ((txtDisplay.Text == "0") || (enter_value))
90                  txtDisplay.Text = "";
91              enter_value = false;
92              Button num = (Button)sender;
93
94              //enter decimal and show only one decimal place
95              if (num.Text == ".")
96              {
97                  if (!txtDisplay.Text.Contains("."))
98                      txtDisplay.Text = txtDisplay.Text + num.Text;
99              }
100             else
101             {
102                 txtDisplay.Text = txtDisplay.Text + num.Text;
103             }
104         }
105
```

Figure 7

For the arithmetic operation we used following line of code.in there we used Switch Case operation to detect clicked operator. (**Figure 8**)

```
            1 reference
141         private void button19_Click(object sender, EventArgs e)
142         {
143             // Group as click events  + - / * mod exp operator buttons and assign arithmetics operation
144             lblShowOp.Text = " ";
145             switch (operation)
146             {
147                 case "+":
148                     txtDisplay.Text = (results + Double.Parse(txtDisplay.Text)).ToString();
149                     break;
150
151                 case "-":
152                     txtDisplay.Text = (results - Double.Parse(txtDisplay.Text)).ToString();
153                     break;
154
155                 case "/":
156                     txtDisplay.Text = (results / Double.Parse(txtDisplay.Text)).ToString();
157                     break;
158
159                 case "*":
160                     txtDisplay.Text = (results * Double.Parse(txtDisplay.Text)).ToString();
161                     break;
162
163                 case "Mod":
164                     txtDisplay.Text = (results % Double.Parse(txtDisplay.Text)).ToString();
165                     break;
166
167                 case "Exp":
168                     double i = Double.Parse(txtDisplay.Text);
169                     double q;
170                     q = (results);
171                     txtDisplay.Text = Math.Exp(i * Math.Log(q * 4)).ToString();
172                     break;
173             }
174         }// END Group
175
```

Figure 8

2.3 Scientific Calculator

For the Scientific calculator, the design is done on Form 1, and buttons are added into the form and aligned in as (**Figure 9**), as we can see at the top of the application strip Tool menu added in there are two options Mode and About. In the mode section, you can select calculator mode. Other than standard mode, it will be able to perform tasks such as finding trig functions, power and exponent, and so on.

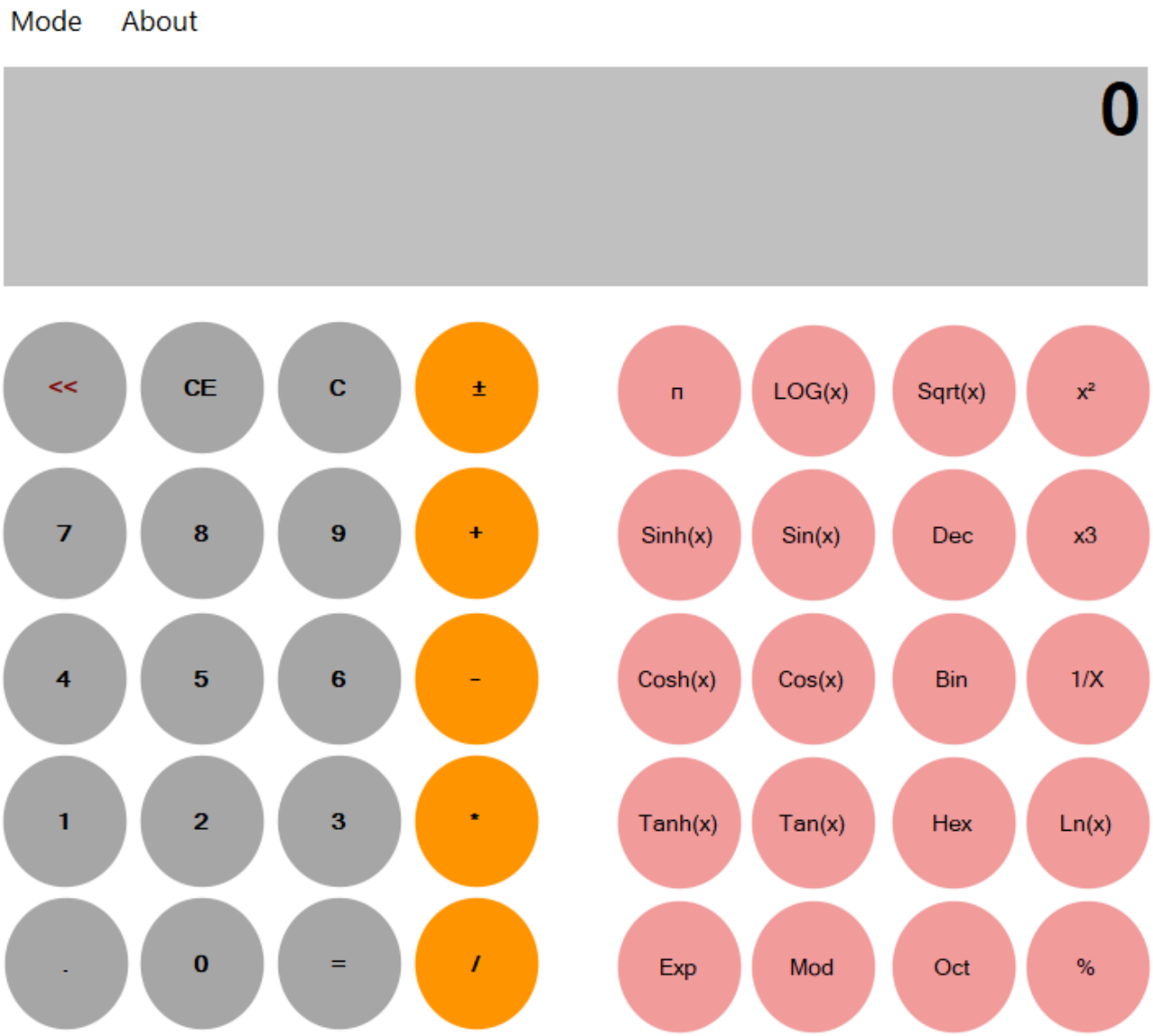

Figure 9

When it come to coding functionalities, we added following code line to find trig functions.

(**Figure 10**)

```
192         //trig identites (Sin, Cos, Tan, Sinh,Cosh,Tanh)
            1 reference
193         private void button23_Click(object sender, EventArgs e)
194         {
195             double Sinh = Double.Parse(txtDisplay.Text);
196             lblShowOp.Text = System.Convert.ToString("Sinh" + "(" + (txtDisplay.Text) + ")");
197             Sinh = Math.Sinh(Sinh);
198             txtDisplay.Text = System.Convert.ToString(Sinh);
199         }
200
            1 reference
201         private void btnsin_Click(object sender, EventArgs e)
202         {
203             double sin = Double.Parse(txtDisplay.Text);
204             lblShowOp.Text = System.Convert.ToString("Sin" + "(" + (txtDisplay.Text) + ")");
205             sin = Math.Sin(sin);
206             txtDisplay.Text = System.Convert.ToString(sin);
207         }
208
            1 reference
209         private void button30_Click(object sender, EventArgs e)
210         {
211             double cos = Double.Parse(txtDisplay.Text);
212             lblShowOp.Text = System.Convert.ToString("Cos" + "(" + (txtDisplay.Text) + ")");
213             cos = Math.Cos(cos);
214             txtDisplay.Text = System.Convert.ToString(cos);
215         }
216
            1 reference
217         private void cosh_Click(object sender, EventArgs e)
218         {
219             double Cosh = Double.Parse(txtDisplay.Text);
220             lblShowOp.Text = System.Convert.ToString("Cosh" + "(" + (txtDisplay.Text) + ")");
221             Cosh = Math.Cosh(Cosh);
222             txtDisplay.Text = System.Convert.ToString(Cosh);
```

```
224
            1 reference
225         private void btntanh_Click(object sender, EventArgs e)
226         {
227             double Tanh = Double.Parse(txtDisplay.Text);
228             lblShowOp.Text = System.Convert.ToString("Tanh" + "(" + (txtDisplay.Text) + ")");
229             Tanh = Math.Tanh(Tanh);
230             txtDisplay.Text = System.Convert.ToString(Tanh);
231         }
232
            1 reference
233         private void button32_Click(object sender, EventArgs e)
234         {
235             double Tan = Double.Parse(txtDisplay.Text);
236             lblShowOp.Text = System.Convert.ToString("Tan" + "(" + (txtDisplay.Text) + ")");
237             Tan = Math.Tan(Tan);
238             txtDisplay.Text = System.Convert.ToString(Tan);
239         }
240
241         // End trig identites
```

Figure 10

And also this lines of codes calculates Log Exp Mod and Ln (**Figure 11**)

```
283
284         // calculate Log
            1 reference
285         private void button39_Click(object sender, EventArgs e)
286         {
287             double ilog = Double.Parse(txtDisplay.Text);
288             lblShowOp.Text = System.Convert.ToString("log" + "(" + (txtDisplay.Text) + ")");
289             ilog = Math.Log(ilog);
290             txtDisplay.Text = System.Convert.ToString(ilog);
291         }
```

```
162
163             case "Mod":
164                 txtDisplay.Text = (results % Double.Parse(txtDisplay.Text)).ToString();
165                 break;
166
167             case "Exp":
168                 double i = Double.Parse(txtDisplay.Text);
169                 double q;
170                 q = (results);
171                 txtDisplay.Text = Math.Exp(i * Math.Log(q * 4)).ToString();
172                 break;
```

Figure 11

Finally, we inserted PI valve into program. (**Figure 12**), as an added feature, we integrated a base converter into the scientific calculator (**Figure 13**) it will convert number between bases.

```
46
            1 reference
47          private void button21_Click(object sender, EventArgs e)
48          {
49              txtDisplay.Text = "3.141592653589976323";
50              // Value of PI
51          }
52
```

```
243         // Base converstion 2 8 10 16
            1 reference
244         private void button31_Click(object sender, EventArgs e)
245         {
246             int a = int.Parse(txtDisplay.Text);
247             txtDisplay.Text = System.Convert.ToString(a, 2);
248         }
249
            1 reference
250         private void button35_Click(object sender, EventArgs e)
251         {
252             int a = int.Parse(txtDisplay.Text);
253             txtDisplay.Text = System.Convert.ToString(a, 8);
254         }
```

Figure 12

```
255
                1 reference
256             private void button28_Click(object sender, EventArgs e)
257             {
258                 int a = int.Parse(txtDisplay.Text);
259                 txtDisplay.Text = System.Convert.ToString(a, 10);
260             }
261
                1 reference
262             private void button33_Click(object sender, EventArgs e)
263             {
264                 int a = int.Parse(txtDisplay.Text);
265                 txtDisplay.Text = System.Convert.ToString(a, 16);
266             }
267
268             // END Base converstion
269
```

Figure 13

2.4 Temperature Converter

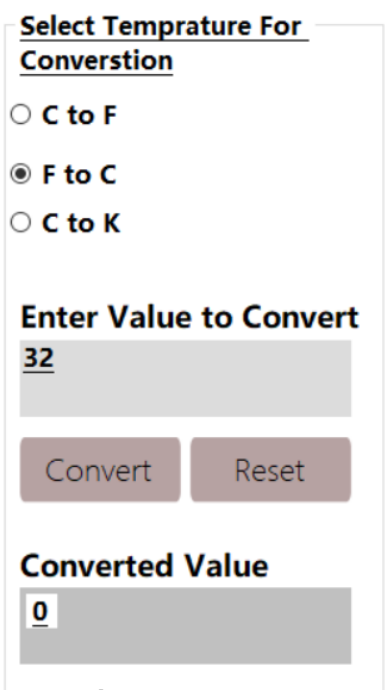

Figure 14

Temperature Converter (**Figure 14**) can be used to Convert temperature between two unit such as Celsius, Fahrenheit and kelvin.1st you need to select unit that you want to convert then enter the value and press convert button you will able to see the result below, just in case if you want to rest all those thing press reset button all will be set to default mode.

For the logic we defined event for radio buttons and pass that value to function below. (**Figure 15**)

```
316             ate void radioButton1_CheckedChanged(object sender, EventArgs e)
311             {
312                 iOperation = 'C';
313             }
314
                1 reference
315             private void rbFtoC_CheckedChanged(object sender, EventArgs e)
316             {
317                 iOperation = 'F';
318             }
319
                1 reference
320             private void radioButton3_CheckedChanged(object sender, EventArgs e)
321             {
322                 iOperation = 'K';
323             }
324
                1 reference
325             private void menuStrip1_ItemClicked(object sender, ToolStripItemClickedEventArgs e)
326             {
327             }
328
```

```
346         // Temperature Conversion
            1 reference
347         private void tempCvt_Click(object sender, EventArgs e)
348         {
349             switch (iOperation)
350             {
351                 case 'C':
352                     // C to F cvt.
353                     iCelsius = float.Parse(Tempin.Text);
354                     lblTempout.Text = (((iCelsius * 1.8) + 32).ToString());
355                     break;
356
357                 case 'F':
358                     // F to C cvt.
359                     iFahrenheit = float.Parse(Tempin.Text);
360                     lblTempout.Text = ((((iFahrenheit - 32) * 5) / 9).ToString());
361                     break;
362
363                 case 'K':
364                     // Cvt C to kelvin
365                     ikelvin = float.Parse(Tempin.Text);
366                     lblTempout.Text = (((ikelvin) + 273.15).ToString());
367                     break;
368             }
369         }
370
            1 reference
371         private void tempReset_Click(object sender, EventArgs e)
372         {
373             Tempout.Clear();
374             Tempin.Clear();
375             lblTempout.Text = "";
376             rbCtoF.Checked = false;
377             rbFtoC.Checked = false;
378             rbKelvin.Checked = false;
379         }
380
381         // Temperature Conversion End
382     }
383 }
```

Figure 15

In the temperature conversion function, we used a switch case operation to identify the radio box input. Once identified, the input value is passed into an equation and the result is displayed on the label in the image below. (**Figure 16**)

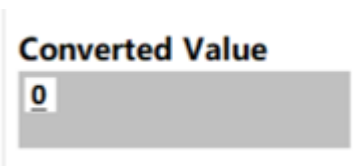

Figure 16

3.Refarance

- https://docs.microsoft.com/en-us/dotnet/desktop/winforms/get-started/create-app-visual-studio?view=netdesktop-5.0
- https://docs.microsoft.com/en-us/visualstudio/install/install-visual-studio?view=vs-2019
- https://docs.microsoft.com/en-us/dotnet/csharp/
- https://www.youtube.com/watch?v=-D0UoCAYG28&t=1923s
- https://www.youtube.com/watch?v=fPpYilRalHk

4.Contribution

Name	Index Number	Contribution
Bhanuka Kamantha	2018T00206	Splash screen and report
Thiyunu Jayawicrama	2018T00205	Temperature Converter (K, C, F)
Tharanga Karunasena	2018T00207	Developed standard cal UI from 0 to 9 and CE, PI, ln(x)
Navod Kamarasiri	2018T00209	Developed the arithmetics (+, - , *,/, MOD, EXP)
Dinesh Buddhika	2018T00210	Developed the trigonometry (sin, cos, tan, sinh, cosh, tanh)
Chirath Janith	2018T00208	Developed the BIN, OCT, DEC, HEX and Logarithm
P K A Lakmal	2018T00211	Developed the Roots and Powers (√, X^2, X^3, 1/X)