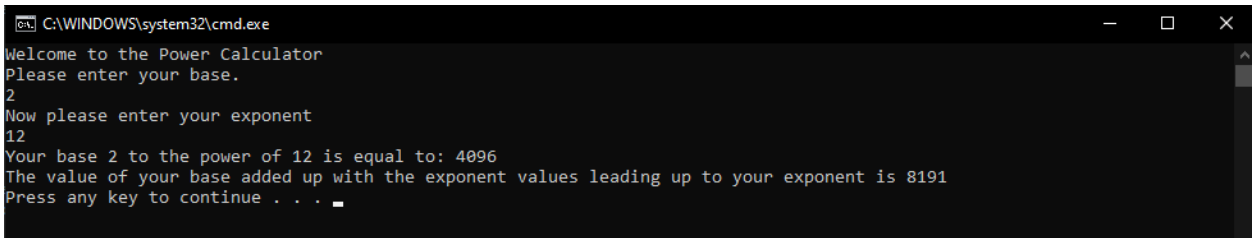
1.

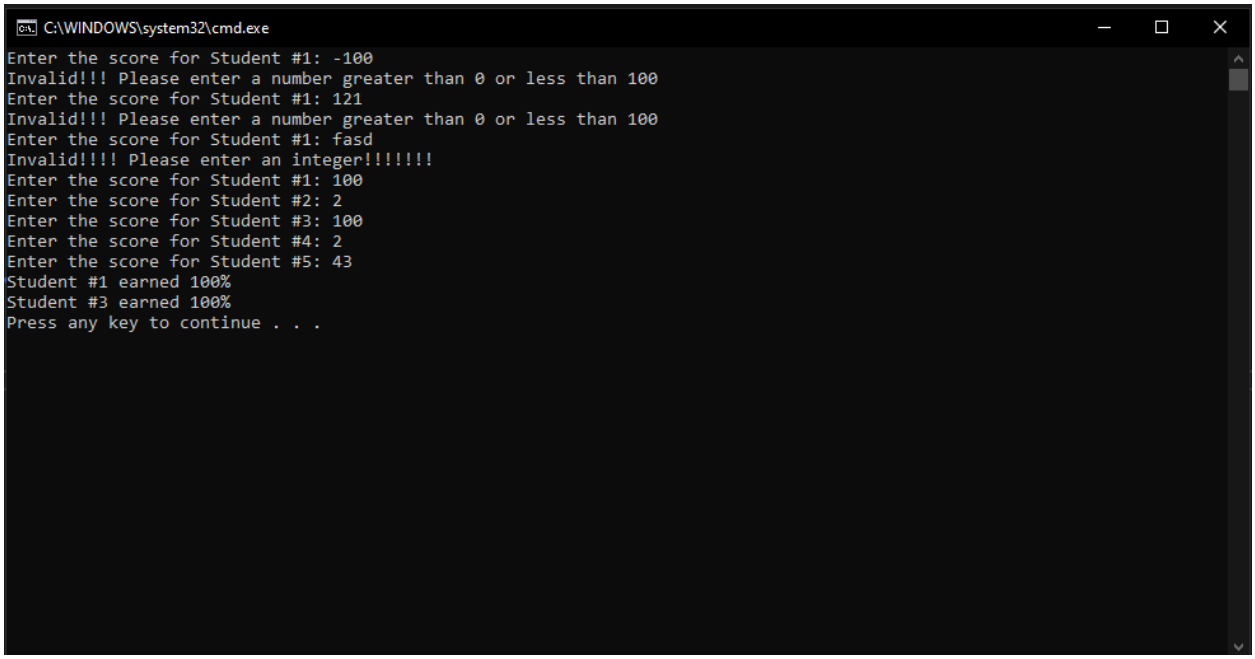



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace HW_7_Problem_1
{
   class Program
   {

      static int GetInput(string x, int low)
      {
         bool valid = true;

         valid = int.TryParse(x, out int y);
         if (!valid)
         {
            Console.WriteLine("Invalid!!!! Please enter an integer!!!!!!!");
            y = -9;
         }
         else if (y <= low)
         {
            Console.WriteLine("Invalid!!! Please enter a number greater than " + low);
            y = -9;
```

```
    }
        return y;

}


static int intPow(int x, int y)
{
    int result = 1;

    for (int i = 1; i <= y; i++)
    {
        result *= x;
    }

    return result;
}

static int intFacPow(int x, int y)
{
    int result = 1;
    int factorialpow = 1;

    for (int i = 1; i <= y; i++)
    {
        result *= x;
        factorialpow += result;

    }

    return factorialpow;
}

static void Main(string[] args)
{



    Console.WriteLine("Welcome to the Power Calculator\nPlease enter your base.");
    int n = GetInput(Console.ReadLine(), 0);
    if (n > 0)
    {
        Console.WriteLine("Now please enter your exponent");
        int k = GetInput(Console.ReadLine(), 0);
```

```
            if (k > 0)
            {
                Console.WriteLine("Your base "+ n + " to the power of " + k + " is equal to: " +
intPow(n, k));
                Console.WriteLine("The value of your base added up with the exponent values
leading up to your exponent is " + intFacPow(n, k));
            }
        }


    }
  }
}
```

2.

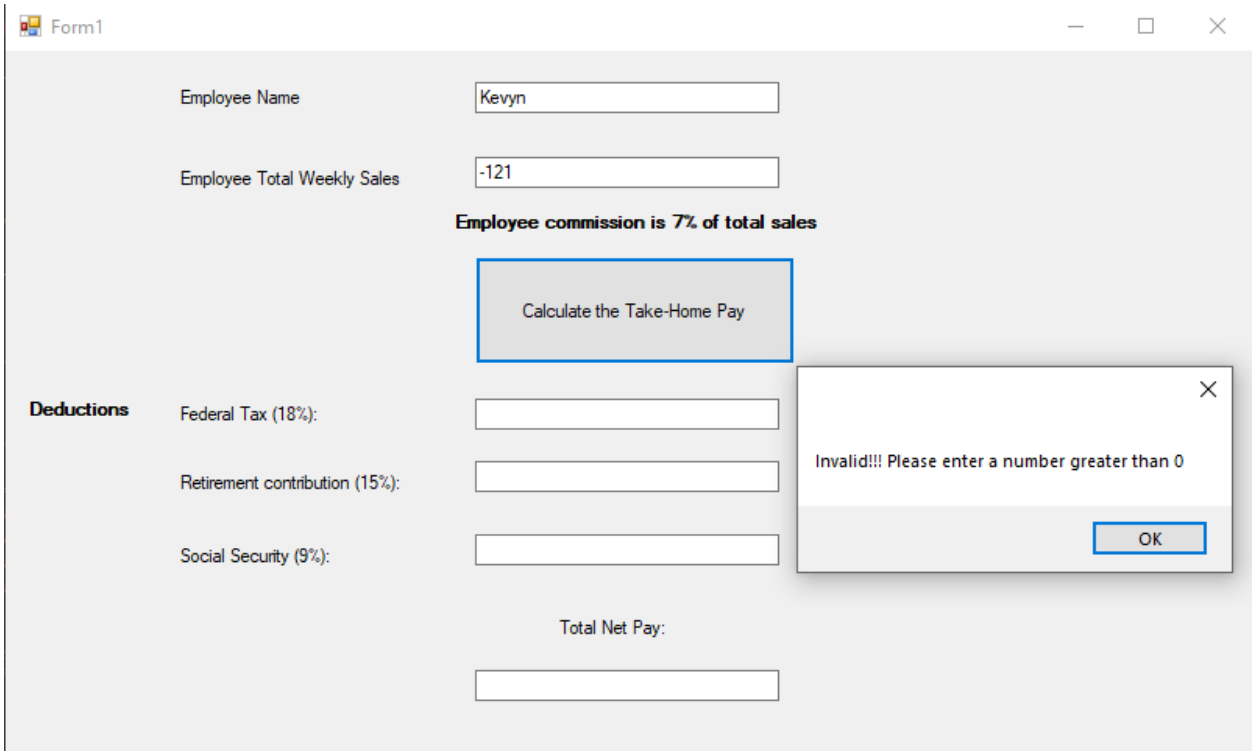


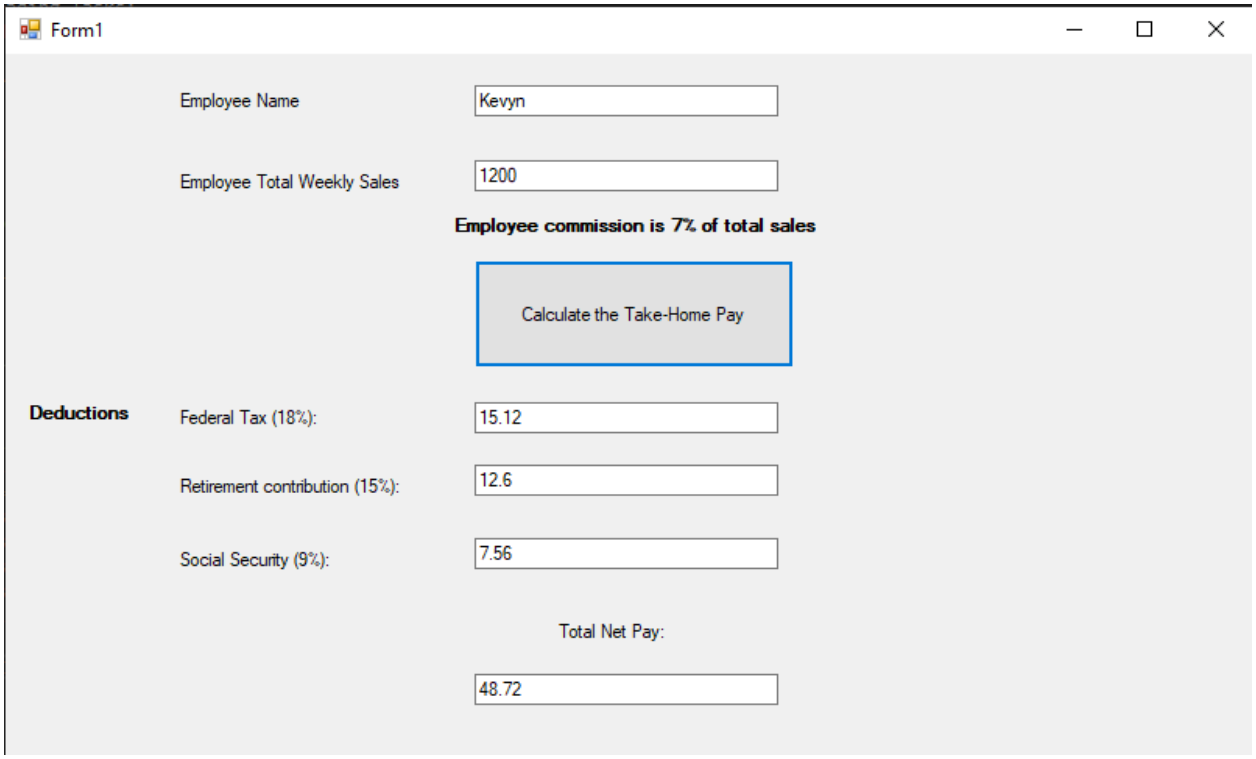

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Hw_7_problem_2
{
    class Program
    {

        static int Search(int[] A, int key)
        {
            int result=-1;
            for (int i=0; i < A.Length; i++)
            {

                if (A[i] == key)
                {
                    result = i;
                    break;
                }
```

```
        }
        return result;
    }

    static int GetInput(string x, int low, int high, int classize)
    {
        bool valid = true;

        valid = int.TryParse(x, out int y);

            if (!valid)
            {
                Console.WriteLine("Invalid!!!! Please enter an integer!!!!!!!");
            y = -9;

            }
            else if (y < low || y > high)
            {
                Console.WriteLine("Invalid!!! Please enter a number greater than " + low + " or less
than " + high);
            y = -9;

            }

        return y;

    }

    static void Main(string[] args)
    {
        int classize = 5;
        int[] scores = new int[classize];
        int key = 100;

        for (int i = 0; i<classize; i++)
        {
            Console.Write("Enter the score for Student #" + (i + 1) + ": ");
            int n = GetInput(Console.ReadLine(), 0, 100, classize);
            if (n >= 0 && n<=100)
            {
                scores[i] = n;
            }
            else
            {
```

```
                i--;
            }
        }

        for (int i = 0; i < classize; i++)
        {
            int StudentIndex = Search(scores, key);
            bool deez = true;

            if (StudentIndex != -1)
            {
                Console.WriteLine("Student #" + (StudentIndex + 1) + " earned {0}%", key);
                scores[StudentIndex] = 0;
                deez = false;
            }
            else if(!deez)
            {
                Console.WriteLine("No one got {0}%", key);
            }

        }


    }
  }
}
```

3.






```
using System;
using System.Collections.Generic;
```

```
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace HW_7_Problem_3
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {

            double sales, fedtax = 0, retirement = 0, social = 0, pay;
            sales = GetInput(textBox2.Text, 0);
            pay = CalculatePay(sales, ref fedtax, ref retirement, ref social);
            textBox3.Text = (Math.Round(fedtax, 2).ToString());
            textBox4.Text = (Math.Round(retirement, 2).ToString());
            textBox5.Text = (Math.Round(social, 2).ToString());
            textBox6.Text = (Math.Round(pay, 2).ToString());
            if (sales < 0)
            {
                Clear();
            }
        }

        static double CalculatePay(double sales, ref double fedtax, ref double retirement, ref
double social)
        {

            double pay = sales * 0.07;
            fedtax = pay * 0.18;
            retirement = pay * 0.15;
            social = pay * 0.09;
            pay = pay - fedtax - retirement - social;

            return pay;
```

```
        }

        static double GetInput(string input, int low)
        {
            bool valid = true;

            valid = double.TryParse(input, out double y);
            if (!valid)
            {
                MessageBox.Show("Invalid!!!! Please enter an integer!!!!!!!");
                y = -9;
            }
            else if (y <= low)
            {
                MessageBox.Show("Invalid!!! Please enter a number greater than " + low);
                y = -9;
            }
            return y;
        }

         void Clear()
        {
            textBox3.Clear();
            textBox4.Clear();
            textBox5.Clear();
            textBox6.Clear();
            textBox2.Clear();
        }
    }
}
```