```csharp
1 program bubble sort

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace consolelabprgs
{
    class prog1_bubblesort
    {
        static void Main(string[] args)
        { int[] arr = new int[10];
            int i,n,j,temp;
            Console.WriteLine("BUBBLE SORT");
            Console.WriteLine("---------------");
            Console.WriteLine("Enter how many elements to sort");
            n=int.Parse(Console.ReadLine());
            Console.WriteLine("Enter "+n+" elements");
            for(i=0;i<n;i++)
                arr[i]=int.Parse(Console.ReadLine());
            Console.WriteLine ("Elements before sorting");
            for(i=0;i<n;i++)
                Console.Write (arr[i]+"\t");
            Console.WriteLine();
            for(i=0;i<n;i++)
                {for(j=0;j<n-1-i;j++)
                    {if(arr[j]>arr[j+1])
                        {temp=arr[j];
```

```csharp
            arr[j]=arr[j+1];

            arr[j+1]=temp;

            }

          }

        }

      Console.WriteLine ("Elements after sorting");

      for(i=0;i<n;i++)

        Console.Write(arr[i]+"\t");

      Console.WriteLine();

      Console.ReadLine();

    }

    }

  }
```

2 programme bank css

```csharp
using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace consolelabprgs

{

  class bank

  {

    int accno;
```

```csharp
string accname;
float balance;
string acctype;
public void init()
{
    Console.Write("\nEnter the account no ");
    accno = int.Parse(Console.ReadLine());
    Console.Write("\nEnter account name ");
    accname = Console.ReadLine();
    Console.Write("\nEnter account type ");
    acctype = Console.ReadLine();
    Console.Write("\nEnter the amount ");
    balance = float.Parse(Console.ReadLine());
    Console.Write("\nAccount created. Press ENTER to continue");
}
public void deposit(float amt)
{
    balance = balance + amt;
    Console.Write("\n{0} deposited . Press ENTER to continue ", amt);
}
public void withdraw(float amt)
{
    if (balance < amt)
        Console.WriteLine("\n !!!!Not enough balance to withdraw ");
    else if (balance - amt < 500)
        Console.WriteLine("\n !!!!Minimum balance should be 500 ");
    else
```

```csharp
        {
            balance = balance - amt;

            Console.Write("\n{0} withdrawn . Press ENTER to continue", amt);

        }

    }


    public void display()

    {

        Console.Clear();

        Console.WriteLine("\n\t\tAccount Details");

        Console.WriteLine("\n\t\t==============");

        Console.WriteLine("The account no is " + accno);

        Console.WriteLine("The account name is " + accname);

        Console.WriteLine("The account type is " + acctype);

        Console.WriteLine("Total balance is " + balance);

        Console.WriteLine("Press ENTER to continue ");

    }
}


class prog2_bank

{

    static void Main()

    {

        bank b = new bank();

        float amt;
```

```csharp
int choice;
do
{

    Console.Clear();

    Console.WriteLine("\n Bank Transaction");

    Console.WriteLine("\n ----------------");

    Console.WriteLine("\n 1.Create an account");

    Console.WriteLine("\n 2.Deposit");

    Console.WriteLine("\n 3.Withdraw");

    Console.WriteLine("\n 4.Display");

    Console.WriteLine("\n 5.Exit");

    Console.Write("\n Enter an option");

    choice = int.Parse(Console.ReadLine());

    switch (choice)

    {

        case 1: b.init();

            Console.ReadLine();

            break;

        case 2: Console.Write("\n Enter amount to deposit ");

            amt = float.Parse(Console.ReadLine());

            b.deposit(amt);

            Console.ReadLine();

            break;

        case 3: Console.WriteLine("\n Enter amount to withdraw ");

            amt = float.Parse(Console.ReadLine());

            b.withdraw(amt);
```

```csharp
                    Console.ReadLine();

                    break;

                case 4: b.display();

                    Console.ReadLine();

                    break;

                case 5: Environment.Exit(0);

                    break;

                default: Console.WriteLine("\n Enter a valid option ");

                    Console.ReadLine();

                    break;
                }

            } while (choice != 5);


            Console.ReadLine();
        }


    }
}



3 program interface CSS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;


namespace consolelabprgs
```

```csharp
{
    interface acc
    {
        void accept();
    }
    interface dis
    {
        void display();
    }
    class student : acc, dis
    {
        int regno;
        string name;
        string course;
        int m1, m2, m3, total;
        float avg;
        public void accept()
        {
            Console.Write(" \nEnter Regno : ");
            regno = int.Parse(Console.ReadLine());
            Console.Write("\n Enter Name : ");
            name = Console.ReadLine();
            Console.Write("\n Enter course ");
            course = Console.ReadLine();
            Console.Write("\n Enter marks in DS : ");
            m1 = int.Parse(Console.ReadLine());
            Console.Write("\n Enter marks in OS : ");
```

```csharp
        m2 = int.Parse(Console.ReadLine());

        Console.Write("\n Enter marks in Maths : ");

        m3 = int.Parse(Console.ReadLine());

        Console.WriteLine("\n***********************************");

        total = m1 + m2 + m3;

        avg = Convert.ToSingle(total) / 3;

    }




    public void display()

    {

        Console.WriteLine("{0}\t{1}\t{2}\t{3}\t{4}\t{5}\t{6}\t{7} ", regno, name, course, m1, m2, m3, total,
avg);

    }

}

class prog3_interface

{

    public static void Main()

    {

        student[] s = new student[3];


        for (int i = 0; i < 3; i++)

            s[i] = new student();

        Console.WriteLine("\n Enter student details for 3 students");


        foreach (student st in s)
```

```csharp
        {
            acc i1 = (acc)st;

            i1.accept();
        }

        Console.Clear();

        Console.WriteLine("\n\t\t STUDENT REPORT");

        Console.WriteLine("\n\t\t --------------");

        Console.WriteLine("\nRegno\tName\tCourse\tDS\tOS\tMaths\tTotal\taverage");

        foreach (student st in s)
        {
            dis i2 = (dis)st;

            i2.display();
        }

        Console.ReadLine();


    }

  }

}


4   add complex


using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;


namespace consolelabprgs
```

```csharp
{
    class complex
    {
        double real;
        double imag;
        public complex()
        {
            real = 0.0; imag = 0.0;
        }

        public void accept()
        {
            Console.WriteLine("\n Enter the real part");
            real = double.Parse(Console.ReadLine());
            Console.WriteLine("\n Enter the imaginary part");
            imag = double.Parse(Console.ReadLine());
        }

        public static complex operator +(complex c1, complex c2)
        {
            complex c3 = new complex();
            c3.real = c1.real + c2.real;
            c3.imag = c1.imag + c2.imag;
            return c3;
        }
        public void display()
        {
            Console.Write("\t" + real);
```

```csharp
            Console.Write(" + i " + imag);

            Console.WriteLine();

        }

}


class prog4_add_complexno

{

    public static void Main()

    {

        complex a, b, c;

        a = new complex();

        b = new complex();

        c = new complex();

        Console.WriteLine("Addition of two complex nos using operator overloading");

        Console.WriteLine("----------------------------------------------------");

        Console.WriteLine("First complex no");

        Console.WriteLine("-----------------");

        a.accept();

        Console.WriteLine("Second complex no");

        Console.WriteLine("----------------");

        b.accept();

        c = a + b;

        Console.Clear();

        Console.WriteLine("\t\t\tSum of two complex numbers ");

        Console.WriteLine("\t\t\t-------------------------");

        Console.Write("\tFirst complex no  = ");
```

```csharp
            a.display();

            Console.Write("\tSecond complex no  = ");

            b.display();

            Console.Write("\tSum  = ");

            c.display();

            Console.ReadLine();

        }

    }

}


5 delagate

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;


namespace consolelabprgs

{

    delegate void printdelegate();

    class prog5_delegate

    {

        double length, breadth, height, base1, side, area;


        public void rect()

        {

            Console.WriteLine("\n Enter the length of the rectangle");

            length = double.Parse(Console.ReadLine());
```

```csharp
        Console.WriteLine("\n Enter the breadth of the rectangle");

        breadth = double.Parse(Console.ReadLine());

        area = length * breadth;

        Console.WriteLine(" Area of rectangle: " + area);

    }

    public void triangle()

    {

        Console.WriteLine("\n Enter the base of the triangle");

        base1 = double.Parse(Console.ReadLine());

        Console.WriteLine("\n Enter the height of the triangle");

        height = double.Parse(Console.ReadLine());

        area = 0.5 * base1 * height;

        Console.WriteLine(" Area of triangle: " + area);

    }

    public void square()

    {

        Console.WriteLine("\n Enter the side of the square");

        side = double.Parse(Console.ReadLine());

        area = side * side;

        Console.WriteLine(" Area of square: " + area);

    }


    static void Main()

    {

        prog5_delegate d = new prog5_delegate();

        int opt;

        do
```

```csharp
{
    Console.Clear();

    Console.WriteLine(" Illustration of delegates");

    Console.WriteLine(" -------------------------");

    Console.WriteLine("1.Area of rectangle");

    Console.WriteLine("2.Area of triangle");

    Console.WriteLine("3.Area of square");

    Console.WriteLine("4.Exit");

    Console.WriteLine("Enter your option");

    opt = int.Parse(Console.ReadLine());

    printdelegate p; // delegate instance

    switch (opt)

    {
        case 1: p = new printdelegate(d.rect);//delegate instance is mapped to rectangle

            p();

            Console.ReadLine();

            break;
        case 2: p = new printdelegate(d.triangle);

            p();

            Console.ReadLine();

            break;
        case 3: p = d.square;

            p();

            Console.ReadLine();

            break;
        case 4: break;
```

```
            }

        } while (opt != 4);


    }

  }


}
```

6 exception handling

```csharp
using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;


namespace consolelabprgs

{

  class MyException : Exception

  {

    public MyException(string str)

      : base(str)

    {

      Console.WriteLine("User Defined Exception called");

    }



  }

  class prog6_exception

  {
```

```csharp
public static void Main()
{
    int opt, number1, number2;
    string name;
    int[] arr = new int[5] { 10, 20, 30, 40, 50 };
    do
    {
        Console.Clear();
        Console.WriteLine("Illustrating exceptions");
        Console.WriteLine("1. Divide by zero");
        Console.WriteLine("2. Array index out of range");
        Console.WriteLine("3. User defined exception");
        Console.WriteLine("4. Exit");
        Console.Write("Enter an option");
        opt = int.Parse(Console.ReadLine());
        try
        {
            switch (opt)
            {
                case 1: Console.WriteLine("Enter  number1");
                    number1 = int.Parse(Console.ReadLine());
                    Console.WriteLine("Enter  number2");
                    number2 = int.Parse(Console.ReadLine());
                    Console.WriteLine("The result of number1/number2 is :" + number1 / number2);
                    Console.ReadLine();
                    break;
                case 2: Console.WriteLine("Array values");
```

```csharp
        for (int i = 0; i < 7; i++)

        {

            Console.Write("arr [ " + i + " ] = ");

            Console.WriteLine(arr[i]);

        }

        Console.ReadLine();

        break;

    case 3: Console.WriteLine("Enter a string <6 characters");

        name = Console.ReadLine();

        if (name.Length > 5)

            throw new MyException("The string length should not be more than 5 characters");

        else

            Console.WriteLine("Entered string is right");

        Console.ReadLine();

        break;

    case 4: break;

    }


}


catch (DivideByZeroException e)

{

    Console.WriteLine("Do not Divide a number by zero");

    Console.ReadLine();

}

catch (IndexOutOfRangeException e)

{
```

```csharp
            Console.WriteLine("Array index is out of range");

            Console.ReadLine();

        }


        catch (MyException e)

        {

            Console.WriteLine(e.Message);

            Console.ReadLine();

        }


        catch (Exception e)

        {

            Console.WriteLine("Exception caught here" + e.Message);

            Console.ReadLine();

        }
    } while (opt != 4);


    Console.ReadLine();

    }


  }
}


7 thread program

using System;

using System.Collections.Generic;
```

```csharp
using System.Linq;

using System.Text;

using System.Threading;


namespace consolelabprgs

{

    public class MyThread

    {


        public void Thread1()

        {

            Console.WriteLine("Executing Thread1");

            for (int i = 1; i <= 20; i++)

            {


                Thread thr = Thread.CurrentThread;

                Console.WriteLine(thr.Name + "=" + i);

                if (i == 5)

                {

                    try

                    {

                        Thread.Sleep(7000);

                    }


                    catch (Exception ae)

                    {
```

```csharp
                Console.WriteLine(ae.ToString());

            }


        }

    }

}

public void Thread2()

{

    Console.WriteLine("Executing Thread2");

    for (int i = 21; i <= 40; i++)

    {


        Thread thr = Thread.CurrentThread;

        Console.WriteLine(thr.Name + "=" + i);

        if (i == 35)

        {

            try

            {

                thr.Abort();

            }

            catch (ThreadAbortException ae)

            {

                Console.WriteLine("Thread 2 is terminated !!!!");

            }

        }


    }
```

```csharp
        }
    }
}

public class prog7_thread
{

    public static void Main()
    {
        Console.WriteLine("Executing Main thread!!!!");


        MyThread thr1 = new MyThread();

        MyThread thr2 = new MyThread();


        Thread tid1 = new Thread(new ThreadStart(thr1.Thread1));

        Thread tid2 = new Thread(new ThreadStart(thr2.Thread2));


        tid1.Name = "Thread 1";

        tid2.Name = "Thread 2";


        tid1.Start();

        tid2.Start();


        Console.WriteLine("End of Main");

        Console.ReadLine();

    }
}
```

```
}
```

9th net seler

```csharp
using System;

using System.Windows.Forms;

namespace WindowsApplication1

{

  public partial class Form1 : Form

  {

    float basic, da, hra, cca, pf, tax, gross, net;


    public Form1()

    {


      InitializeComponent();

    }

    private void button1_Click(object sender, EventArgs e)

    {


      basic = float.Parse(txtbasic.Text);

      da = basic * 66 / 100;

      hra = basic * 20 / 100;

      pf = basic * 15 / 100;

      cca = basic*5/100;
```

```csharp
        if (basic >= 10000 && basic < 15000)

            tax = basic * 2 / 100;

        else if (basic >= 15000 && basic < 20000)

            tax = basic * 3 / 100;

        else

            tax = basic * 5 / 100;

        gross = basic + da + hra + cca;

        net = gross - (pf + tax);


        txtda.Text = Convert.ToString(da);

        txthra.Text = Convert.ToString(hra);

        txtcca.Text = Convert.ToString(cca);

        txtpf.Text = Convert.ToString(pf);

        txttax.Text = Convert.ToString(tax);

        txtgross.Text = Convert.ToString (gross);

        txtnet.Text=Convert .ToString (net );

        }

    private void button2_Click(object sender, EventArgs e)

    {

        this.Close();



    }



    }

}
```

10 th.

```csharp
using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;


namespace windowsprgs

{

    public partial class changefont_prog9 : Form

    {

        public changefont_prog9()

        {

            InitializeComponent();

        }


        private void button1_Click(object sender, EventArgs e)

        {

            FontDialog dlg = new FontDialog();

            if (dlg.ShowDialog() == DialogResult.OK)

            {

                textBox1.Font = new Font(dlg.Font.Name, dlg.Font.Size, dlg.Font.Style);


            }

            else
```

```csharp
            MessageBox.Show("you clicked cancel");

    }



    private void button2_Click(object sender, EventArgs e)

    {

        ColorDialog cdlg = new ColorDialog();

        if (cdlg.ShowDialog() == DialogResult.OK)

        { textBox1.ForeColor = cdlg.Color; }

        else

            MessageBox.Show("you clicked cancel");



    }



    private void button3_Click(object sender, EventArgs e)

    {

        ColorDialog cdlg = new ColorDialog();

        if (cdlg.ShowDialog() == DialogResult.OK)

        { textBox1.BackColor  = cdlg.Color; }

        else

            MessageBox.Show("you clicked cancel");



    }



    private void changefont_prog9_Load(object sender, EventArgs e)

    {
```

```
        }

    }

}
```