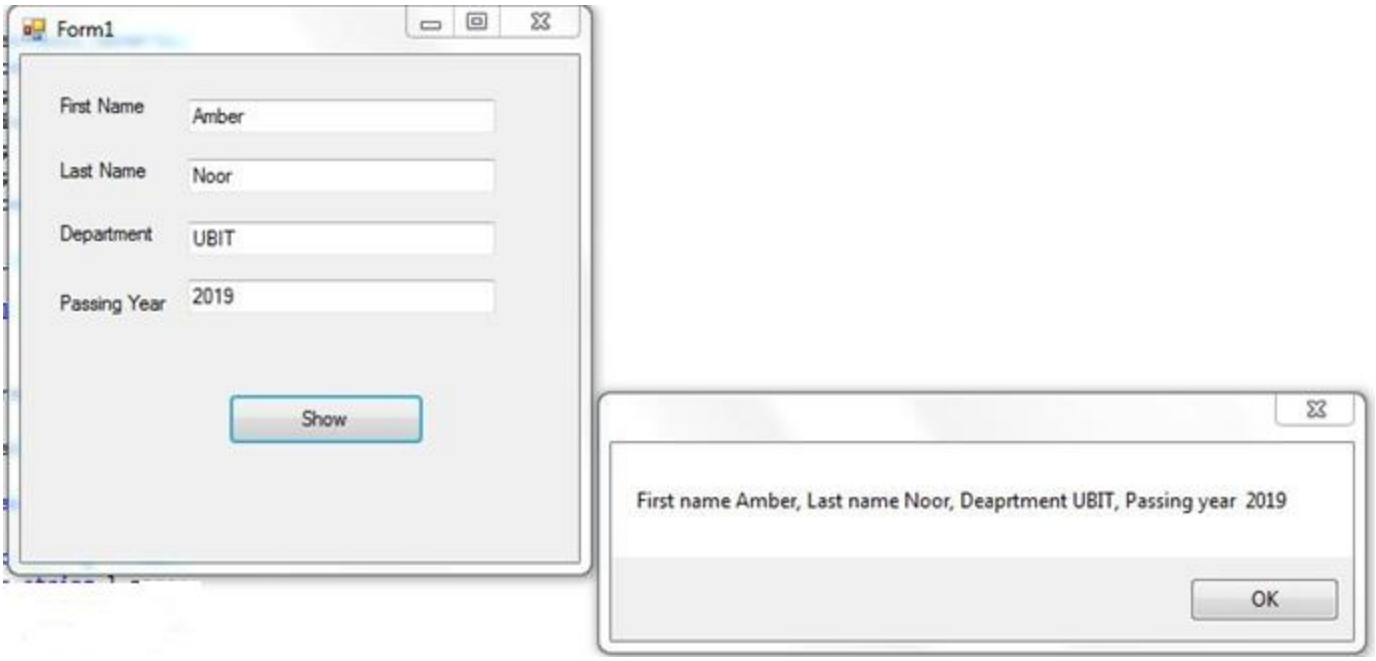
LAB # 04

Create and use of a class and its attributes:

```
public class student
        {
            public string f_name;
            public string l_name;
            public string dept;
            public int pass_year;
        }


        private void button1_Click(object sender, EventArgs e)
        {
            student stu = new student();
            stu.f_name = textBox1.Text;
            stu.l_name = textBox2.Text;
            stu.dept = textBox3.Text;
            stu.pass_year = int.Parse(textBox4.Text);
            MessageBox.Show(" First name " + stu.f_name + "," + " Last name " +
                stu.l_name + "," + " Deaprtment " + stu.dept +
                                "," +   " Passing year  " + stu.pass_year);
        }
    }
}
```

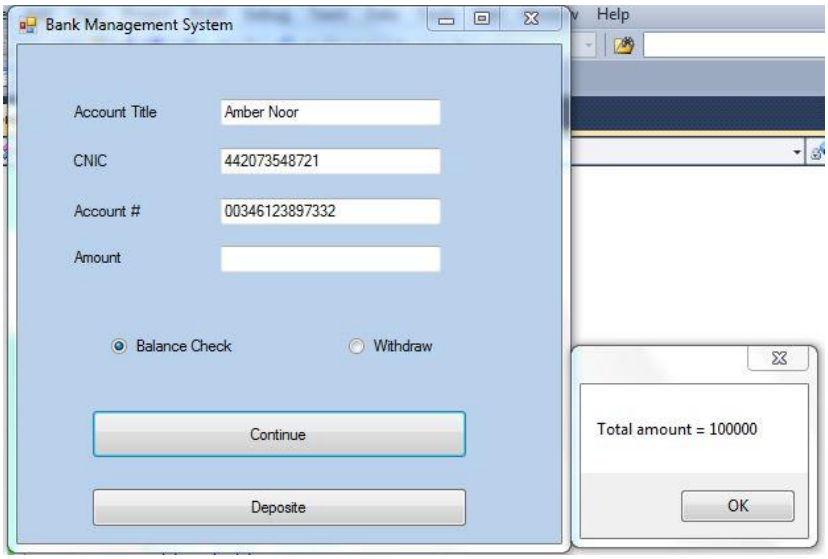


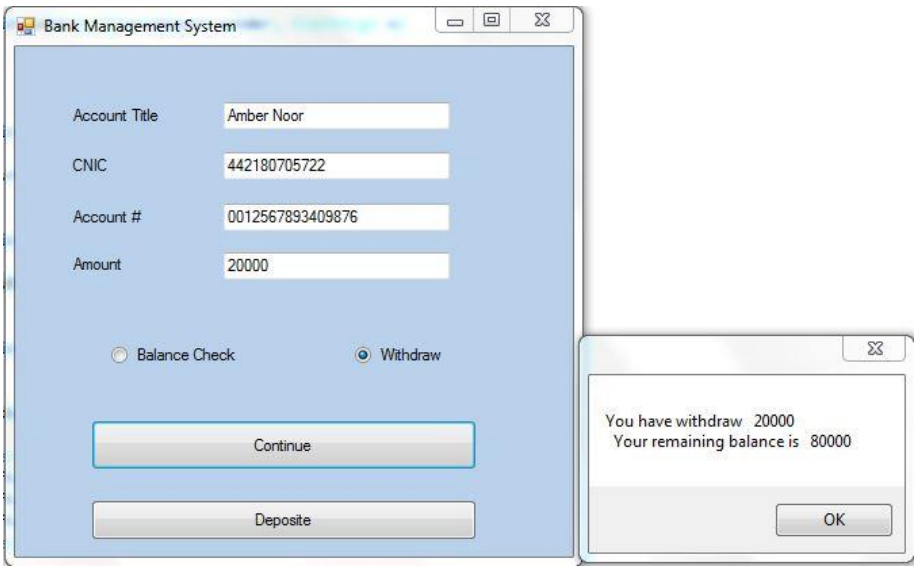
LAB # 05

Bank Management System:

```
public class UserInfo
        {
            public string acc_title;
            public double user_id;
            public double acc_num;
        }
        public class Withdraw
        {
            public double amount;
            public double remain;
        }

        double balance = 100000;


        private void label2_Click(object sender, EventArgs e)
        {

        }

        private void radioButton1_CheckedChanged(object sender, EventArgs e)
        {
            radioButton1.Checked = true;
        }

        private void radioButton2_CheckedChanged(object sender, EventArgs e)
        {
            radioButton2.Checked = true;
        }

        private void button1_Click(object sender, EventArgs e)
        {

            if (radioButton1.Checked == true)
            {
                UserInfo Info = new UserInfo();
                Info.acc_title = textBox1.Text;
                Info.user_id = double.Parse(textBox2.Text);
                Info.acc_num = double.Parse(textBox3.Text);
                 MessageBox.Show("Total amount = " + balance);
            }

            else if (radioButton2.Checked == true)
            {
                UserInfo Info = new UserInfo();
                Info.acc_title = textBox1.Text;
                Info.user_id = double.Parse(textBox2.Text);
                Info.acc_num = double.Parse(textBox3.Text);

                Withdraw rupee = new Withdraw();
                rupee.amount = double.Parse(textBox4.Text);
                rupee.remain = balance - rupee.amount;
                MessageBox.Show("You have withdraw " + "  "+ rupee.amount  + "\n" + " " +
                        "Your remaining balance is " + "  " + rupee.remain);

            }

        }
```

```
    private void button2_Click(object sender, EventArgs e)
    {

            double deposite = double.Parse(textBox4.Text);
            balance = balance + deposite;
            MessageBox.Show("You deposite" + " " + balance + "\n" + " Your total amount is" + " " +
            balance);

        }
    }
}
```



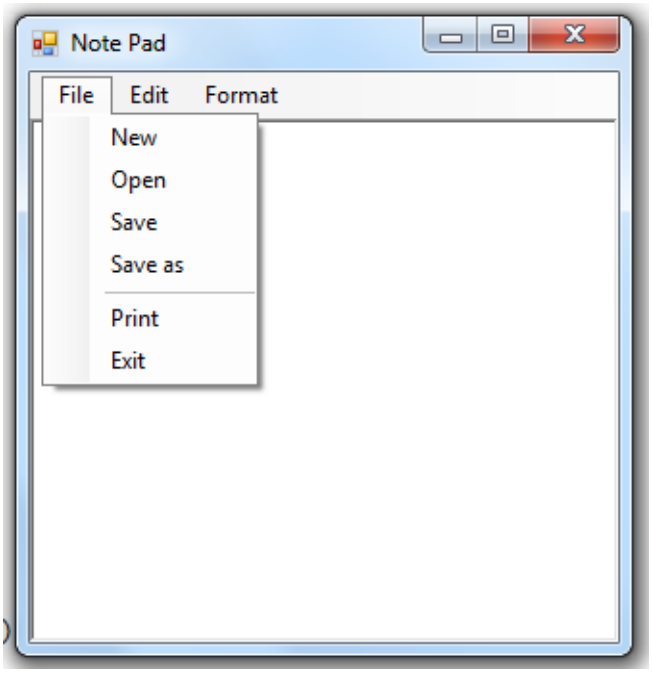


LAB # 06

Note Pad Window Application:

```
public Form1()
        {
            InitializeComponent();
        }

        private void newToolStripMenuItem_Click(object sender, EventArgs e)
        {

          MessageBox.Show(" Do you want save file","save", MessageBoxButtons.YesNo);
          DialogResult result = new DialogResult();
          if (result == DialogResult.Yes)
          {

                    richTextBox1.SaveFile(this.Text);

          }

          else
          {

              richTextBox1.Clear();
          }
        }

        private void openToolStripMenuItem_Click(object sender, EventArgs e)
        {
            openFileDialog1.ShowDialog();
            System.IO.StreamReader openfile = new System.IO.StreamReader(openFileDialog1.FileName);
            richTextBox1.Text = openfile.ReadToEnd();
            openfile.Close();
        }

        private void saveToolStripMenuItem_Click(object sender, EventArgs e)
        {
            System.IO.StreamWriter savefile = new StreamWriter(openFileDialog1.FileName);
            savefile.WriteLine(richTextBox1.Text);
            savefile.Close();
        }

        private void saveAsToolStripMenuItem_Click(object sender, EventArgs e)
        {
            saveFileDialog1.ShowDialog();
            System.IO.StreamWriter sfd = new StreamWriter(saveFileDialog1.FileName);
            sfd.WriteLine(richTextBox1.Text);
            sfd.Close();
        }

        private void printToolStripMenuItem_Click(object sender, EventArgs e)
        {
            System.Drawing.Printing.PrintDocument prntDoc = new
System.Drawing.Printing.PrintDocument();
        }

        private void exitToolStripMenuItem_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }
```

```
        private void undoToolStripMenuItem_Click(object sender, EventArgs e)
        {
            richTextBox1.Undo();
        }

        private void cutToolStripMenuItem_Click(object sender, EventArgs e)
        {
            richTextBox1.Cut();
        }

        private void copyToolStripMenuItem_Click(object sender, EventArgs e)
        {
            richTextBox1.Copy();
        }

        private void pasteToolStripMenuItem_Click(object sender, EventArgs e)
        {
            richTextBox1.Paste();
        }

        private void selectToolStripMenuItem_Click(object sender, EventArgs e)
        {
            richTextBox1.Clear();
        }

        private void deselectAllToolStripMenuItem_Click(object sender, EventArgs e)
        {
            richTextBox1.DeselectAll();
        }

        private void selectToolStripMenuItem1_Click(object sender, EventArgs e)
        {
            richTextBox1.Select();
        }

        private void wordWarpToolStripMenuItem_Click(object sender, EventArgs e)
        {
            richTextBox1.WordWrap = wordWarpToolStripMenuItem.Checked;
        }

        private void selectAllToolStripMenuItem_Click(object sender, EventArgs e)
        {
            richTextBox1.SelectAll();
        }

    }
}
```



LAB # 07

Learning constructors:

```
namespace constructors
{
    class car
    {
        public bool Whitecar;
        public bool Blackcar;

        //1. Default constructor
        public car()
        {
            Whitecar = true;
            Blackcar = true;
        }

        //2. Parameterized constructor
        public car(bool white, bool black)
        {
            Whitecar = white;
            Blackcar = black;
        }
        //(3) Copy Constructor
       public car (car mycar)
       {
            mycar.Whitecar = true;
            mycar.Blackcar = true;
       }

    }

    class Program
    {
        static void Main(string[] args)
        {
            bool display;
            car myCar = new car();
            display = myCar.Whitecar;
            display = myCar.Blackcar;
            Console.WriteLine(" White car=" + display +" " + "Black car=" + display);

            //(2) using parameterized contstructor.
            car mycar = new car(true, true);
            display = mycar.Blackcar;
            Console.WriteLine("White car is  " + display);

            //(3) using copy contstructor.
            car mycar3 = new car();
            display = mycar3.Whitecar;
            Console.WriteLine("White car is  " + display);


            Console.ReadLine();



        }
    }
}
```

```
 White car=True Black car=True
White car is  True
White car is  True
```

LAB # 08

Learning Inheritance:

```
namespace Inheritance
{
    class shapes
    {
        protected double Height;
        protected double Width;
        protected double Radius;
        protected double pi = 3.14;

        public void Setheight(double height)
        {
            Height = height;
        }

        public void Setwidth(double width)
        {
            Width = width;
        }
        public void Setradius(double radius)
        {
            Radius = radius;
        }
    }

    class square:shapes
    {
        // calculate area of square function
        public double getSqArea()
    {
        return(Height * Width);
    }

    }

    class rectangle: square
    {
        //calcualte area of rectangle
        public double getRect()
        {
            return (Height * Width);
        }
    }

    class circle:shapes
    {
        //calculate area of circle
        public double getCircle()
        {
            return (2* pi * (Radius * Radius));
        }
    }

    }
    class Program
    {
        static void Main(string[] args)
        {

            //double h, w, c;
```

```
    double area;

        square sqr = new square();
        //Console.Write("Enter the height= ");
        //h = Console.Read();
        //Console.Write("Enter the width= ");
        //w = Console.Read();
        sqr.Setheight(5);
        sqr.Setwidth(5);
        area = sqr.getSqArea();
        Console.WriteLine(" The area of square is {0}cm", area);

        rectangle rect = new rectangle();
        rect.Setheight(15);
        rect.Setwidth(4.6);
        area = rect.getRect();
       Console.WriteLine(" The area of rectangle is {0}cm", area);

        circle cir = new circle();
        cir.Setradius(5.5);
        area = cir.getCircle();
        Console.WriteLine(" The area of circle is {0}cm", area);
    Console.ReadLine();
    }


}
```