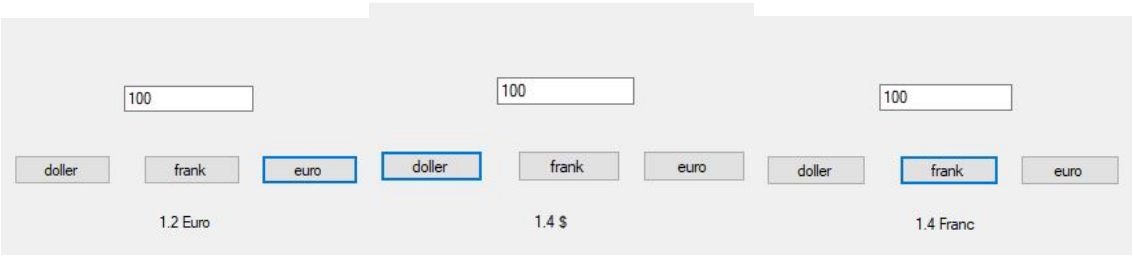
PART -2 WINDOWS FORMS

AIM 1:**Create a Windows Form to Convert following currency conversion. Rupees to dollar, frank, euro.**

PROGRAM:

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

namespace GDI_
{
   public partial class Practical1 : Form
   {
      public Practical1()
      {
         InitializeComponent();
      }

      private void doller_Click(object sender, EventArgs e)
      {
         double doller_price = Convert.ToDouble(textBox1.Text) * 0.014;
         label1.Text = Convert.ToString(doller_price)+" $";
      }

      private void Practical1_Load(object sender, EventArgs e)
      {
         textBox1.Focus();
      }

      private void frank_Click(object sender, EventArgs e)
      {
         double doller_price = Convert.ToDouble(textBox1.Text) * 0.014;
         label1.Text = Convert.ToString(doller_price)+" Franc";
      }

      private void euro_Click(object sender, EventArgs e)
      {
         double doller_price = Convert.ToDouble(textBox1.Text) * 0.012;
         label1.Text = Convert.ToString(doller_price)+" Euro";
      }
   }
}
```

OUTPUT:



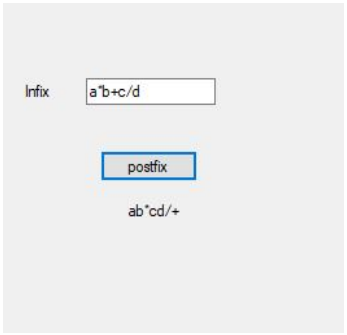
2 : **Create a Windows Form to convert infix notation to postfix notation. Ex: a+b-c =>ab+c-**

PROGRAM:

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

namespace Practical_2_2
{
   public partial class Form1 : Form
   {
      public Form1()
      {
         InitializeComponent();
      }

      private void button1_Click(object sender, EventArgs e)
      {
         string infix=textBox1.Text;
         string postfix = convert(infix);
         label1.Text = postfix;
      }
      static string convert( string infix )
      {
         string postfix;
         int prio = 0;
         postfix = "";
         Stack<Char> s1 = new Stack<char>();
         for (int i = 0; i < infix.Length; i++)
         {
            char ch = infix[i];
            if (ch == '+' || ch == '-' || ch == '*' || ch == '/')
            {
               if (s1.Count <= 0)
                  s1.Push(ch);
               else
               {
                  if (s1.Peek() == '*' || s1.Peek() == '/')
```

```
                    prio = 1;
                else
                    prio = 0;
                if (prio == 1)
                {
                    if (ch == '+' || ch == '-')
                    {
                        postfix += s1.Pop();
                        i--;
                    }
                    else
                    {
                        postfix += s1.Pop();
                        i--;
                    }
                }
                else
                {
                    if (ch == '+' || ch == '-')
                    {
                        postfix += s1.Pop();
                        s1.Push(ch);

                    }
                    else
                        s1.Push(ch);
                }
            }
        }
        else
        {
            postfix += ch;
        }
    }
    int len = s1.Count;
    for (int j = 0; j < len; j++)
        postfix += s1.Pop();
    return postfix;
  }
 }
}
```

OUTPUT:



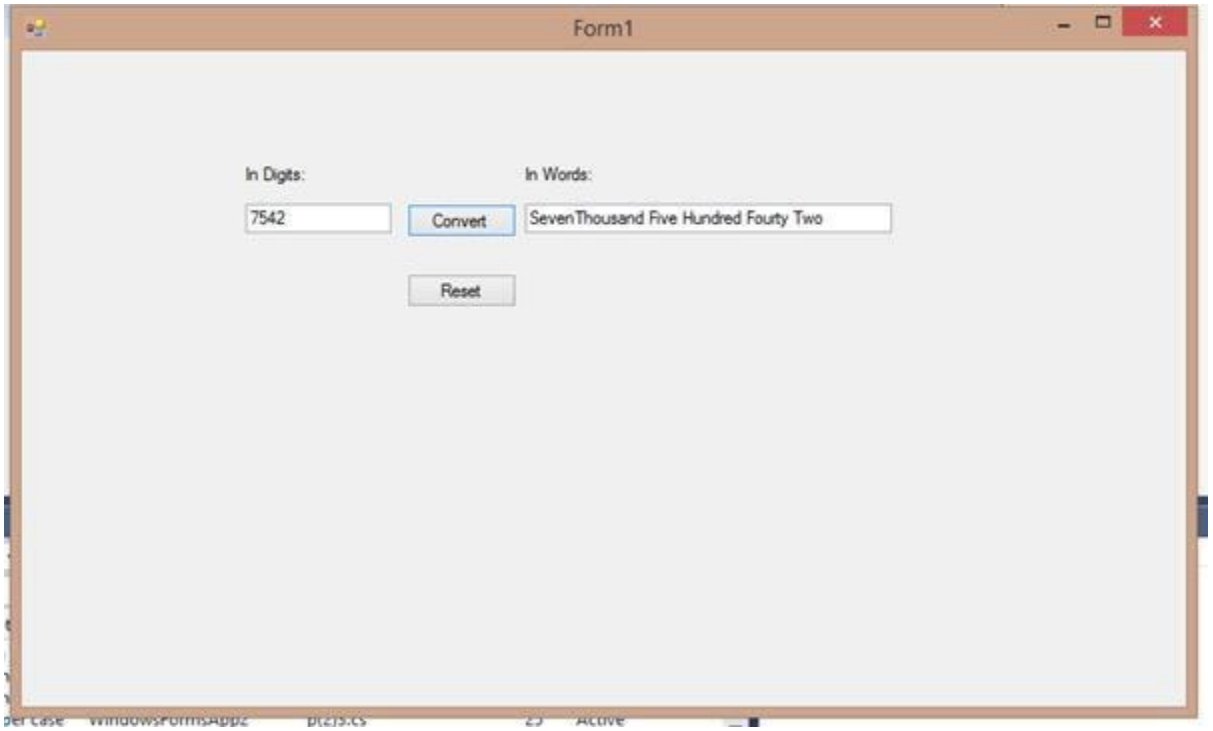
AIM 3 : **Create a Windows Form to convert digits to words Ex: 123 =>One Hundred and Twenty-three.**

PROGRAM:

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

namespace WindowsFormsApplication1
{
   public partial class Form1 : Form
   {
      int a = 0;
      int number;
      int num;
      int[] dg = new int[4]{0,0,0,0};
      string str;
      string[] digits = new
string[10]{"zero","one","two","three","four","five","six","seven","eight","nine"};
      public Form1()
      {
         InitializeComponent();
      }

      private void textBox1_TextChanged(object sender, EventArgs e)
      {

      }

      private void button1_Click(object sender, EventArgs e)
      {
         int i=3;
         number = Convert.ToInt32(textBox1.Text);
         num = number;
         while (num != 0)
         {
            dg[i] = num%10;
            num = num / 10;
            i--;
         }
         for (; a < 4; a = a + 1)
         {
            if (a == 0)
            {
```

```
                str = digits[dg[a]] + " thousand ";
            }
            if (a == 1)
            {
                str = str + digits[dg[a]] + " hundred and ";
            }
            if (a == 2)
            {
                str = str + digits[dg[a]] + "ty-";
            }
            if (a == 3)
            {
                str = str + digits[dg[a]];
            }
        }
        textBox2.Text = str;
    }

    private void label2_Click(object sender, EventArgs e)
    {

    }

    private void textBox2_TextChanged(object sender, EventArgs e)
    {

    }
  }
}
```

OUTPUT:



AIM 4 : **Create a Windows Form to implement Calculator Application**

PROGRAM:

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

namespace cal
{
   public partial class Form1 : Form
   {
      Double result = 0;
      String operation = "";
      bool enter_value = false;

      public Form1()
      {
         InitializeComponent();
      }

      private void standardToolStripMenuItem_Click(object sender, EventArgs
      e) {
         this.Width = 250;
         textBox1.Width = 218;
      }

      private void scToolStripMenuItem_Click(object sender, EventArgs
      e) {
         this.Width = 581;
         textBox1.Width = 475;
      }

      private void Form1_Load(object sender, EventArgs e)
      {
         this.Width = 250;
         textBox1.Width = 218;
      }

      private void button_Click(object sender, EventArgs e)
      {
         if ((textBox1.Text == "0") || (enter_value))
         {
            textBox1.Clear();
```

```
    }
    enter_value = false;

    Button button = (Button)sender;
    if (button.Text == ".")
    {
        if (!(textBox1.Text.Contains(".")))
        {
            textBox1.Text = textBox1.Text + button.Text;
        }
    }
    else
        textBox1.Text = textBox1.Text + button.Text;
}

private void button2_Click(object sender, EventArgs e)
{
    textBox1.Text = "0";
    label1.Text = "";
}

private void button3_Click(object sender, EventArgs e)
{
    textBox1.Text = "0";
    label1.Text = "";
    result = 0;
}

private void button1_Click(object sender, EventArgs e)
{
    if (textBox1.Text.Length > 0)
    {
        textBox1.Text = textBox1.Text.Remove(textBox1.Text.Length-1,1);
    }
    if (textBox1.Text== "0")
    {
        textBox1.Text = "0";
    }
}

private void operator_click(object sender, EventArgs e)
{
    Button button = (Button)sender;
    if (result != 0)
    {
        EQUAL.PerformClick();
        operation = button.Text;
        label1.Text = result + "" + operation;
        enter_value= true;
    }
```

```
        else
        {
            operation = button.Text;
            result = Convert.ToDouble(textBox1.Text);
            label1.Text = result + "" + operation;
            enter_value = true;
        }
    }

    private void button18_Click(object sender, EventArgs e)
    {
        label1.Text = "";
        switch (operation)
        {
            case "+":
                textBox1.Text = Convert.ToString(result
+ Convert.ToDouble(textBox1.Text));
                break;
            case "-":
                textBox1.Text = Convert.ToString(result
- Convert.ToDouble(textBox1.Text));
                break;
            case "*":
                textBox1.Text = Convert.ToString(result *
Convert.ToDouble(textBox1.Text));
                break;
            case "/":
                textBox1.Text = Convert.ToString(result
/ Convert.ToDouble(textBox1.Text));
                break;
            case "Mod":
                textBox1.Text = Convert.ToString(result %
Convert.ToDouble(textBox1.Text));
                break;
            case "Exp":
                double i=Convert.ToDouble(textBox1.Text);
                double q;
                q = (result);
                textBox1.Text = Convert.ToString(Math.Exp(i * Math.Log(q *
                4))); break;
            default:
                break;
        }
        result = Convert.ToDouble(textBox1.Text);
        label1.Text = "";
    }

    private void button40_Click(object sender, EventArgs e)
    {
        textBox1.Text = "3.141592653589976323";
```

```
}

private void button39_Click(object sender, EventArgs e)
{
   double ilog = Convert.ToDouble(textBox1.Text);
   label1.Text = Convert.ToString("log " + "( " + (textBox1.Text) + "
   )"); ilog= Math.Log10(ilog);
   textBox1.Text = Convert.ToString(ilog);
}

private void button38_Click(object sender, EventArgs e)
{
   double sq = Convert.ToDouble(textBox1.Text);
   label1.Text = Convert.ToString("sqrt " + "( " + (textBox1.Text) + "
   )"); sq = Math.Sqrt(sq);
   textBox1.Text = Convert.ToString(sq);
}

private void button36_Click(object sender, EventArgs e)
{
   double sinh = Convert.ToDouble(textBox1.Text);
   label1.Text = Convert.ToString("Sinh " + "( " + (textBox1.Text) + " )");
   sinh = Math.Sinh(sinh);
   textBox1.Text = Convert.ToString(sinh);
}

private void button35_Click(object sender, EventArgs e)
{
   double sin = Convert.ToDouble(textBox1.Text);
   label1.Text = Convert.ToString("Sin " + "( " + (textBox1.Text) + " )");
   sin = Math.Sin(sin);
   textBox1.Text = Convert.ToString(sin);
}

private void button32_Click(object sender, EventArgs e)
{
   double cosh = Convert.ToDouble(textBox1.Text);
   label1.Text = Convert.ToString("Cosh " + "( " + (textBox1.Text) + " )");
   cosh = Math.Cosh(cosh);
   textBox1.Text = Convert.ToString(cosh);
}

private void button31_Click(object sender, EventArgs e)
{
   double cos = Convert.ToDouble(textBox1.Text);
   label1.Text = Convert.ToString("Cos " + "( " + (textBox1.Text) + " )");
   cos = Math.Cos(cos);
   textBox1.Text = Convert.ToString(cos);
}
```

```
private void button28_Click(object sender, EventArgs e)
{
   double tanh = Convert.ToDouble(textBox1.Text);
   label1.Text = Convert.ToString("Tanh " + "( " + (textBox1.Text) + " )");
   tanh = Math.Tanh(tanh);
   textBox1.Text = Convert.ToString(tanh);
}

private void button27_Click(object sender, EventArgs e)
{
   double tan = Convert.ToDouble(textBox1.Text);
   label1.Text = Convert.ToString("Tan " + "( " + (textBox1.Text) + "
   )"); tan = Math.Tan(tan);
   textBox1.Text = Convert.ToString(tan);
}

private void button30_Click(object sender, EventArgs e)
{
   int a = Convert.ToInt32(textBox1.Text);
   textBox1.Text = Convert.ToString(a, 2);
}

private void button26_Click(object sender, EventArgs e)
{
   int a = Convert.ToInt32(textBox1.Text);
   textBox1.Text = Convert.ToString(a, 16);
}

private void button22_Click(object sender, EventArgs e)
{
   int a = Convert.ToInt32(textBox1.Text);
   textBox1.Text = Convert.ToString(a, 8);
}

private void button34_Click(object sender, EventArgs e)
{
   int a = Convert.ToInt32(textBox1.Text);
   textBox1.Text = Convert.ToString(a);
}

private void button37_Click(object sender, EventArgs e)
{
   double a;
   a = Convert.ToDouble(textBox1.Text) *
   Convert.ToDouble(textBox1.Text); textBox1.Text = Convert.ToString(a);
}

private void button33_Click(object sender, EventArgs e)
{
   double a;
```

```
        a = Convert.ToDouble(textBox1.Text) * Convert.ToDouble(textBox1.Text) *
Convert.ToDouble(textBox1.Text);
        textBox1.Text = Convert.ToString(a);
    }

    private void button29_Click(object sender, EventArgs e)
    {
        double a;
        a = Convert.ToDouble(1.0 /
        Convert.ToDouble(textBox1.Text)); textBox1.Text =
        Convert.ToString(a);
    }

    private void button25_Click(object sender, EventArgs e)
    {
        double ln = Convert.ToDouble(textBox1.Text);
        label1.Text = Convert.ToString("ln " + "( " + (textBox1.Text) + " )");
        ln = Math.Log(ln);
        textBox1.Text = Convert.ToString(ln);
    }

    private void button21_Click(object sender, EventArgs e)
    {
        double a;
        a = Convert.ToDouble(textBox1.Text) *
        Convert.ToDouble(100); textBox1.Text = Convert.ToString(a);
    }
  }
}
```

OUTPUT:

AIM 5 : **To drag and drop item from one textbox to other textbox.**

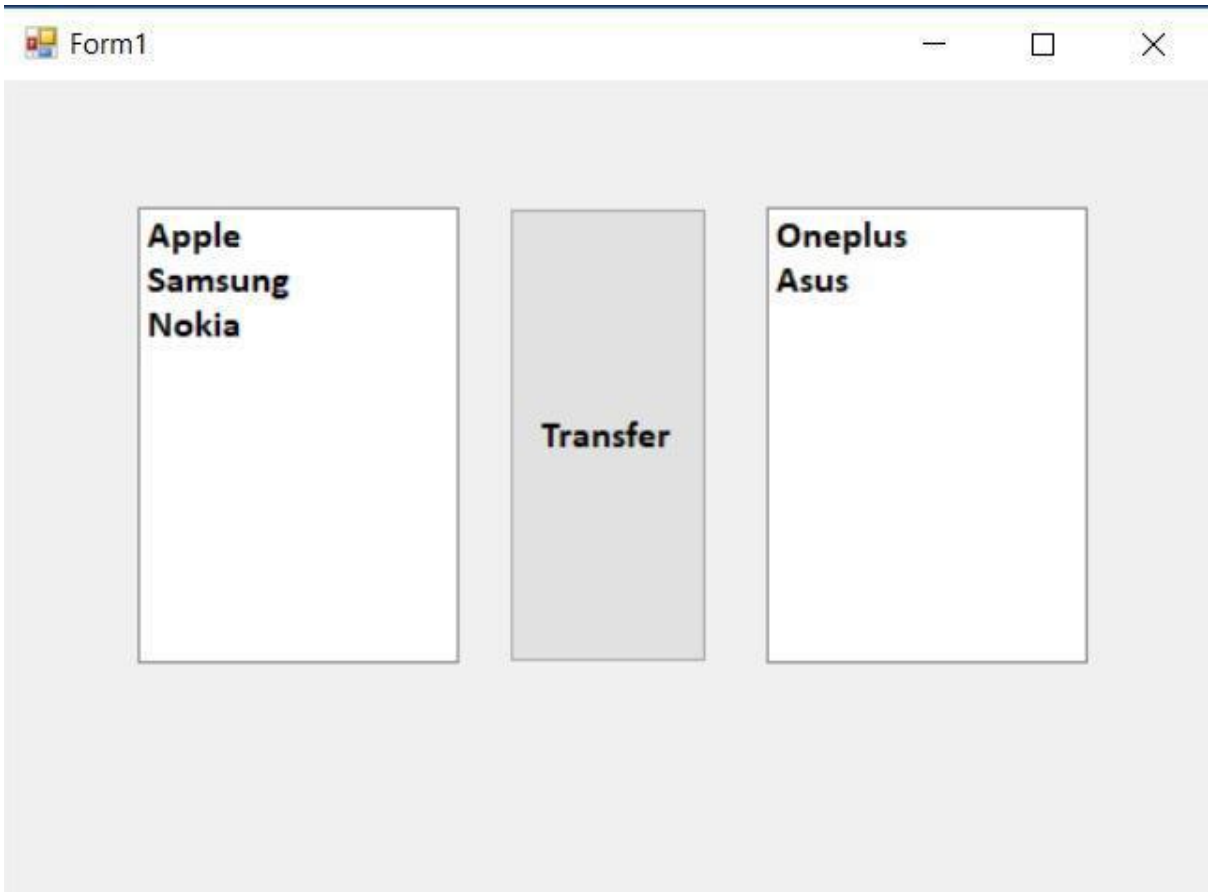
PROGRAM:

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

namespace WindowsFormsApplication1
{
   public partial class Windows5 : Form
   {
      int index;
      public Windows5()
      {
         InitializeComponent();
      }

      private void Windows5_Load(object sender, EventArgs e)
      {

         listBox1.Items.Add("Apple");
         listBox1.Items.Add("Orange");
         listBox1.Items.Add("Pineapple");
         listBox1.Items.Add("Kiwi");

         listBox2.Items.Add("Red Rose");
         listBox2.Items.Add("Sunflower");
         listBox2.Items.Add("White Rose");
         listBox2.Items.Add("Tulips");
      }

      private void button1_Click(object sender, EventArgs e)
      {
         for(int i = 0; i < listBox1.SelectedItems.Count; i++)
         {
            listBox2.Items.Add(listBox1.SelectedItems[i].ToString());
         }
         for(int i = listBox1.SelectedItems.Count-1; i>=0; i--)
         {
```

```
        if (listBox1.GetSelected(i) == true)
        {
            listBox1.Items.RemoveAt(i);
        }
    }
}

private void button2_Click(object sender, EventArgs e)
{
    for (int i = 0; i < listBox2.SelectedItems.Count; i++)
    {
        listBox1.Items.Add(listBox2.SelectedItems[i].ToString());
    }
    for (int i = listBox2.Items.Count - 1; i >= 0; i--)
    {
        if (listBox2.GetSelected(i) == true)
        {
            listBox2.Items.RemoveAt(i);
        }
    }
}

private void listBox1_MouseDown(object sender, MouseEventArgs
e) {
    ListBox listbox = (ListBox)sender;
    index = listbox.IndexFromPoint(e.X, e.Y);
    listbox.DoDragDrop(listbox.Items[index],
    DragDropEffects.Move);
}

private void listBox1_DragOver(object sender, DragEventArgs e)
{
    if (e.Data.GetDataPresent(DataFormats.StringFormat))
    {
        e.Effect = DragDropEffects.Move;
    }
    else
    {
        e.Effect = DragDropEffects.None;
    }
}

private void listBox1_DragDrop(object sender, DragEventArgs e)
{
    ListBox listBox = (ListBox)sender;
```

```
            listBox.Items.Add(e.Data.GetData(DataFormats.StringFormat).ToString());
            if (listBox.Name == "listBox1")
            {
                listBox2.Items.RemoveAt(index);
            }
            else
            {
                listBox1.Items.RemoveAt(index);
            }
        }
    }
}
```

OUTPUT:



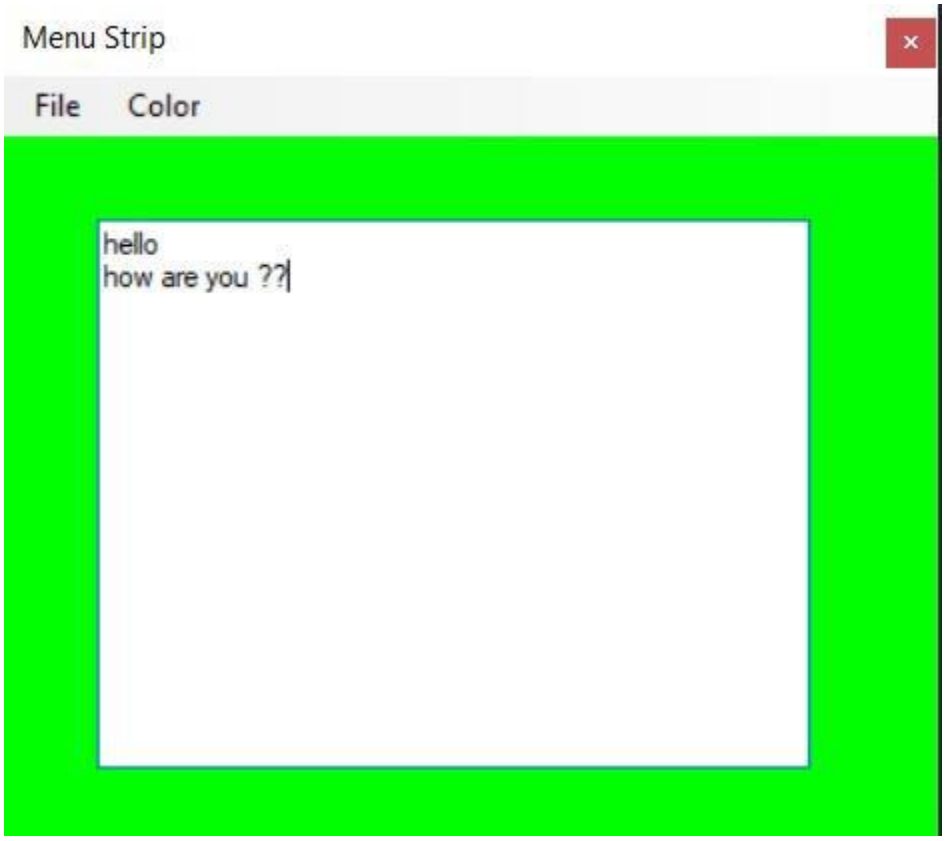
AIM 6 : **Create a windows form which implements Menus (File, Color) and a Text Box. File menu item has 4 options Fonts, Open, Save, Print (Print Preview and Print), Exit. Color menu opens a Color Dialog box. Clicking on the colors sets the background of the form with respective color. Open menu item opens a file, reads the contents and displays in the TextBox while Save menu items saves the content of Text Box in to a file.**

PROGRAM:

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

namespace WindowsFormsApplication1
{
   public partial class Form1 : Form
   {
        public Form1()
        {
                InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {

        }
        private void newToolStripMenuItem_Click(object sender, EventArgs e)
        {
                richTextBox1.Clear();
        }
        private void openToolStripMenuItem_Click(object sender, EventArgs e)
        {
                if (openFileDialog1.ShowDialog() == DialogResult.OK)
                {
                StreamReader sr = new
                StreamReader(openFileDialog1.FileName); richTextBox1.Text =
                sr.ReadToEnd(); sr.Close();
                }
        }
        private void saveToolStripMenuItem_Click(object sender, EventArgs e)
        {
                saveFileDialog1.FileName = "unknown.txt";
                saveFileDialog1.Filter = "text file(*.txt)|*.txt|All files(*.*)|*.*";
```

```
            if (saveFileDialog1.ShowDialog()== DialogResult.OK)
            {
            StreamWriter sw = new StreamWriter(saveFileDialog1.FileName)
            {
                  sw.WriteLine(richTextBox1.Text);
            }
            }
      }
      private void printToolStripMenuItem_Click(object sender, EventArgs e)
      {
            printDialog1.Document = printDocument1;
            if (printDialog1.ShowDialog() == DialogResult.OK)
            {
            printDocument1.Print();
            }
      }
      private void printPreviewToolStripMenuItem_Click(object sender, EventArgs
e)
      {
            printPreviewDialog1.Document =
            printDocument1;
            printPreviewDialog1.ShowDialog();
      }
      private void fontToolStripMenuItem_Click(object sender, EventArgs e)
      {
            if (richTextBox1.SelectedText != "")
            {
            richTextBox1.Select();
            if (fontDialog1.ShowDialog() == DialogResult.OK)
            {
                  richTextBox1.Font = fontDialog1.Font;
            }
            }
      }
      private void exitToolStripMenuItem_Click(object sender, EventArgs e)
      {
            Application.Exit();
      }
      private void colorToolStripMenuItem_Click(object sender, EventArgs e)
      {
            ColorDialog cd = new ColorDialog();
            ColorDialog cd1 = new ColorDialog();
            if (cd.ShowDialog() == DialogResult.OK)
            {
            BackColor = cd.Color;
            }
            if (cd1.ShowDialog() == DialogResult.OK)
            {
            richTextBox1.ForeColor = cd1.Color;
            }}}}
```

OUTPUT:



AIM 7 : **Implement an MDI form with Menu item “Add”, clicking on Add generates a new childform(blank). And Menu item “Child Forms” which contains sub menu for implementingError Provider, Tool tip, Panel and Group Box controls each on a separate form.**

PROGRAM:

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

namespace Prac7
{
   public partial class Form1 : Form
   {
      public Form1()
      {
        InitializeComponent();
      }
      private void addChildToolStripMenuItem_Click(object sender, EventArgs e)
      {
        Form f = new Form();
        f.Text = "Child";
        f.MdiParent = this;
        f.Show();
      }
      private void errorProviderToolStripMenuItem_Click(object sender, EventArgs e)
      {
        Form f = new error();
        f.MdiParent = this;
        f.Show();
      }
      private void groupBoxToolStripMenuItem_Click(object sender, EventArgs e)
      {
        Form f = new GB();
        f.MdiParent = this;
        f.Show();
      }
      private void panelDemoToolStripMenuItem_Click(object sender, EventArgs e)
```

```
        {
          Form f = new panel();
          f.MdiParent = this;
          f.Show();
        }
        private void toolTipToolStripMenuItem_Click(object sender, EventArgs e)
        {
          Form f = new tooltip();
          f.MdiParent = this;
          f.Show();
        }
    }
}
```

//Error Provider

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

namespace Prac7
{
    public partial class error : Form
    {
        public error()
        {
          InitializeComponent();
        }

        private void textBox1_TextChanged(object sender, EventArgs e)
        {
          if (textBox1.Text.Length > 10)
          {
          errorProvider1.SetError(textBox1, "Error");
          }
        }
    }
}
```

//Group Box

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

namespace Prac7
{
   public partial class GB : Form
   {
      public GB()
      {
        InitializeComponent();
      }

      private void GroupBox_Load(object sender, EventArgs e)
      {
        groupBox1.Text = "Select Color : ";
        this.BackColor = Color.AliceBlue;
        radioButton1.Text = "Red";
        radioButton2.Text = "Blue";
        radioButton3.Text = "Green";
      }
      private void radioButton1_CheckedChanged(object sender, EventArgs e)
      {
        this.BackColor = Color.Red;
      }

      private void radioButton2_CheckedChanged(object sender, EventArgs
      e) {
        this.BackColor = Color.Blue;
      }

      private void radioButton3_CheckedChanged(object sender, EventArgs
      e) {
        this.BackColor = Color.Green;
      }
   }
}
```

//Panel

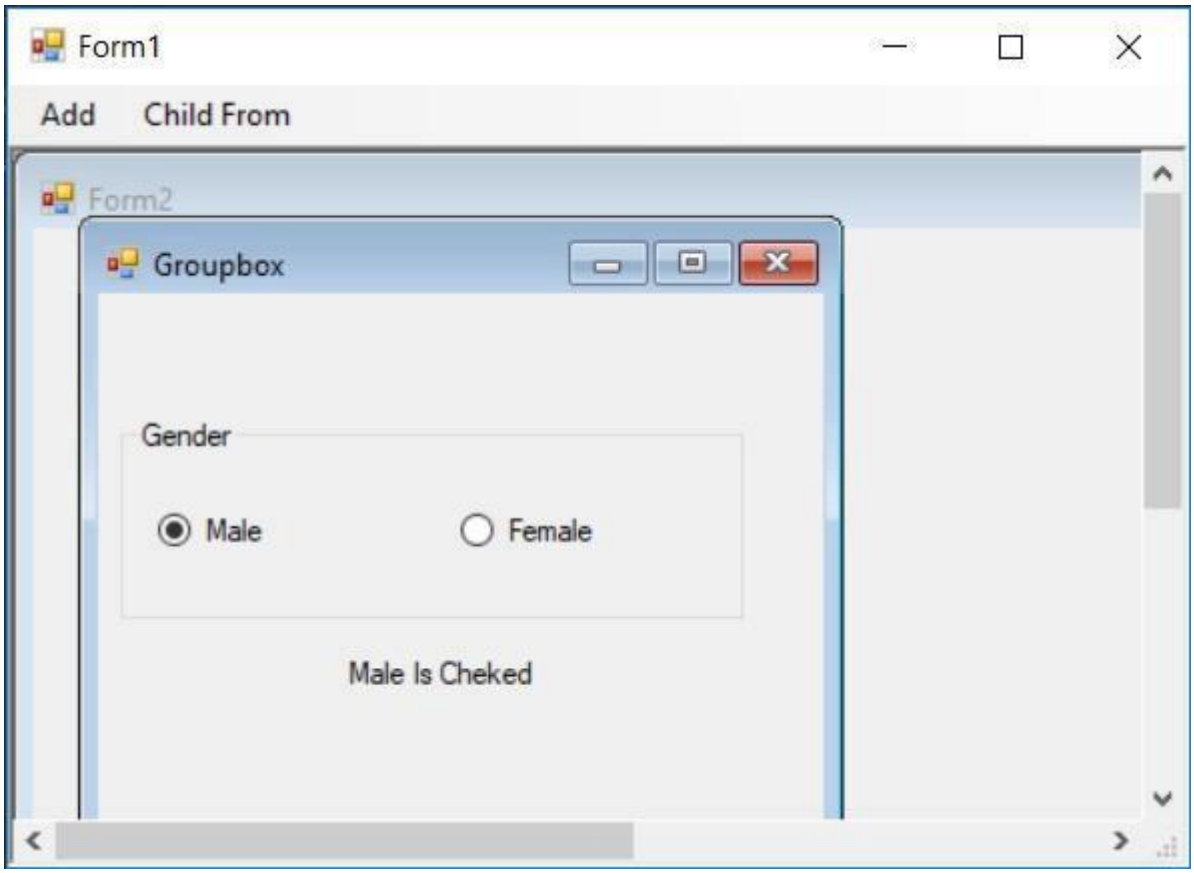
```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace Prac7
{
   public partial class panel : Form
   {
      public panel()
      {
         InitializeComponent();
      }
      private void panel_Load(object sender, EventArgs e)
      {
        checkBox1.Text = "RED";
        checkBox2.Text = "GREEN";
        checkBox3.Text = "BLUE";
        checkBox4.Text = "YELLOW";
      }
      private void checkBox1_CheckedChanged(object sender, EventArgs e)
      {
        if (checkBox1.Checked)
        {
        BackColor = Color.Red;
        }
      }
      private void checkBox2_CheckedChanged(object sender, EventArgs e)
      {
        if (checkBox2.Checked)
        {
        BackColor = Color.Green;
        }
      }
      private void checkBox3_CheckedChanged(object sender, EventArgs e)
      {
        if (checkBox3.Checked)
        {
        BackColor = Color.Blue;
```

```
        }
      }

      private void checkBox4_CheckedChanged(object sender, EventArgs
      e) {
        if (checkBox4.Checked)
        {
        BackColor = Color.Yellow;
        }
      }
  }
}
```

//ToolTip

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
namespace Prac6
{
  public partial class tooltip : Form
  {
    public tooltip()
    {
      InitializeComponent();
    }
    private void tooltip_Load(object sender, EventArgs e)
    {
      ToolTip buttonToolTip = new ToolTip();
      buttonToolTip.ToolTipTitle = "Button Tooltip";
      buttonToolTip.ShowAlways = true;
      buttonToolTip.SetToolTip(button1, "Click me to execute..");

      ToolTip textboxToolTip = new ToolTip();
      textboxToolTip.ToolTipTitle = "Text Box Tooltip";
      textboxToolTip.SetToolTip(textBox1, "Enter
      something..");
    }
  }
}
```

OUTPUT:


Form1
Add
Child From
Form2
Groupbox
Gender
Male
Female
Male Is Cheked

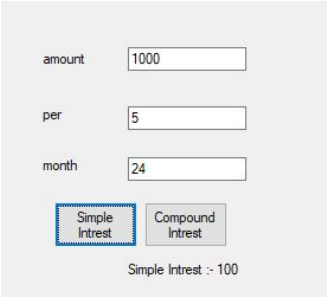
AIM 8 : **Create a custom control for calculating Simple Interest and Compound Interest.**

PROGRAM:

Program.cs

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

namespace Intrest
{
   public partial class UserControl1 : UserControl
   {
      public UserControl1()
      {
         InitializeComponent();
      }

      private void button1_Click(object sender, EventArgs e)
      {
         double amount, per, month;
         amount = Convert.ToDouble(text_amount.Text);
         per = Convert.ToDouble(text_per.Text);
         month = Convert.ToDouble(text_month.Text);
         double intrest;
         intrest = amount * per * month / (100*12);
         label4.Text = "Simple Intrest :- " + intrest.ToString();
      }

      private void button2_Click(object sender, EventArgs e)
      {
         double amount, per, month;
         amount = Convert.ToDouble(text_amount.Text);
         per = Convert.ToDouble(text_per.Text);
         month = Convert.ToDouble(text_month.Text);
         double cintrest;
         cintrest=amount;
         for (int i = 0; i < month/12; i++)
         {
            amount+=(double)amount*per*month/(100*12);
         }
         cintrest = amount - cintrest;
         label4.Text = "Compound Intrest :- " + cintrest.ToString();
      }
   }
}
```

OUTPUT:



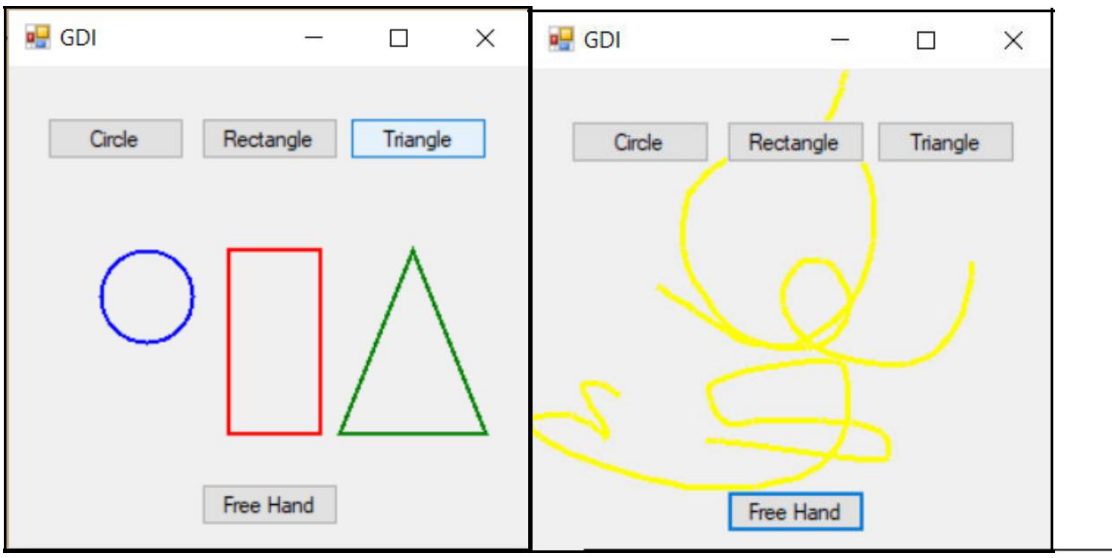
AIM 9 : **Implement Paint application to draw various shapes using GDI+. Use menu to provide various options. Free Hand Drawing should also be implemented.**

PROGRAM:

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

namespace P9
{
   public partial class Form1 : Form
   {
      Pen pen;
      Graphics g;
      float x, y;
int flag=0;
      public Form1()
      {
InitializeComponent();
         g = this.CreateGraphics();
      }

      private void button1_Click(object sender, EventArgs e)
      {
         Graphics g = this.CreateGraphics();
         pen = new Pen(Color.Blue, 2);
g.DrawEllipse(pen, 50, 100, 50, 50);
      }

      private void button2_Click(object sender, EventArgs e)
      {
         pen = new Pen(Color.Red, 2);
g.DrawRectangle(pen, 120, 100, 50, 100);
      }
```

```
        private void button3_Click(object sender, EventArgs e)
        {
            Pen = new Pen(Color.Green, 2);
PointF pt1 = new PointF(220, 100);
PointF pt2 = new PointF(180, 200);
PointF pt3 = new PointF(260, 200);
PointF[] ptsArray =
            {
                pt1, pt2, pt3
            };
g.DrawPolygon(pen, ptsArray);
        }

        private void button4_Click(object sender, EventArgs e)
        {
            flag = 1;
            pen = new Pen(Color.Yellow, 3);
        }

        private void Form1_MouseMove(object sender, MouseEventArgs e)
        {
            if (e.Button == MouseButtons.Left&& flag==1)
            {
g.DrawLine(pen, x, y, e.X, e.Y);
                x = e.X;
                y = e.Y;
            }
        }

        private void Form1_MouseDown(object sender, MouseEventArgs e)
        {
            x = e.X;
            y = e.Y;
        }

        private void Form1_MouseUp(object sender, MouseEventArgs e)
        {
            flag = 0;
        }
    }
}
```

OUTPUT:


GDI
Circle
Rectangle
Triangle
Free Hand
GDI
Circle
Rectangle
Triangle
Free Hand