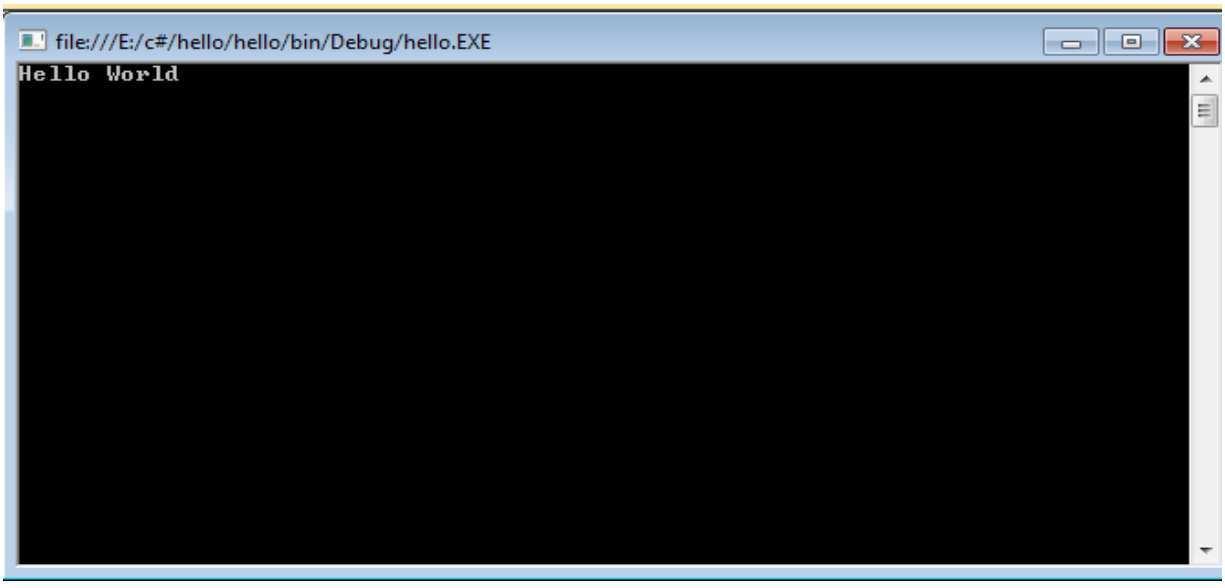
PRACTICAL FILE OF C# LANGUAGE

INDEX

- **WAP to print the words “Hello World”.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespace hello
{
class Program
   {
static void Main(string[] args)
      {
Console.WriteLine("Hello World");
Console.ReadKey();
      }
   }
}
```

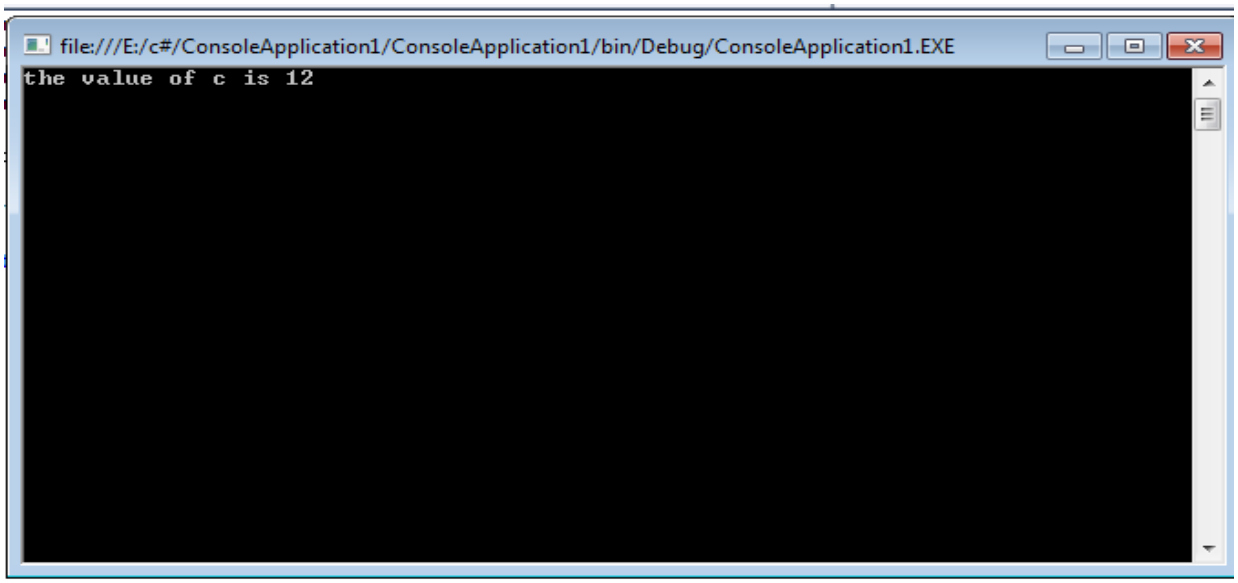
OUTPUT:



- **WAP to display sum of two numbers.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespace ConsoleApplication1
{
class Program
   {
static void Main(string[] args)
      {
int a, b, c;
         a = 10;
         b = 2;
         c = a + b;
Console.WriteLine("the value of c is {0}", c);
Console.ReadKey();
      }
   }
}
```

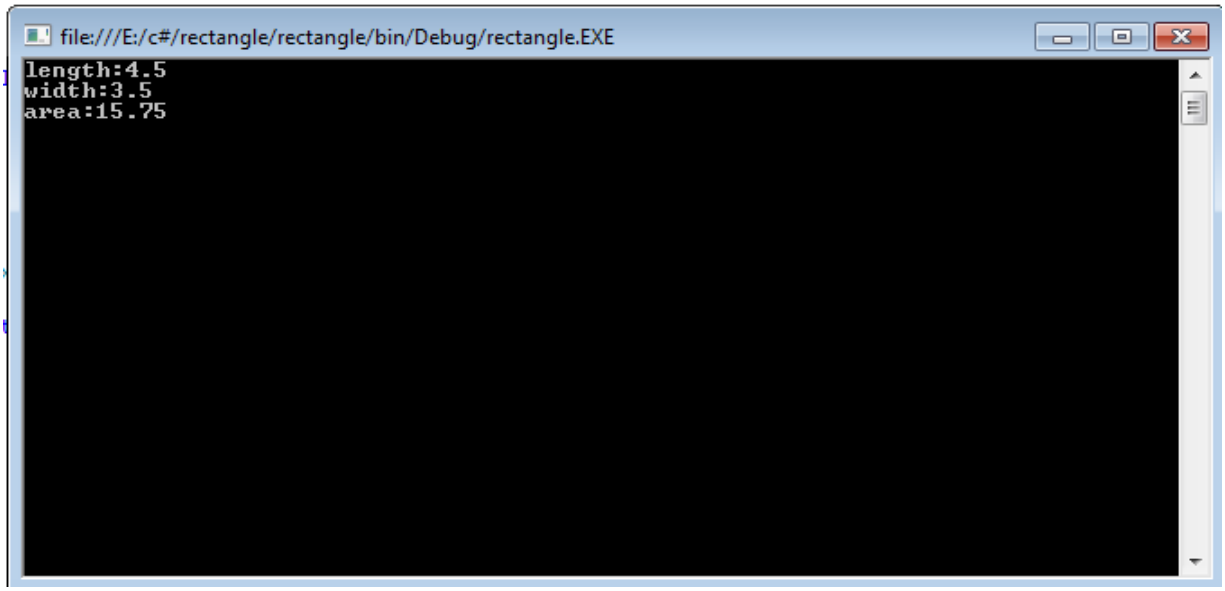
OUTPUT:



- **WAP to find area of rectangle.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespace rectangle
{
class rectangle
   {
double length;
double width;
public void acceptdetail()
       {
length = 4.5;
width = 3.5;
       }
public double getarea()
       {
return length * width;
       }
public void display()
       {
Console.WriteLine("length:{0}", length);
Console.WriteLine("width:{0}", width);
Console.WriteLine("area:{0}", getarea());
       }
   }
classExecuteRectangle
   {
static void Main(string[] args)
       {
rectangle r = new rectangle();
r.acceptdetail();
r.display();
Console.ReadLine();
       }
      }
   }
```

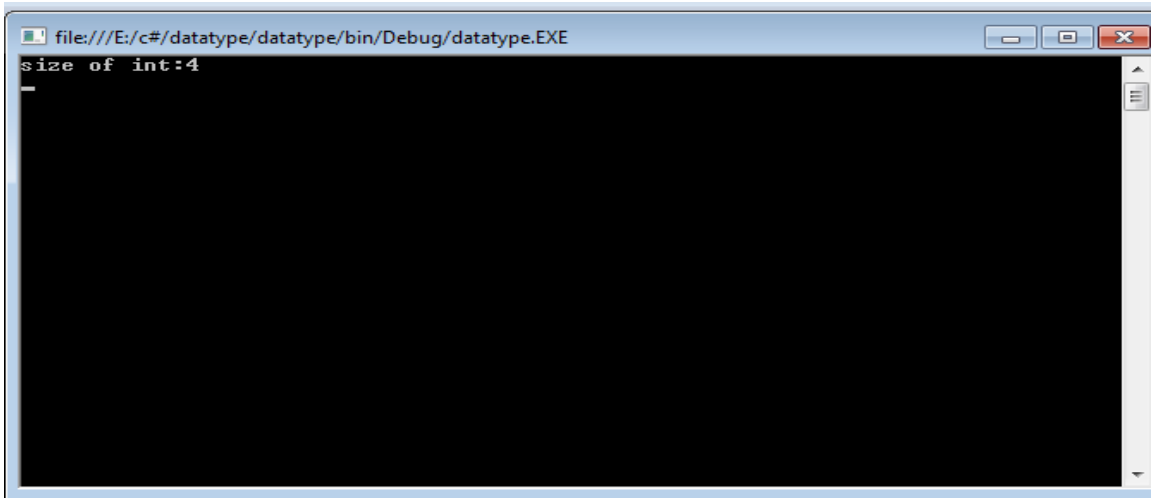
OUTPUT:


file:///E:/c#/rectangle/rectangle/bin/Debug/rectangle.EXE
length:4.5
width:3.5
area:15.75

- **WAP to get the size of data type.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacedatatype
{
class Program
   {
static void Main(string[] args)
      {
Console.WriteLine("size of int:{0}", sizeof(int));
Console.ReadLine();
      }
   }
}
```

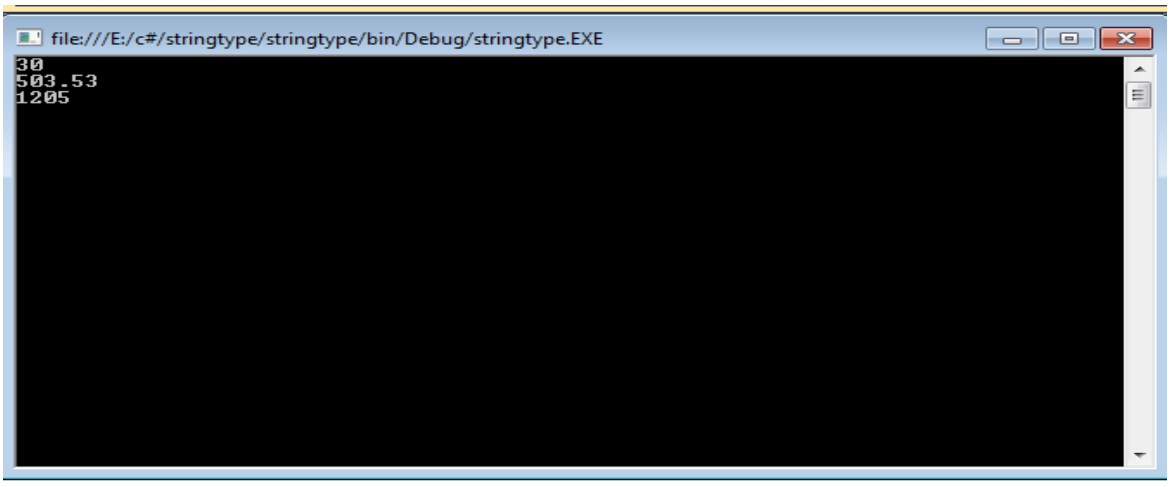
OUTPUT:



- **WAP to convert various value to string type.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacestringtype
{
class Program
   {
static void Main(string[] args)
      {
int a = 30;
double b=503.53;
         Int32 c=1205;
Console.WriteLine(a.ToString());
Console.WriteLine(b.ToString());
Console.WriteLine(c.ToString());
         String str=a.ToString();
Console.ReadLine();
      }
   }
}
```

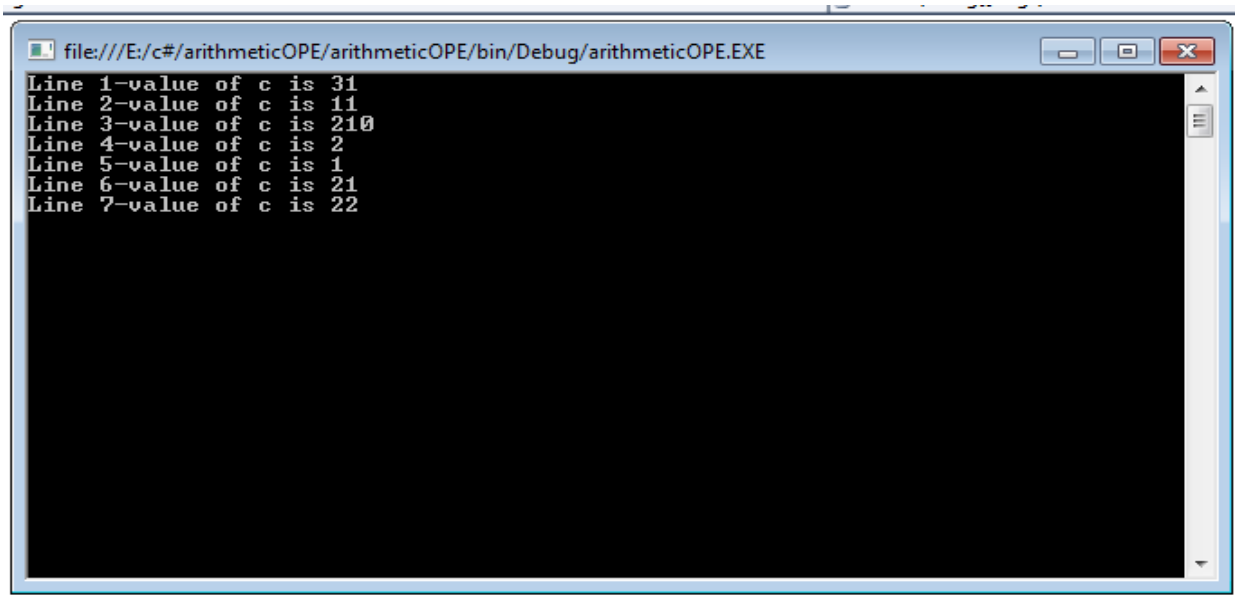
OUTPUT:



- **WAP using arithmetic operators.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacearithmeticOPE
{
class Program
   {
static void Main(string[] args)
      {
int a = 21;
int b = 10;
int c;
         c = a + b;
Console.WriteLine("Line 1-value of c is {0}", c);
         c = a - b;
Console.WriteLine("Line 2-value of c is {0}", c);
         c = a * b;
Console.WriteLine("Line 3-value of c is {0}", c);
         c = a / b;
Console.WriteLine("Line 4-value of c is {0}", c);
         c = a % b;
Console.WriteLine("Line 5-value of c is {0}", c);
         c = a++;
Console.WriteLine("Line 6-value of c is {0}", c);
         c = a--;
Console.WriteLine("Line 7-value of c is {0}", c);
Console.ReadLine();
      }
   }
}
```

OUTPUT:

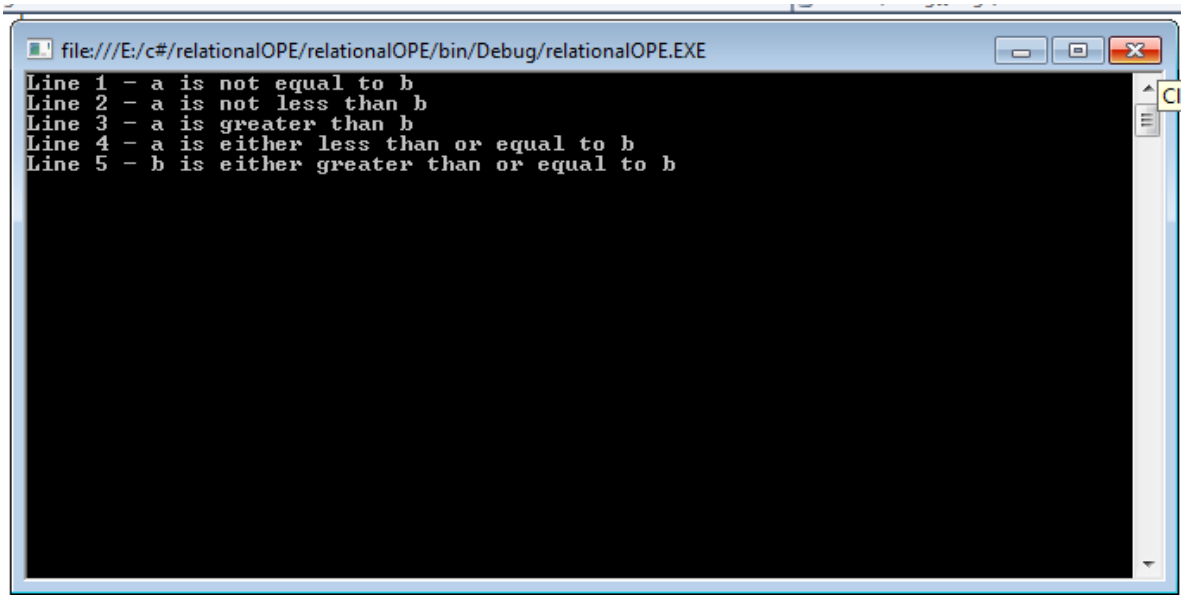

file:///E:/c#/arithmeticOPE/arithmeticOPE/bin/Debug/arithmeticOPE.EXE
Line 1-value of c is 31
Line 2-value of c is 11
Line 3-value of c is 210
Line 4-value of c is 2
Line 5-value of c is 1
Line 6-value of c is 21
Line 7-value of c is 22

- **WAP using relational operator.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacerelationalOPE
{
class Program
   {
static void Main(string[] args)
      {
int a = 21;
int b = 10;
if (a == b)
         {
Console.WriteLine("Line 1 - a is equal to b");
         }
else
         {
Console.WriteLine("Line 1 - a is not equal to b");
         }
if (a < b)
         {
Console.WriteLine("Line 2 - a is less than b");
         }
else
         {
Console.WriteLine("Line 2 - a is not less than b");
         }
if (a > b)
         {
Console.WriteLine("Line 3 - a is greater than b");
         }
else
         {
Console.WriteLine("Line 3 - a is not greater than b");
         }
         a = 5;
         b = 20;
if (a <= b)
         {
Console.WriteLine("Line 4 - a is either less than or equal to b");
         }
if (b >= a)
        {
```

```
Console.WriteLine("Line 5 - b is either greater than or equal to b");
            }
Console.ReadLine();

        }

    }

    }
```

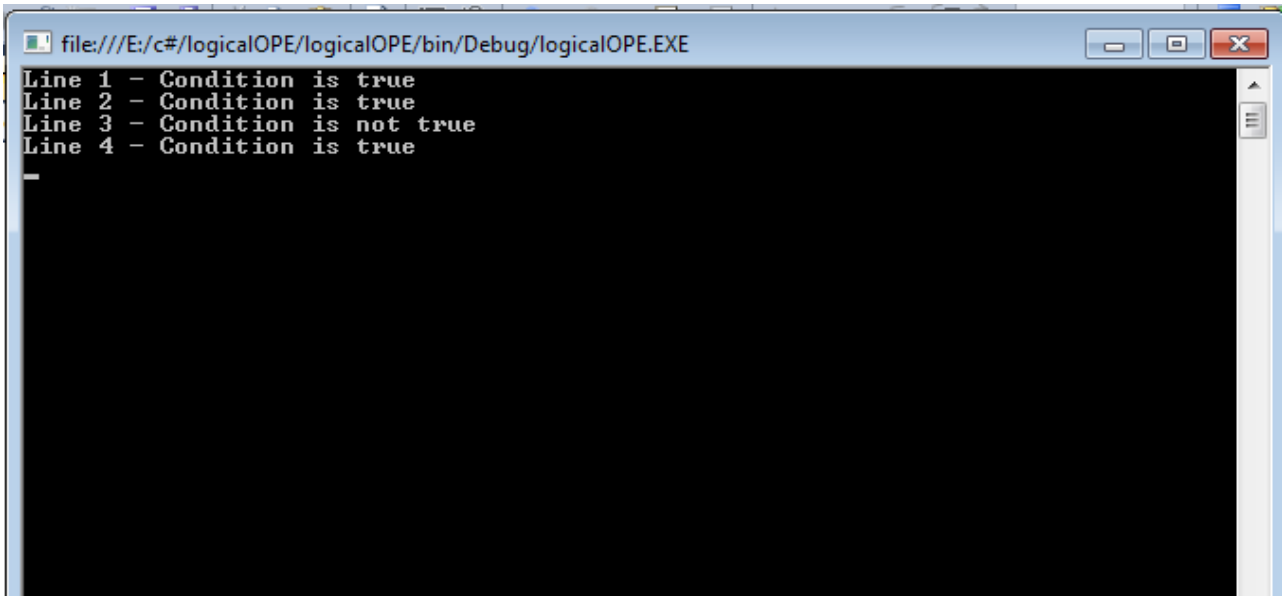
OUTPUT:



- **WAP using logical operators.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacelogicalOPE
{
class Program
   {
static void Main(string[] args)
      {

bool a = true;
bool b = true;
if (a && b)
         {
Console.WriteLine("Line 1 - Condition is true");
         }
if (a || b)
         {
Console.WriteLine("Line 2 - Condition is true");
         }
         a = false;
         b = true;
if (a && b)
         {
Console.WriteLine("Line 3 - Condition is true");
         }
else
         {
Console.WriteLine("Line 3 - Condition is not true");
         }
if (!a && b)
         {
Console.WriteLine("Line 4 - Condition is true");
         }
Console.ReadLine();

      }

   }

   }
```

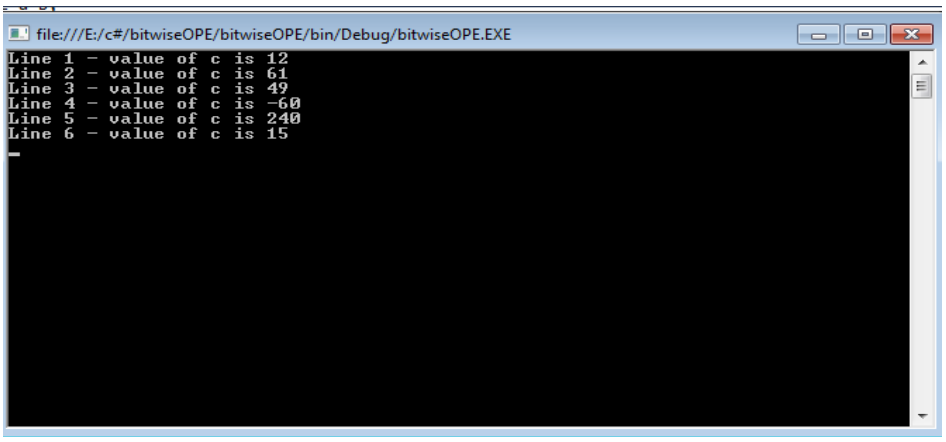
OUTPUT:


file:///E:/c#/logicalOPE/logicalOPE/bin/Debug/logicalOPE.EXE
Line 1 - Condition is true
Line 2 - Condition is true
Line 3 - Condition is not true
Line 4 - Condition is true

- **WAP using bitwise operator.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacebitwiseOPE
{
class Program
    {
static void Main(string[] args)
        {
int a = 60;
int b = 13;
int c = 0;
 c = a&b;
Console.WriteLine("Line 1 - value of c is {0}",c);
 c=a|b;
Console.WriteLine("Line 2 - value of c is {0}",c);
 c=a^b;
Console.WriteLine("Line 3 - value of c is {0}",c);
 c=-a;
Console.WriteLine("Line 4 - value of c is {0}",c);
 c=a<<2;
Console.WriteLine("Line 5 - value of c is {0}",c);
 c=a>>2;
Console.WriteLine("Line 6 - value of c is {0}",c);
Console.ReadLine();
        }
    }
    }
```

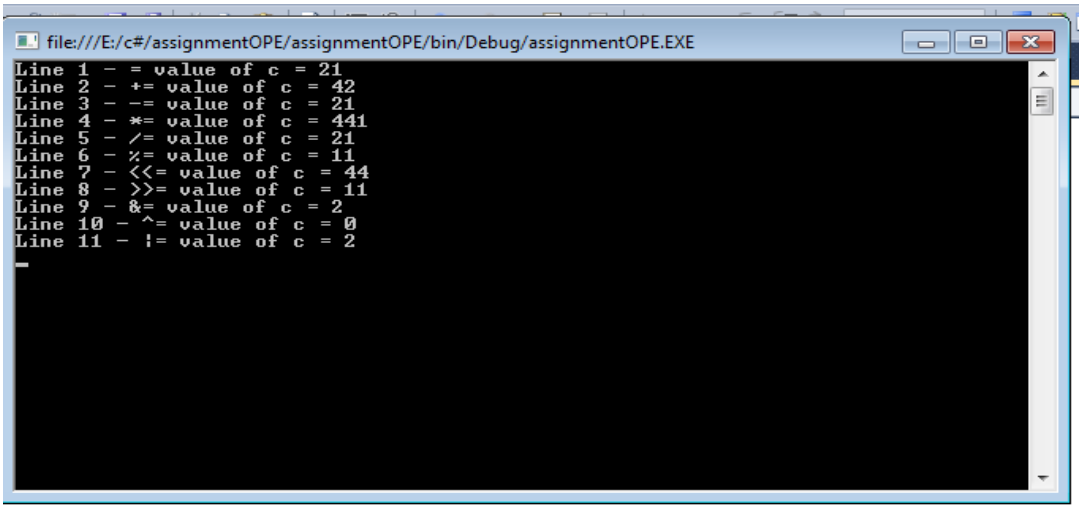
OUTPUT:



- **WAP using assignment operator.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespaceassignmentOPE
{
class Program
   {
static void Main(string[] args)
      {
int a = 21;
int c;
c=a;
Console.WriteLine("Line 1 - = value of c = {0}",c);
c += a;
Console.WriteLine("Line 2 - += value of c = {0}",c);
 c -= a;
Console.WriteLine("Line 3 - -= value of c = {0}",c);
c *= a;
Console.WriteLine("Line 4 - *= value of c = {0}",c);
c /= a;
Console.WriteLine("Line 5 - /= value of c = {0}",c);
 c=200;
c %= a;
Console.WriteLine("Line 6 - %= value of c = {0}",c);
 c <<= 2;
Console.WriteLine("Line 7 - <<= value of c = {0}",c);
 c >>= 2;;
Console.WriteLine("Line 8 - >>= value of c = {0}",c);
c&=2;
Console.WriteLine("Line 9 - &= value of c = {0}",c);
c ^=2;
Console.WriteLine("Line 10 - ^= value of c = {0}",c);
c |=2;
Console.WriteLine("Line 11 - |= value of c = {0}",c);
Console.ReadLine();
      }

   }

   }
```

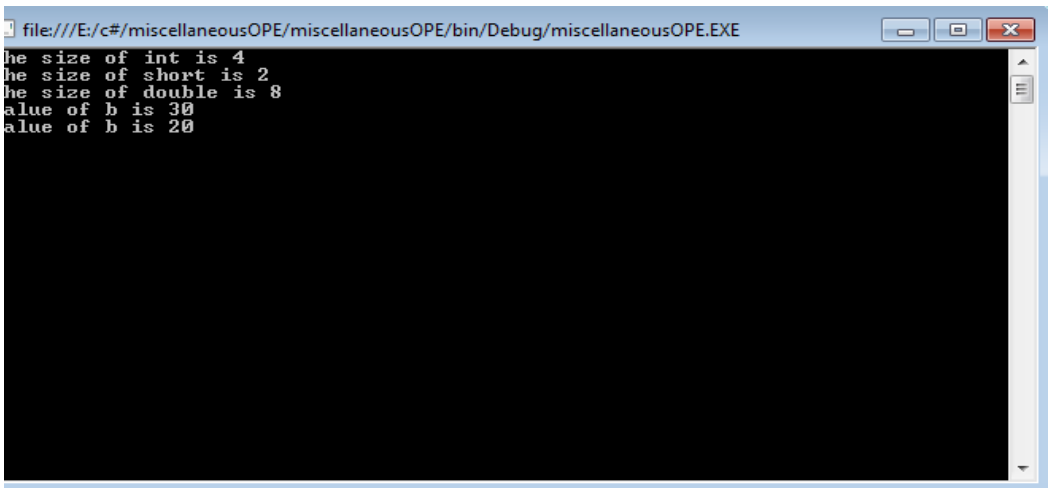
OUTPUT:


file:///E:/c#/assignmentOPE/assignmentOPE/bin/Debug/assignmentOPE.EXE
Line 1 - = value of c = 21
Line 2 - += value of c = 42
Line 3 - -= value of c = 21
Line 4 - *= value of c = 441
Line 5 - /= value of c = 21
Line 6 - %= value of c = 11
Line 7 - <<= value of c = 44
Line 8 - >>= value of c = 11
Line 9 - &= value of c = 2
Line 10 - ^= value of c = 0
Line 11 - |= value of c = 2

- **WAP using miscellaneous operator.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacemiscellaneousOPE
{
class Program
   {
static void Main(string[] args)
      {
Console.WriteLine("The size of int is {0}", sizeof(int));
Console.WriteLine("The size of short is {0}", sizeof(short));
Console.WriteLine("The size of double is {0}", sizeof(double));

int a, b;
         a = 10;
         b = (a == 1) ?20 : 30;
Console.WriteLine("value of b is {0}", b);

         b = (a == 10) ?20 : 30;
Console.WriteLine("value of b is {0}",b);
Console.ReadLine();

      }
   }
}
```

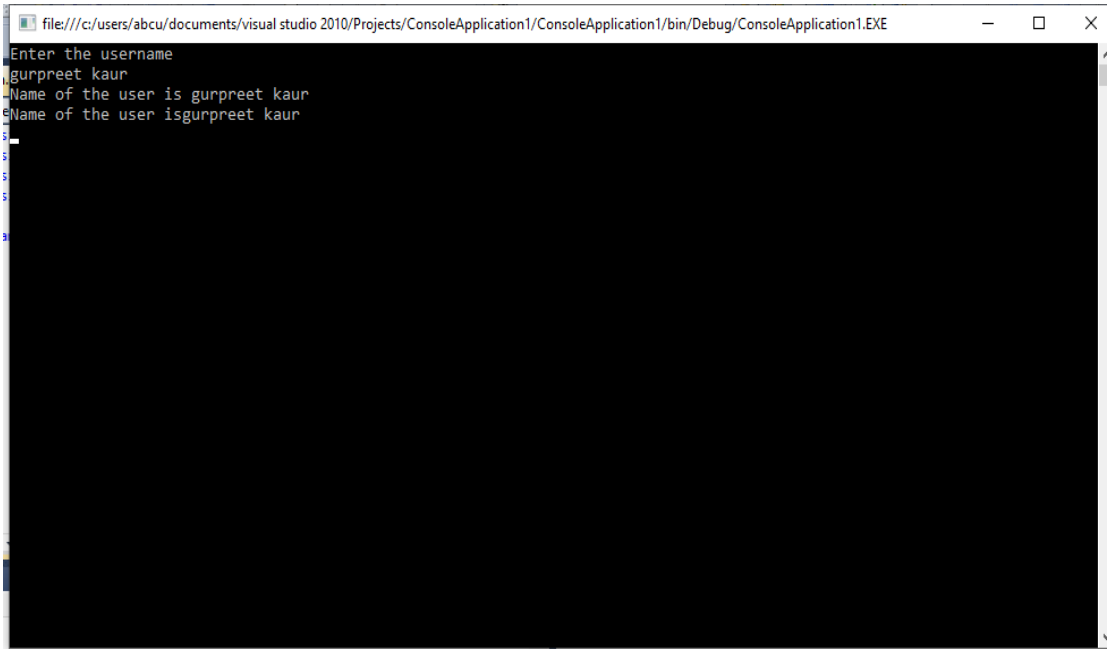
OUTPUT:



- **WAP to display name of user using input output method.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;

namespacenameofuser
{
class Program
   {
static void Main(string[] args)
      {
Console.WriteLine("Enter the username");
string username = Console.ReadLine();
Console.WriteLine("Name of user is {0}", username);
Console.WriteLine("Name of user is " +  username);
Console.ReadLine();
      }
   }
}
```

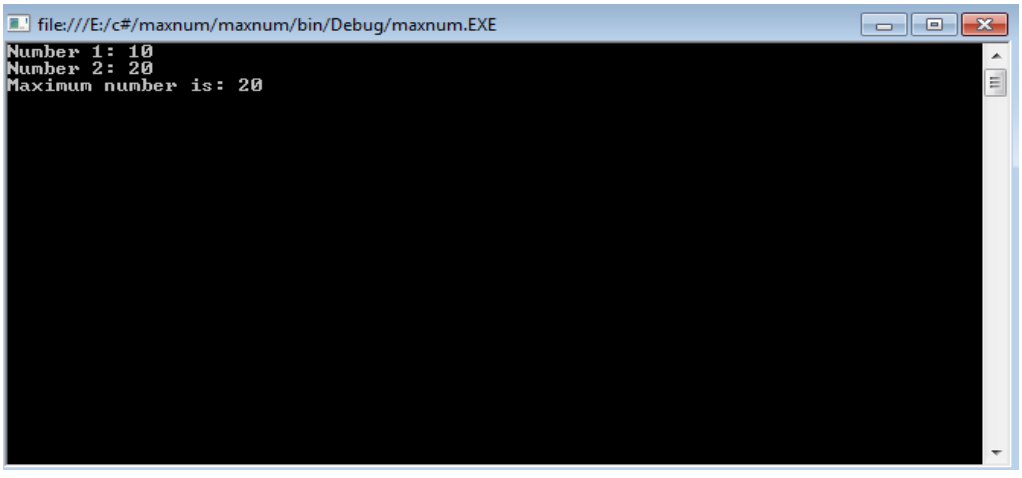
OUTPUT:



- **WAP to find maximum number between two numbers.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacemaxnum
{
classfindmax
   {
static void Main(string[] args)
         {
int num1 = 10, num2 = 20;
intmaxNum;
Console.WriteLine("Number 1: "+num1);
Console.WriteLine("Number 2: "+num2);
if(num1>num2)
            {
maxNum = num1;
            }
else
            {
maxNum = num2;
            }
Console.WriteLine("Maximum number is: "+maxNum);
Console.ReadKey();
Console.ReadLine();
         }
      }
   }
```

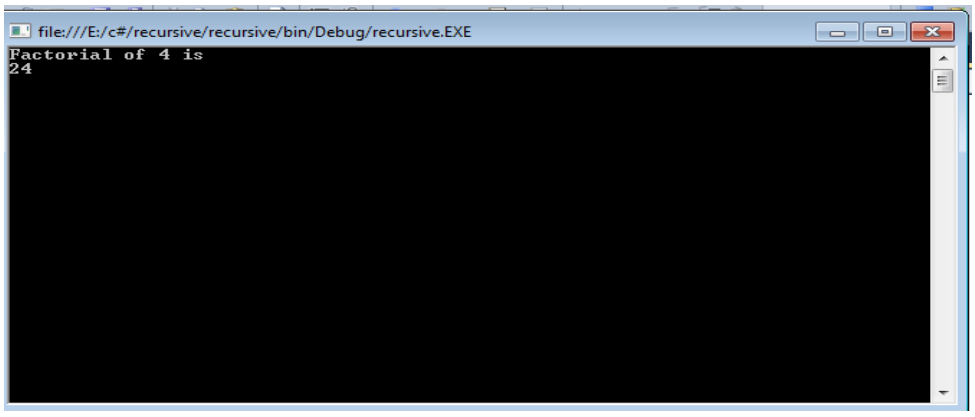
OUTPUT:



- **WAP of factorial using recursive method.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespace recursive
{
classfactorial
   {
publicint fact(int n)
      {
int r;
if (n == 1)
         {
return 1;
         }
else
         {
            r = fact(n - 1) * n;
return r;
         }
      }
staticvoid Main(string[] args)
      {
factorial f = newfactorial();
int result = f.fact(4);
Console.WriteLine("Factorial of 4 is");
Console.WriteLine(result);
Console.ReadLine();
}
   }
}
```

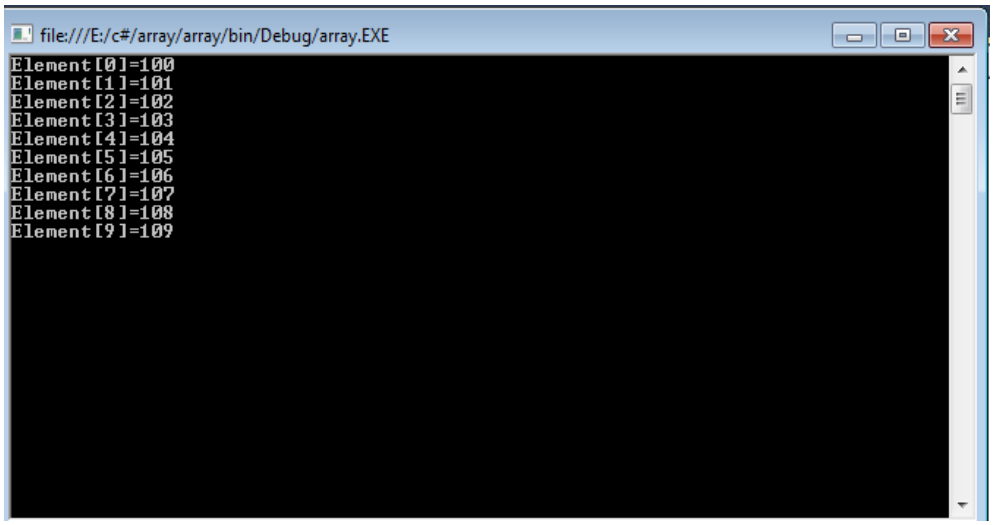
OUTPUT:



- **WAP to initialize and Display the value of array.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespace array
{
classarray
   {
staticvoid Main(string[] args)
      {
int[] n = newint[10];
for (inti = 0; i< 10; i++)
         {
n[i] = i + 100;
         }
for (int j=0;j<10;j++)
         {
Console.WriteLine("Element[{0}]={1}",j,n[j]);
         }
Console.ReadKey();


      }
   }
}
```

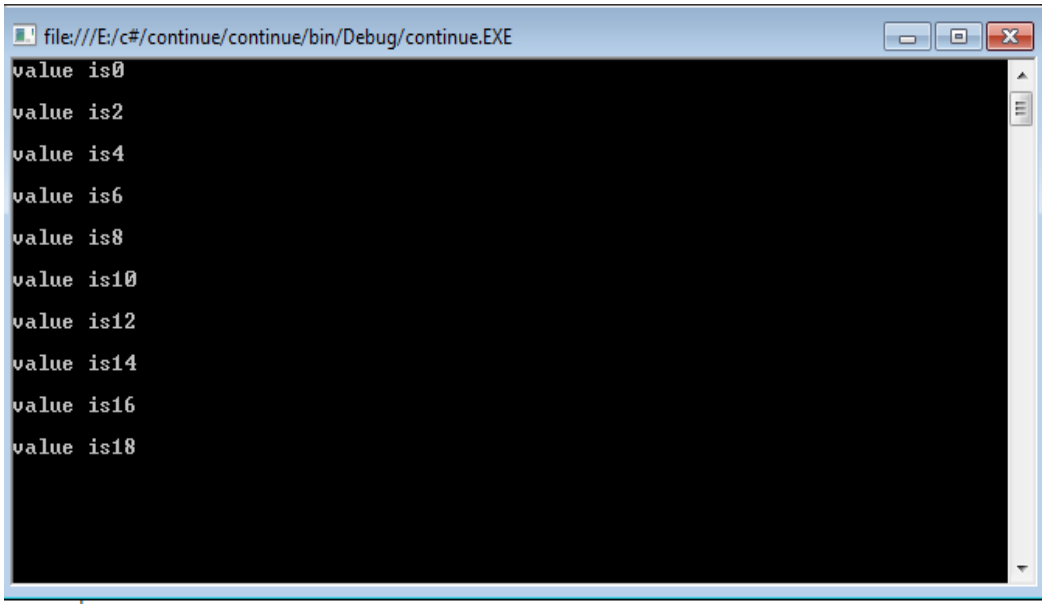
OUTPUT:



- **WAP to print even numbers from 0 to 20 using continue statement.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacecontinuest
{
classProgram
   {
staticvoid Main(string[] args)
      {
for (inti = 0; i< 20; i++)
         {
if (i % 2 == 1)
continue;
Console.WriteLine("value is{0}", i);
Console.ReadLine();
         }
      }
   }
}
```

OUTPUT:

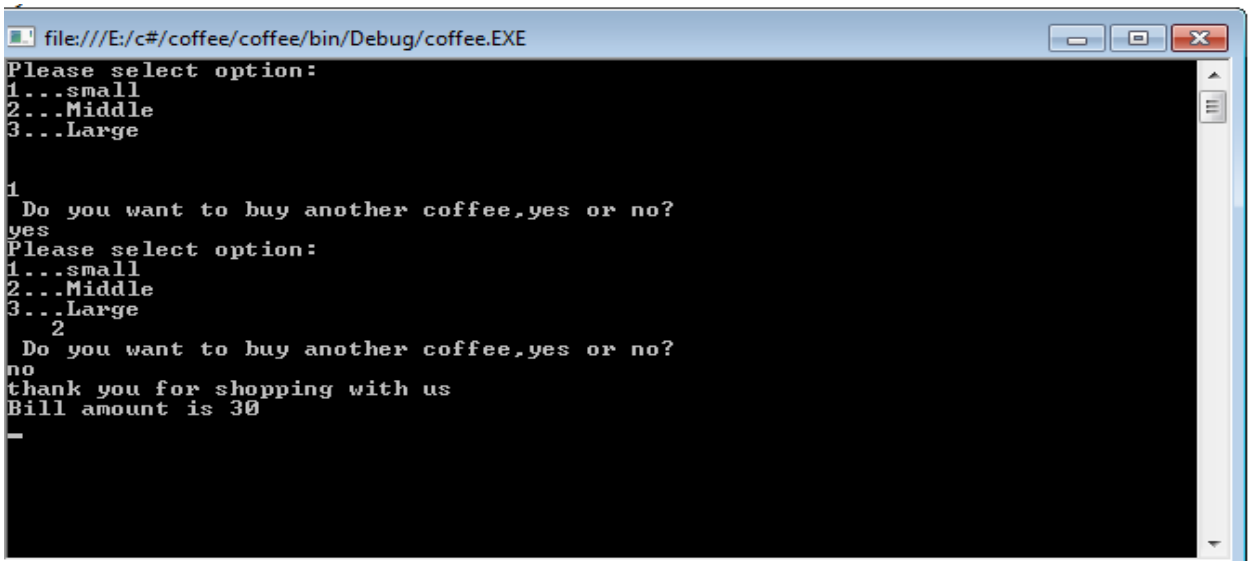


- **WAP of Coffee Shopping using switch.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespace coffee
{
class shopping
   {
static void Main(string[] args)
      {
intTotalCoffeeCost = 0;
         Start:
Console.WriteLine("Please select option: \n1...small \n2...Middle \n3...Large");
intUserChoice = int.Parse(Console.ReadLine());
switch (UserChoice)
         {
case 1:
TotalCoffeeCost += 10;
break;
case 2:
TotalCoffeeCost += 20;
break;
case 3:
TotalCoffeeCost += 30;
break;
default:
Console.WriteLine("your choice {0} is invalid", UserChoice);
goto Start;
         }
      Decide:
Console.WriteLine(" Do you want to buy another coffee,yes or no?");
stringUserDecision = Console.ReadLine();
switch (UserDecision.ToUpper())
      {
case "YES":
goto Start;
case "NO":
break;
default:
Console.WriteLine("your choice {0} is invalid:please try again.");
goto Decide;
      }
Console.WriteLine("thank you for shopping with us");
Console.WriteLine("Bill amount is {0}", TotalCoffeeCost);
```

```
Console.ReadLine();
      }
   }
}
```

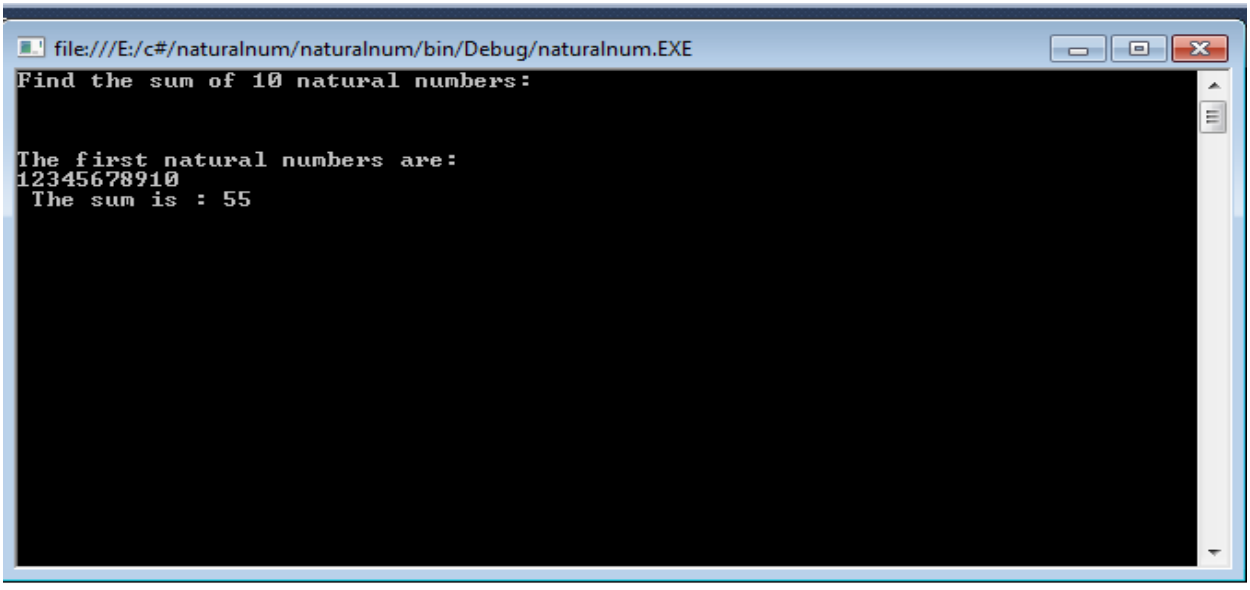
OUTPUT:



- **WAP to find the sum of first 10 natural numbers.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacenaturalnum
{
class Program
   {
static void Main(string[] args)
      {
int j, sum = 0;
Console.WriteLine("Find the sum of 10 natural numbers:\n");
Console.Write("\n\n");
Console.Write("The first natural numbers are:\n");
for (j = 1; j <= 10; j++)
         {
sum = sum + j;
Console.Write("{0}", j);
         }
Console.Write("\n The sum is : {0}\n", sum);
Console.ReadLine();
      }
   }
}
```

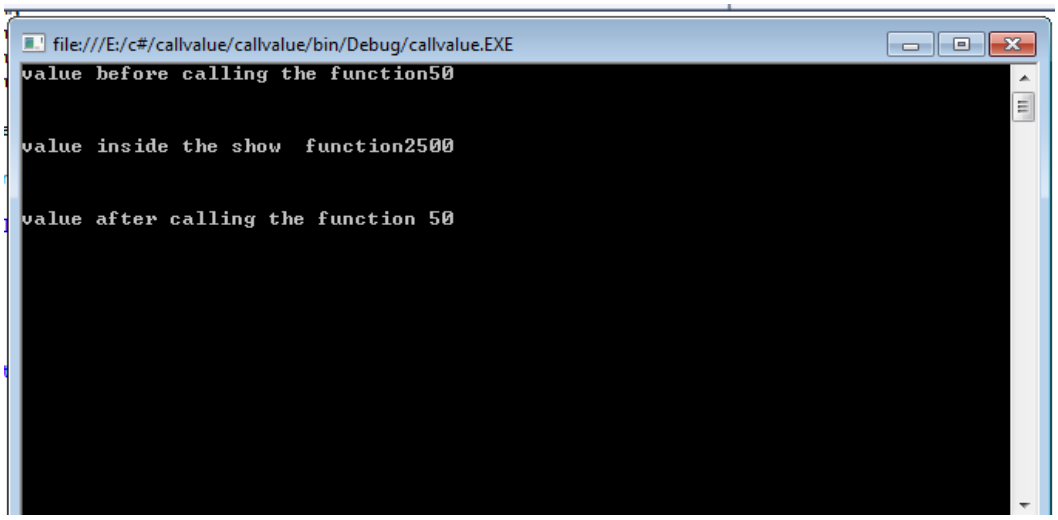
OUTPUT:



- **WAP using call by value.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacecallvalref
{
classProgram
   {
publicvoid show(intval)
      {
val *= val;
Console.WriteLine("value inside the show  function" + val);
Console.WriteLine("\n\n");
      }
staticvoid Main(string[] args)
      {
intval = 50;
Program p = newProgram();
Console.WriteLine("value before calling the function" + val);
Console.WriteLine("\n\n");
p.show(val);
Console.WriteLine("value after calling the function " + val);
Console.ReadLine();
      }
   }
}
```

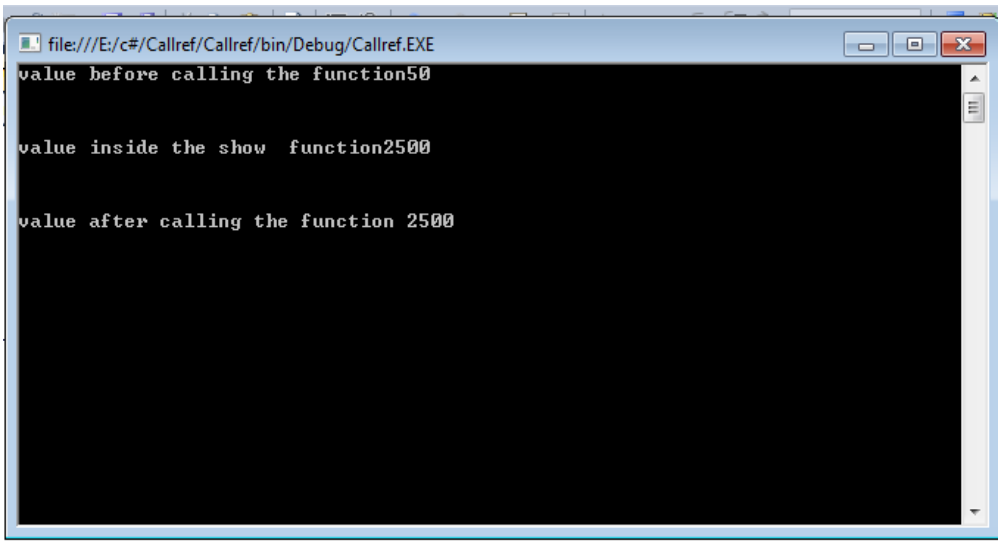
OUTPUT:



- **WAP using call by reference.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespaceCallref
{
classProgram
   {
publicvoid show(refintval)
      {
val *= val;
Console.WriteLine("value inside the show  function" + val);
Console.WriteLine("\n\n");
      }
staticvoid Main(string[] args)
      {
intval = 50;
Program p = newProgram();
Console.WriteLine("value before calling the function" + val);
Console.WriteLine("\n\n");
p.show(refval);
Console.WriteLine("value after calling the function " + val);
Console.ReadLine();
      }
   }
}
```

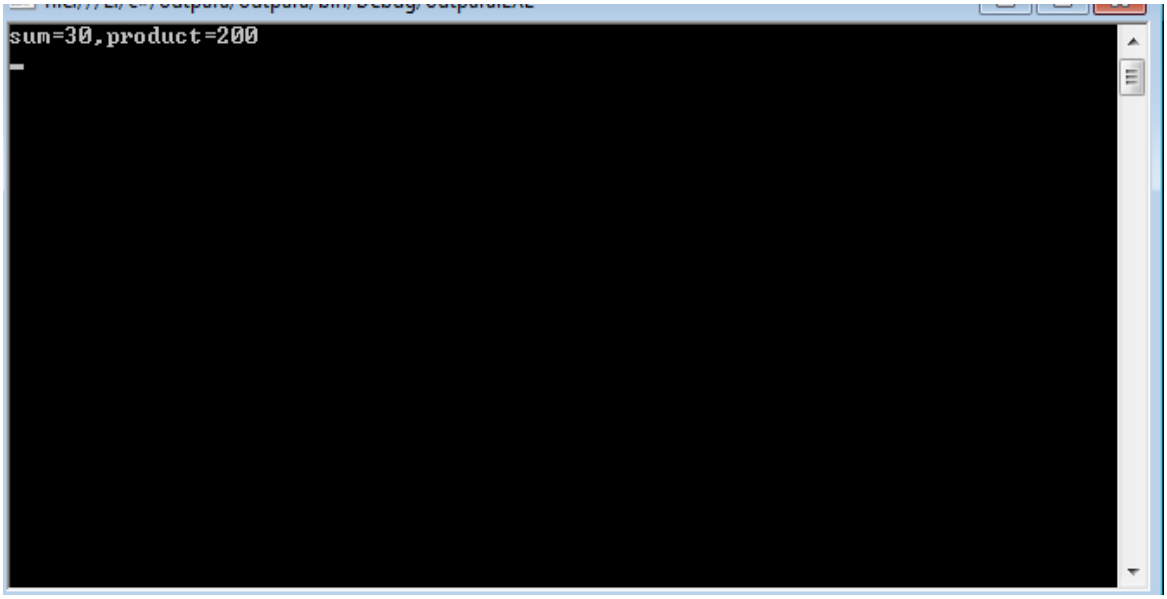
OUTPUT:



- **WAP using out parameters.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;

namespaceoutpara
{
classProgram
   {
staticvoid Main(string[] args)
      {
int total = 0;
int product = 0;
calculator(10, 20, out total, out product);
Console.WriteLine("sum={0},product={1}", total, product);
Console.ReadLine();
      }
publicstaticvoid calculator(int n1,int n2,outints,outint p)
      {
         s = n1 + n2;
         p = n1 * n2;
      }
      }
   }
```

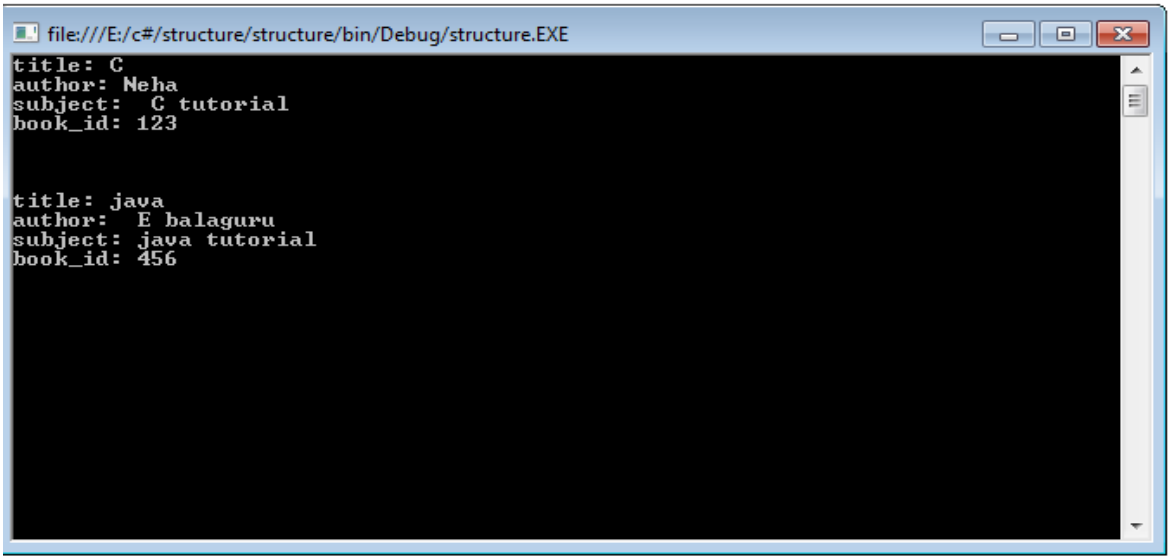
OUTPUT:



- **WAP using Structure.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespace structure
{
struct Books
   {
private string title;
private string author;
private string subject;
privateintbook_id;
public void getvalues(string t, string a, string s, int id)
      {
title = t;
author = a;
subject = s;
book_id = id;
      }
public void display()
      {
Console.WriteLine("title: {0}", title);
Console.WriteLine("author: {0}", author);
Console.WriteLine("subject: {0}", subject);
Console.WriteLine("book_id: {0}", book_id);
      }
   };
public class testStructure
   {
public static void Main(string[] args)
      {
         Books Book1 = new Books();
         Books Book2 = new Books();
Book1.getvalues("C", "Neha", " C tutorial", 123);
Book2.getvalues("java", " E balaguru", "java tutorial", 456);
Book1.display();
Console.Write("\n\n\n");
Book2.display();
Console.ReadKey();
      }
   }
}
```

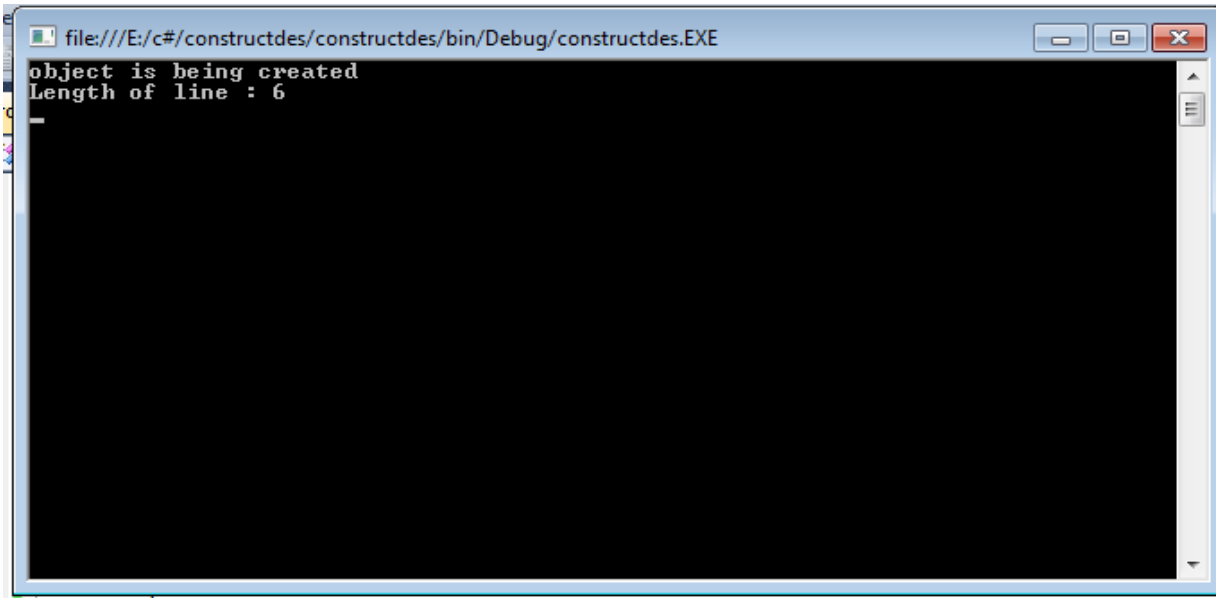
OUTPUT:


file:///E:/c#/structure/structure/bin/Debug/structure.EXE
title: C
author: Neha
subject: C tutorial
book_id: 123
title: java
author: E balaguru
subject: java tutorial
book_id: 456

- **WAP using Constructor and Destructor.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespaceconstructdes
{
class Line
   {
private double length;
public Line() // constructor
      {
Console.WriteLine("object is being created");
      }
      ~Line() // destructor
      {
Console.WriteLine("object is being deleted");
      }
public void setlength(double len)
      {
length=len;
      }
public double getlength()
      {
return length;
      }
static void Main(string[] args)
      {
         Line line = new Line();
line.setlength(6.0);
Console.WriteLine("Length of line : {0}",line.getlength());
Console.ReadLine();
      }
   }
}
```

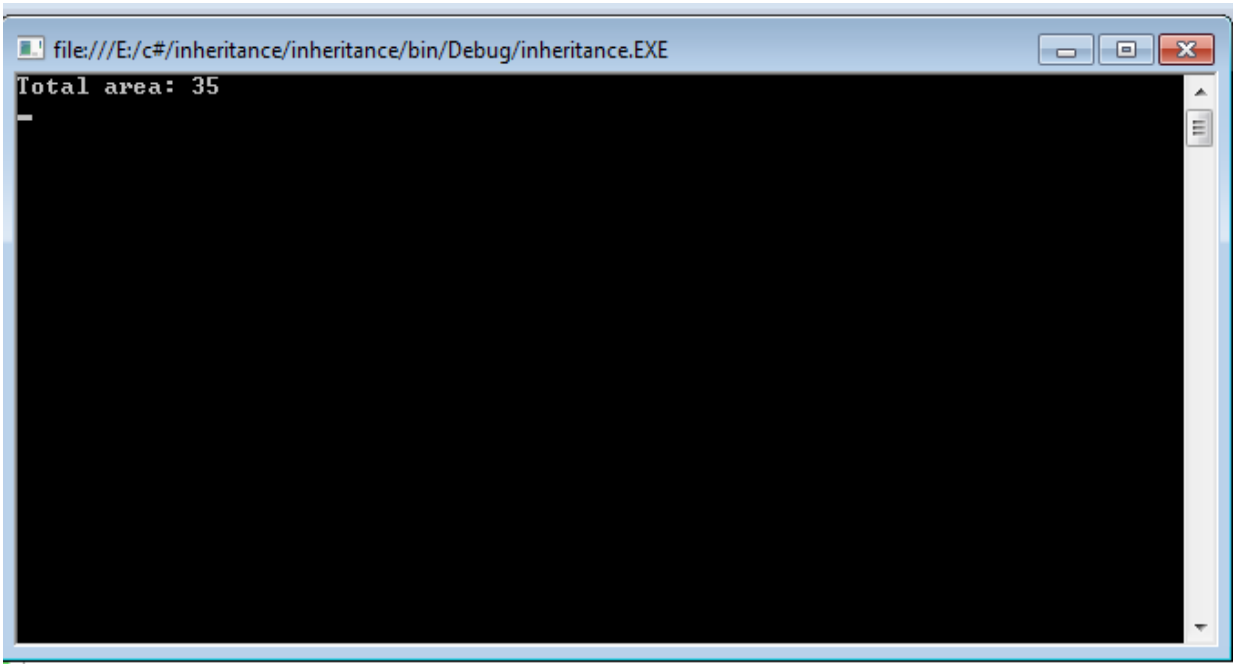
OUTPUT:


file:///E:/c#/constructdes/constructdes/bin/Debug/constructdes.EXE
object is being created
Length of line : 6

- **WAP using Inheritance.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespace inheritance
{
class shape
   {
public void setwidth(int w)
       {
width=w;
       }
public void setheight(int h)
       {
height = h;
       }
protectedint width;
protectedint height;
   }
class rectangle : shape
   {
publicintgetarea()
       {
return (width * height);
       }
   }
classrectangletest
   {
static void Main(string[] args)
       {
rectanglerect = new rectangle();
rect.setwidth(5);
rect.setheight(7);
Console.WriteLine("Total area: {0}", rect.getarea());
Console.ReadKey();
       }
   }
}
```

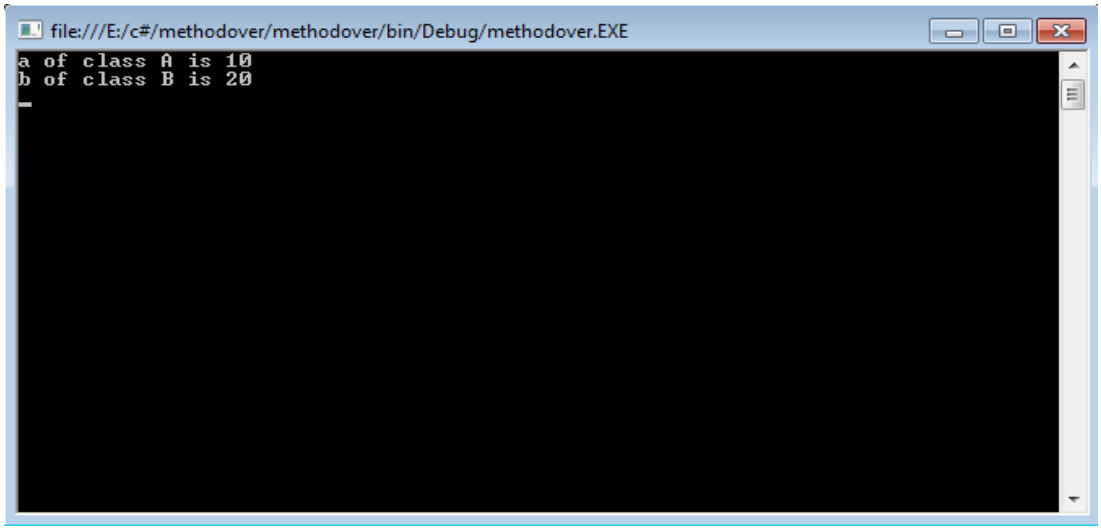
OUTPUT:


file:///E:/c#/inheritance/inheritance/bin/Debug/inheritance.EXE
Total area: 35

- **WAP using Method Overriding.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespacemethodover
{
classA
   {
protectedint a;
public A(int x)
      {
         a = x;
      }
publicvirtualvoid display()
      {
Console.WriteLine(" a of class A is {0}", a);
      }
classB : A
      {
protectedint b;
public B(int x, int y)
            : base(x)
         {
            b = y;
         }
publicoverridevoid display()
         {
Console.WriteLine("a of class A is {0}", a);
Console.WriteLine("b of class B is {0}", b);
Console.ReadLine();
         }
      }
classmethodoverride
      {
staticvoid Main(string[] args)
      {
Bobj = newB(10, 20);
obj.display();
      }
      }
   }
}
```

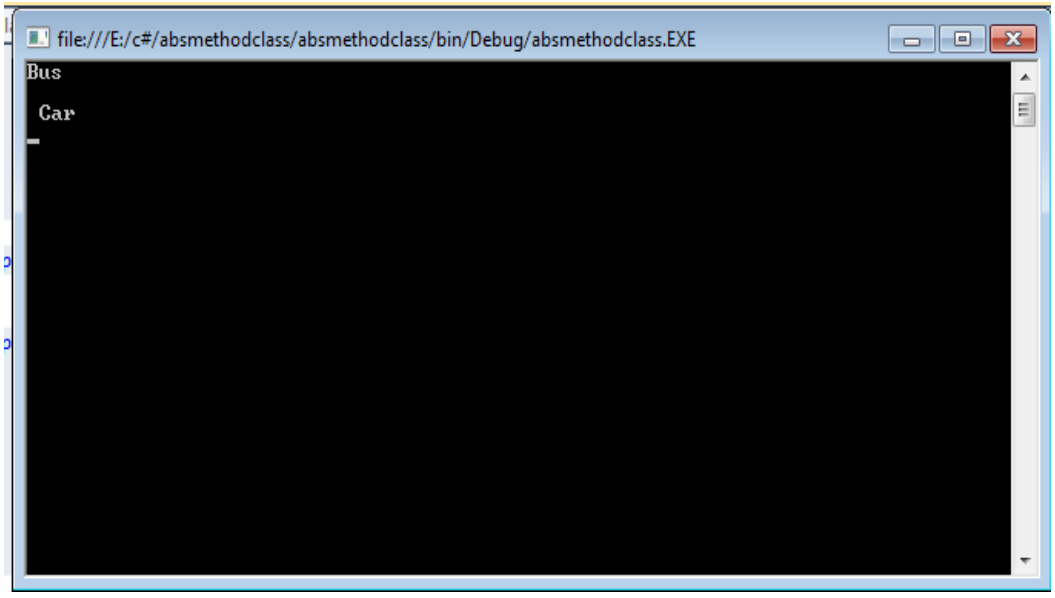
OUTPUT:


file:///E:/c#/methodover/methodover/bin/Debug/methodover.EXE
a of class A is 10
b of class B is 20

- **WAP using abstract method and class.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
namespaceabsmethodclass
{
publicabstractclassVehicle
   {
publicabstractvoid display();
   }
publicclassBus : Vehicle
   {
publicoverridevoid display()
      {
Console.WriteLine("Bus");
Console.ReadLine();
      }
   }
publicclassCar : Vehicle
         {
publicoverridevoid display()
            {
Console.WriteLine(" Car");
Console.ReadLine();
            }
         }
publicclassProgram
   {

publicstaticvoid Main()
      {
Vehicle v;
         v = newBus();
v.display();
         v = newCar();
v.display();

      }
   }
}
```

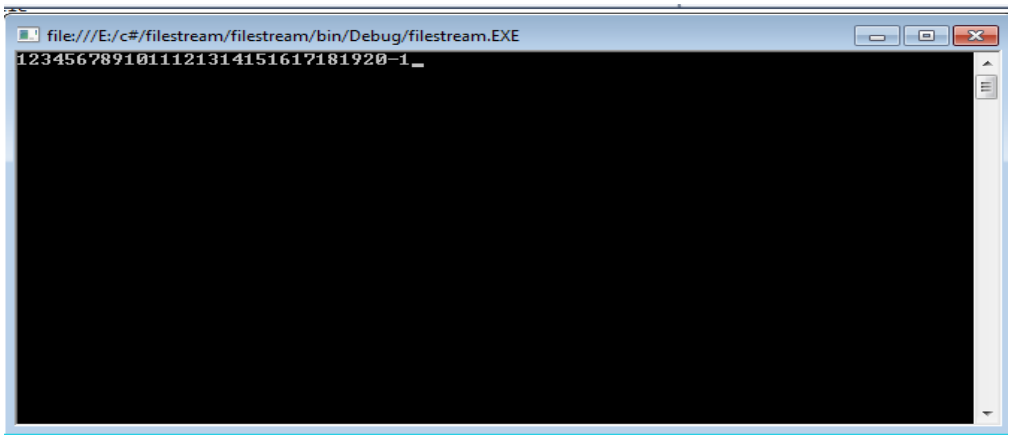
OUTPUT:


file:///E:/c#/absmethodclass/absmethodclass/bin/Debug/absmethodclass.EXE
Bus
Car

- **WAP using file stream.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
using System.IO;
namespace file
{
class Program
   {
static void Main(string[] args)
      {
FileStreamF = new FileStream("file.docx", FileMode.OpenOrCreate, FileAccess.ReadWrite);
for (inti = 1; i<= 20; i++)
         {
F.WriteByte((byte)i);
         }
F.Position = 0;
for (inti = 0; i<= 20; i++)
         {
Console.Write(F.ReadByte() + "");
         }
F.Close();
Console.ReadKey();

      }
   }
}
```

OUTPUT:



- **WAP using BinaryReader and BinaryWriter.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.Linq;
usingSystem.Text;
using System.IO;
namespacebinreadwrite
{
public class BinStream
   {
publicBinStream()
      {
Writer();
Reader();
      }

public static void Main()
      {
BinStreambs = new BinStream();
Console.ReadLine();
      }
private void Writer()
      {
try
         {
Console.Out.WriteLine("preparing to write...");
FileStreamfout = new FileStream(@"file.docx", FileMode.OpenOrCreate, FileAccess.Write,
FileShare.ReadWrite);
 BinaryWriter bw = new BinaryWriter(fout);
string name = "gurpreet";
int age = 21;
double height = 5.4;
bool single = true;
char gender = 'f';

bw.Write(name);
bw.Write(age);
bw.Write(height);
bw.Write(single);
bw.Write(gender);
bw.Close();
Console.WriteLine("data written");
Console.WriteLine();
         }
catch (IOException e)
         {
Console.WriteLine("An IO Exception Occured:" + e);
```

```
        }
    }
private void Reader()
    {
try
        {
Console.WriteLine("preparing to read...");
FileStream fin = new FileStream(@"file.docx", FileMode.Open, FileAccess.Read,
FileShare.ReadWrite);
          BinaryReader b = new BinaryReader(fin);
b.BaseStream.Seek(0, SeekOrigin.Begin);

string name = b.ReadString();
int age = b.ReadInt32();
double height = b.ReadDouble();
bool single = b.ReadBoolean();
char gender = b.ReadChar();
Console.WriteLine("Name:" + name);
Console.WriteLine("Age:" + age);
Console.WriteLine("Height:" + height);
Console.WriteLine("single:" + single);
Console.WriteLine("Gender M/F:" + gender);
b.Close();
Console.WriteLine("data Read");

Console.ReadLine();
        }
catch (IOException e)
        {
Console.WriteLine("An IO Exception Occured:" + e);
        }

    }
  }
}
```

OUTPUT:

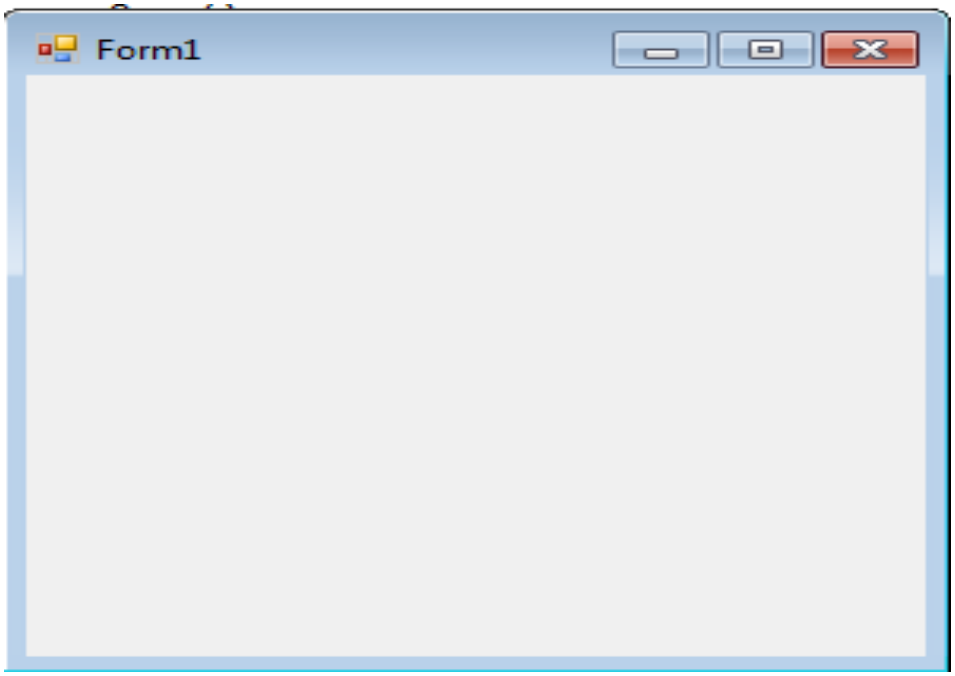

file:///c:/users/abcu/documents/visual studio 2010/Projects/ConsoleApplication1/ConsoleApplication1/bin/Debug/ConsoleApplication1.EXE
preparing to write...
data written
preparing to read...
Name:gurpreet
Age:21
Height:5.4
single:True
Gender M/F:f
data Read

- **WAP to display form using oledb connection.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.ComponentModel;
usingSystem.Data;
usingSystem.Drawing;
usingSystem.Linq;
usingSystem.Text;
usingSystem.Windows.Forms;
usingSystem.Data.OleDb;
namespaceform_oledb
{
public partial class Form1 : Form
   {
public Form1()
      {
InitializeComponent();
      }
private void Form1_Load(object sender, EventArgs e)
      {
stringconstr = null;
OleDbCommandcmd;
OleDbConnection con;
OleDbDataReader reader;
stringsql = null;
constr = @"Provider=Microsoft.ACE.OLEDB.12.0;Data Source=E:\c#\Database1.accdb;Persist
Security Info=false";
sql="select *from emp_data";
con=new OleDbConnection(constr);
try
         {
con.Open();
cmd=new OleDbCommand(sql,con);
reader=cmd.ExecuteReader();
while(reader.Read())
            {
MessageBox.Show(reader.GetValue(0)+ "-" + reader.GetValue(1)+"-"+reader.GetValue(2));
            }
reader.Close();
cmd.Dispose();
con.Close();
         }
catch(Exception ex)
        {
```

```
MessageBox.Show("can not open connection" +ex);
            }
        }
    }
}
```

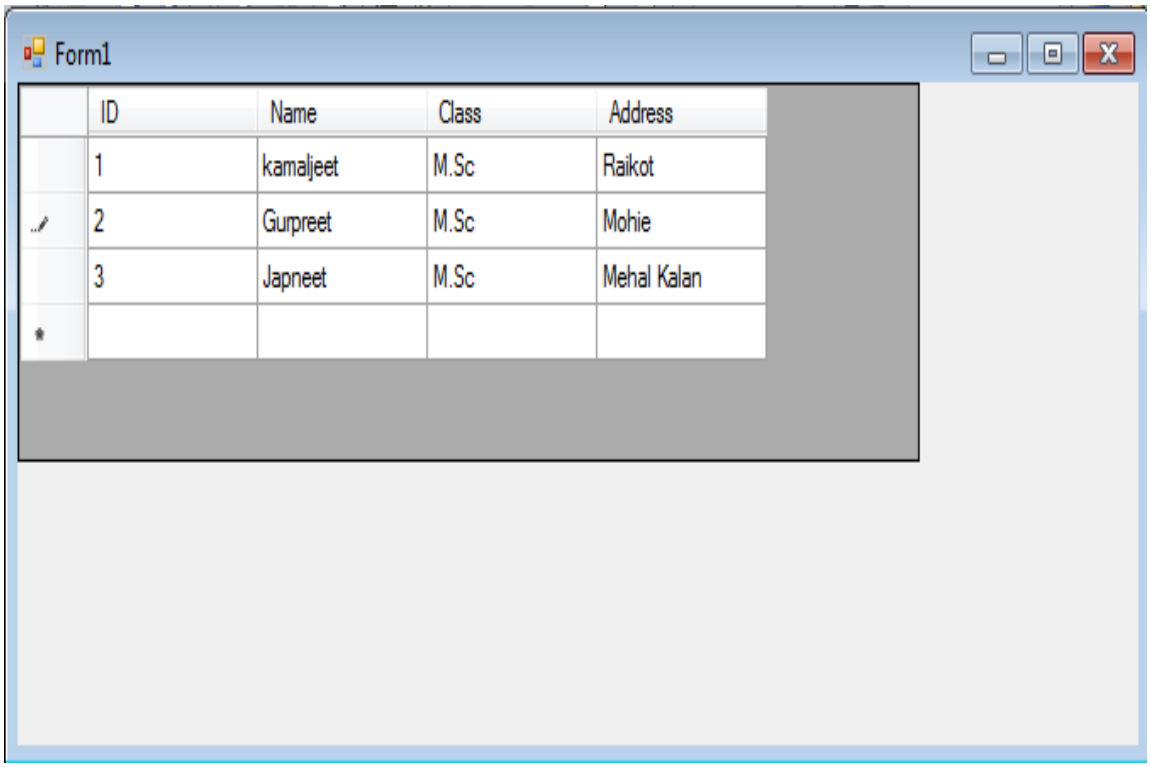
OUTPUT:



- **Create a form to display data in Gridview.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.ComponentModel;
usingSystem.Data;
usingSystem.Drawing;
usingSystem.Linq;
usingSystem.Text;
usingSystem.Windows.Forms;
usingSystem.Data.OleDb;
namespaceoledb_dataset
{
public partial class Form1 : Form
   {
public Form1()
       {
InitializeComponent();
       }

private void Form1_Load(object sender, EventArgs e)
       {

stringque = @"Provider=Microsoft.ACE.OLEDB.12.0;
Data Source=E:\c#\Database1.accdb;Persist Security Info=false";
OleDbConnection con = new OleDbConnection(que);
con.Open();
string query = "select * from emp_data";
OleDbDataAdapter da = new OleDbDataAdapter(query, con);
DataSet ds = new DataSet();
da.Fill(ds, "emp_data");
           dataGridView1.DataSource = ds.Tables["emp_data"];

       }
   }
}
```

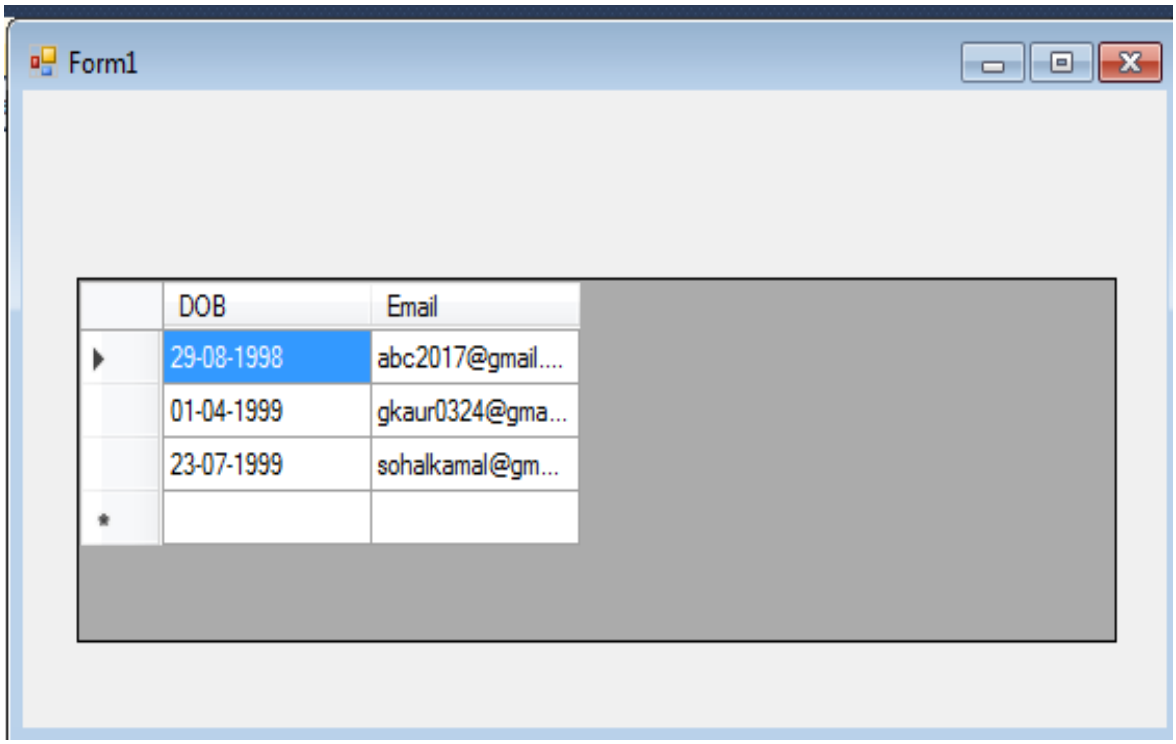
OUTPUT:


Form1
ID
Name
Class
Address
1
kamaljeet
M.Sc
Raikot
2
Gurpreet
M.Sc
Mohie
3
Japneet
M.Sc
Mehal Kalan

- **Create a form to add a column into table .**

```
using System;
usingSystem.Collections.Generic;
usingSystem.ComponentModel;
usingSystem.Data;
usingSystem.Drawing;
usingSystem.Linq;
usingSystem.Text;
usingSystem.Windows.Forms;
namespacedatatable
{
public partial class Form1 : Form
   {
public Form1()
      {
InitializeComponent();
      }

private void Form1_Load(object sender, EventArgs e)
      {
DataTable table = new DataTable();

table.Columns.Add("DOB");
table.Columns.Add("Email");
         table.Rows.Add(“29-08-1998”, “abc@gmail.com”)
table.Rows.Add("01-04-1999", "gkaur0324@gmail.com");
table.Rows.Add("23-07-1999", "sohalkamal2017@gmail.com");

         dataGridView1.DataSource = table;

      }
   }
}
```

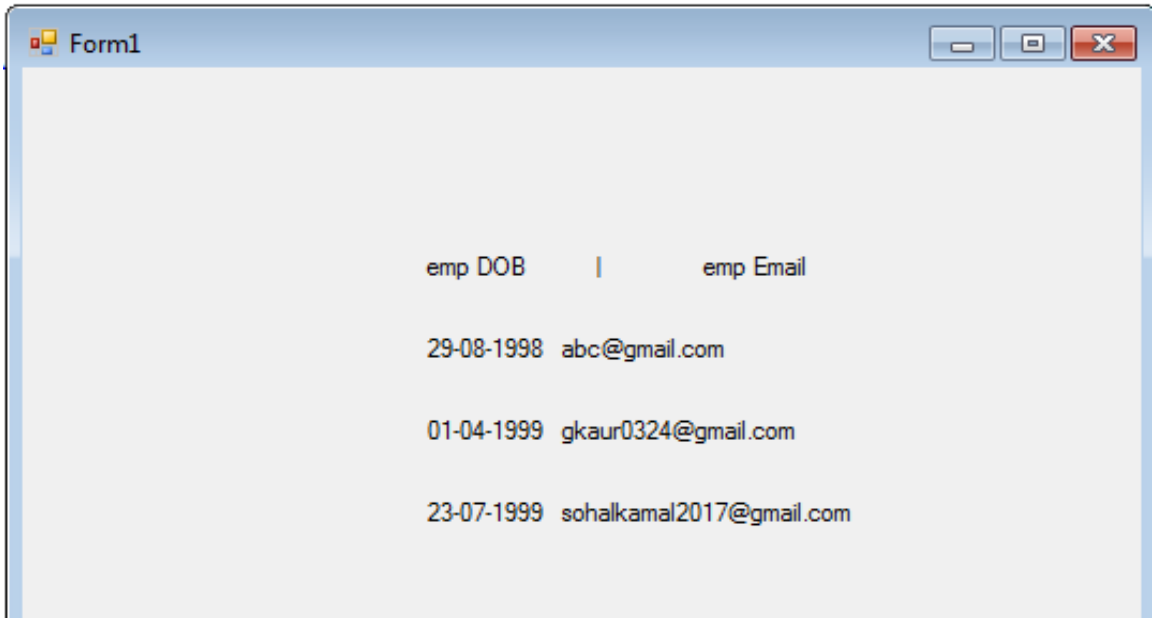
OUTPUT:


Form1
DOB
Email
29-08-1998
abc2017@gmail....
01-04-1999
gkaur0324@gma...
23-07-1999
sohalkamal@gm...

- **Create a form to show data in Lable using Oledb data table.**

```
using System;
usingSystem.Collections.Generic;
usingSystem.ComponentModel;
usingSystem.Data;
usingSystem.Drawing;
usingSystem.Linq;
usingSystem.Text;
usingSystem.Windows.Forms;
namespacedatatviewable
{
public partial class Form1 : Form
   {
public Form1()
      {
InitializeComponent();
      }

private void Form1_Load(object sender, EventArgs e)
      {
DataTable table = new DataTable();

table.Columns.Add("DOB");
table.Columns.Add("Email");
table.Rows.Add(“29-08-1998”, “abc@gmail.com”)
table.Rows.Add("01-04-1999", "gkaur0324@gmail.com");
table.Rows.Add("23-07-1999", "sohalkamal2017@gmail.com");
DataView dv = new DataView(table);
         label1.Text = "emp DOB          |              emp Email";
foreach (DataRowView view in dv)
         {
            label1.Text = label1.Text + " \n\n\n" + view["DOB"].ToString();
            label1.Text = label1.Text + "    " + view["Email"].ToString();
         }

      }
   }
}
```

OUTPUT:


Form1
emp DOB | emp Email
29-08-1998 abc@gmail.com
01-04-1999 gkaur0324@gmail.com
23-07-1999 sohalkamal2017@gmail.com