PROGRAM-1

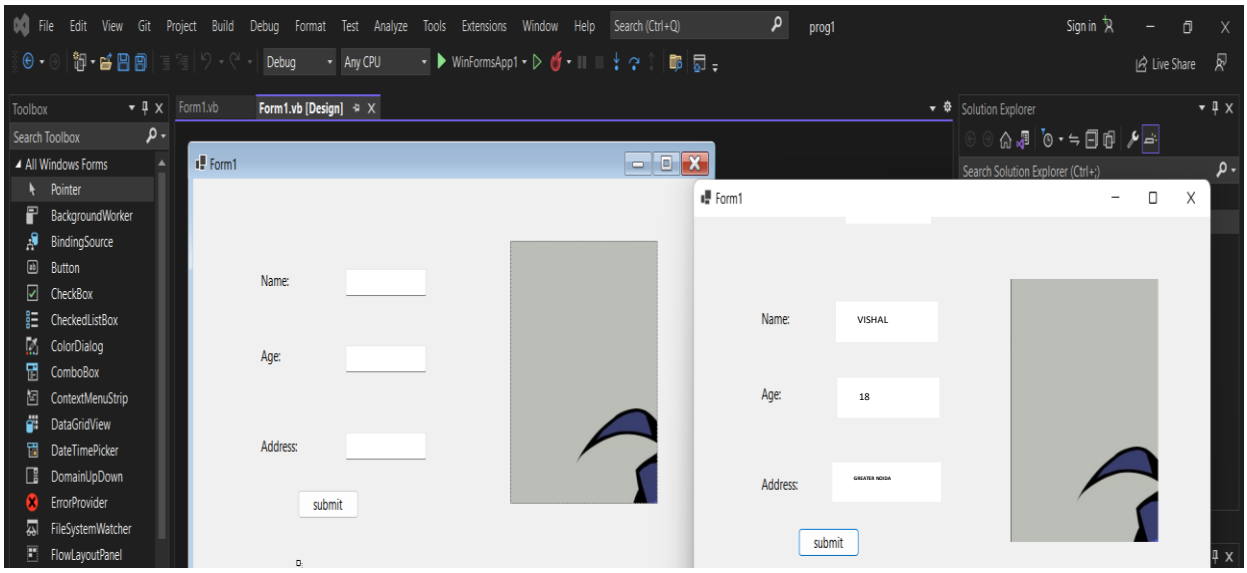
OBJECTIVE:

Write a program to print name, age, address and image on the click of submit button.

CODING:

```
PublicClass Form1

PrivateSub Label2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label2.Click
EndSub

ProtectedSub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Label4.Visible = True
        Label4.Text = " hi..!!! " + TextBox1.Text + " your age is " +
TextBox2.Text + " your address is " + TextBox3.Text
EndSub

PrivateSub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) HandlesMyBase.Load
EndSub

PrivateSub PictureBox1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox1.Click
EndSub

PrivateSub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox1.TextChanged
EndSub

PrivateSub Label1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label1.Click
EndSub

PrivateSub TextBox2_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox2.TextChanged
EndSub

PrivateSub Label3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label3.Click
EndSub

PrivateSub TextBox3_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox3.TextChanged
EndSub

EndClass
```

OUTPUT:


File Edit View Git Project Build Debug Format Test Analyze Tools Extensions Window Help
Search (Ctrl+Q)
prog1
Sign in
Debug
Any CPU
WinFormsApp1
Live Share
Toolbox
Search Toolbox
All Windows Forms
Pointer
BackgroundWorker
BindingSource
Button
CheckBox
CheckedListBox
ColorDialog
ComboBox
ContextMenuStrip
DataGridView
DateTimePicker
DomainUpDown
ErrorProvider
FileSystemWatcher
FlowLayoutPanel
Form1.vb
Form1.vb [Design]
Form1
Name:
Age:
Address:
submit
Solution Explorer
Search Solution Explorer (Ctrl+;)
Form1
Name:
VISHAL
Age:
18
Address:
GREATER NOIDA
submit

PROGRAM-2

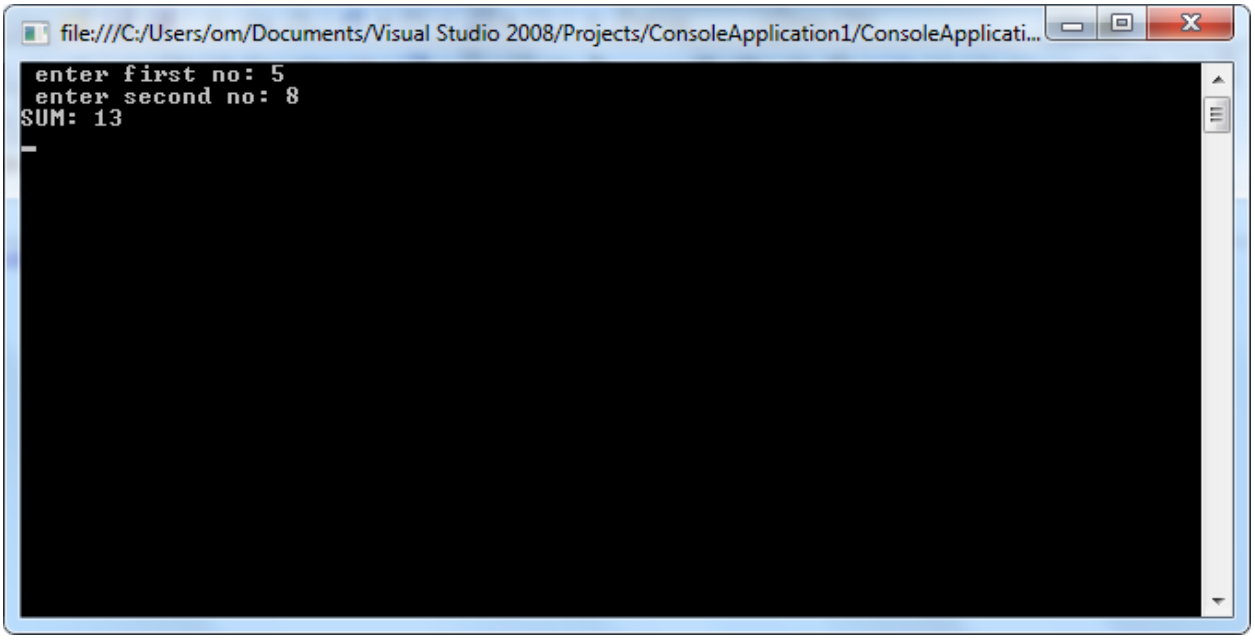
OBJECTIVE:

Write a program to calculate the sum of two numbers in console application and windows form application.

CODING (console application):

```
Module Module1

Sub Main()
Dim a, b AsInteger
        Console.Write(" enter first no: ")
        a = Console.ReadLine()
        Console.Write(" enter second no: ")
        b = Console.ReadLine()
        Console.Write("SUM: ")
        Console.WriteLine(a + b)
        Console.ReadLine()
EndSub

EndModule
```

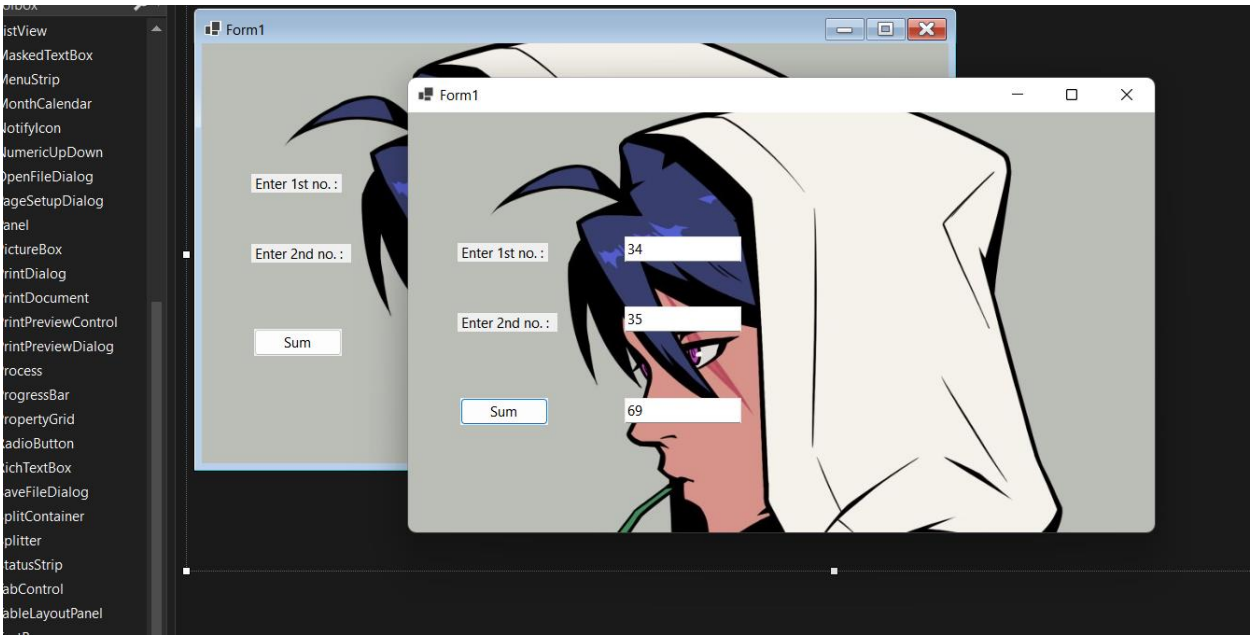
OUTPUT:



CODING(windows form application):

```
PublicClass Form1

PrivateSub Label2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label2.Click
EndSub

PrivateSub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Dim a As Integer = CInt(TextBox1.Text)
Dim b As Integer = CInt(TextBox2.Text)
Dim c As Integer
        c = a + b
        TextBox3.Text = c
EndSub
PrivateSub Label1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label1.Click
EndSub

PrivateSub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox1.TextChanged
EndSub

PrivateSub TextBox2_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox2.TextChanged
EndSub

PrivateSub TextBox3_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox3.TextChanged
EndSub
EndClass
```

OUTPUT:



PROGRAM-3

OBJECTIVE:

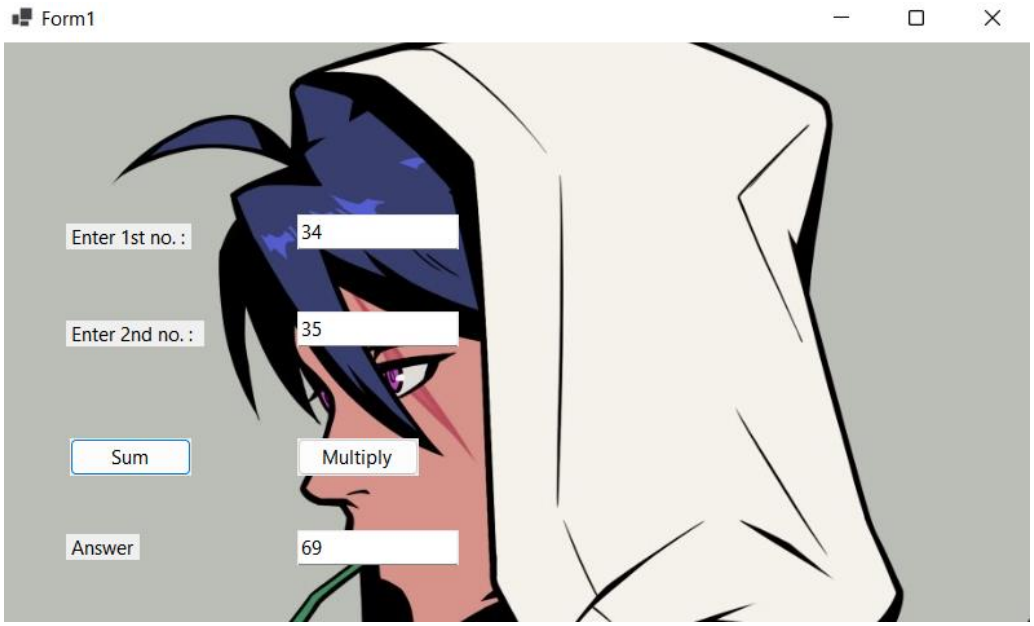
Write a program to calculate the sum and multiply of two numbers in windows form application.

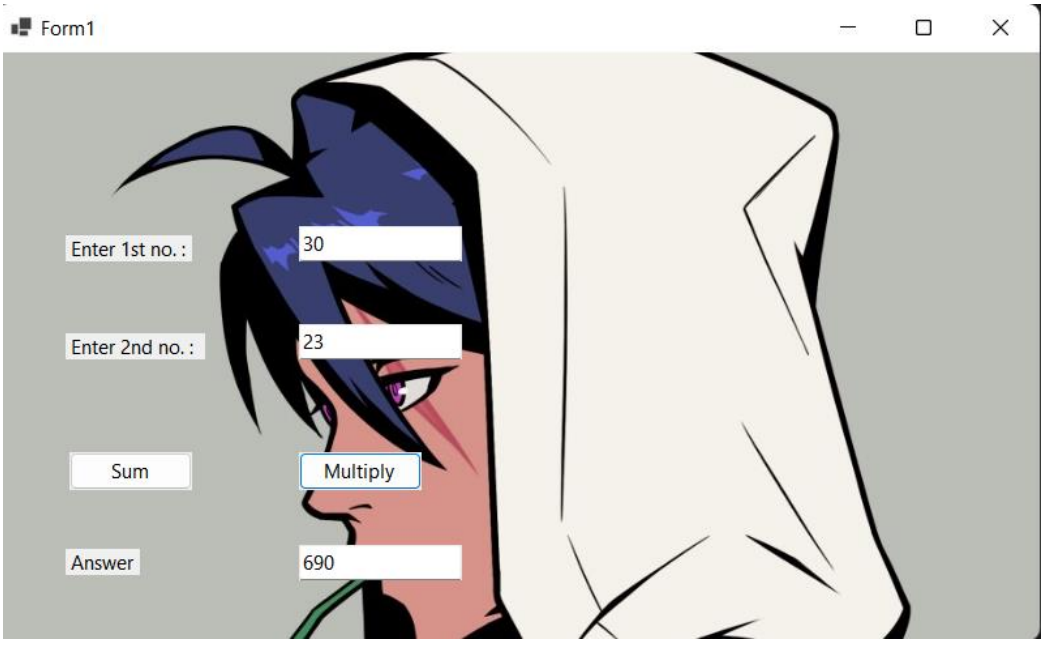
CODING:

```
PublicClass Form1

PrivateSub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
Dim a AsInteger = CInt(TextBox1.Text)
Dim b AsInteger = CInt(TextBox2.Text)
Dim d AsInteger
        d = a * b
        TextBox3.Text = d
EndSub

PrivateSub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Dim a AsInteger = CInt(TextBox1.Text)
Dim b AsInteger = CInt(TextBox2.Text)
Dim c AsInteger
        c = a + b
        TextBox3.Text = c
EndSub

PrivateSub Label1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label1.Click

EndSub

PrivateSub Label2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label2.Click

EndSub

PrivateSub Label3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label3.Click

EndSub

PrivateSub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox1.TextChanged

EndSub

PrivateSub TextBox2_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox2.TextChanged

EndSub

PrivateSub TextBox3_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox3.TextChanged

EndSub
```

```
EndClass
```

OUTPUT:





PROGRAM-4

OBJECTIVE:

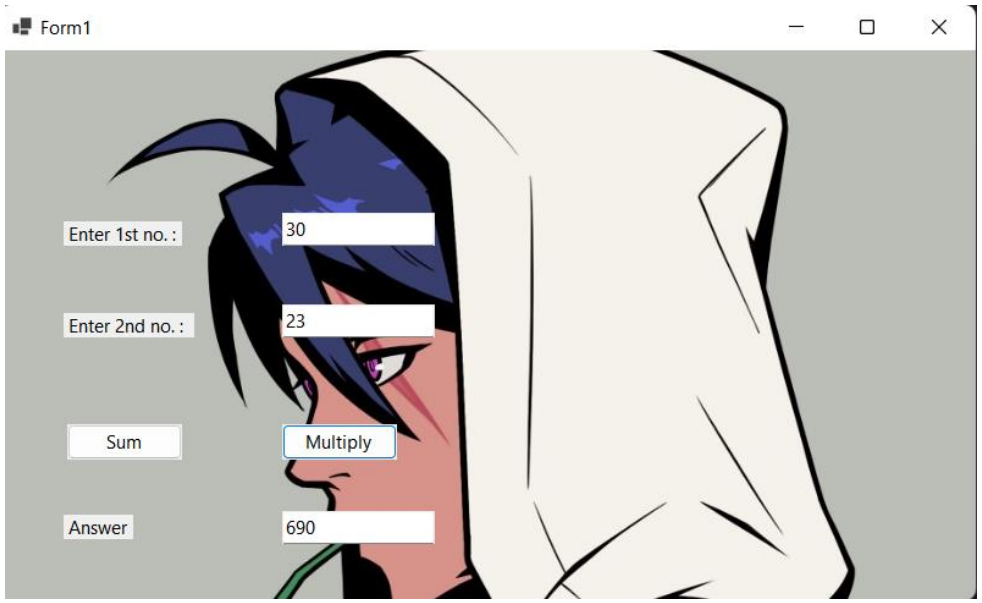
Write a program to create the calculator.

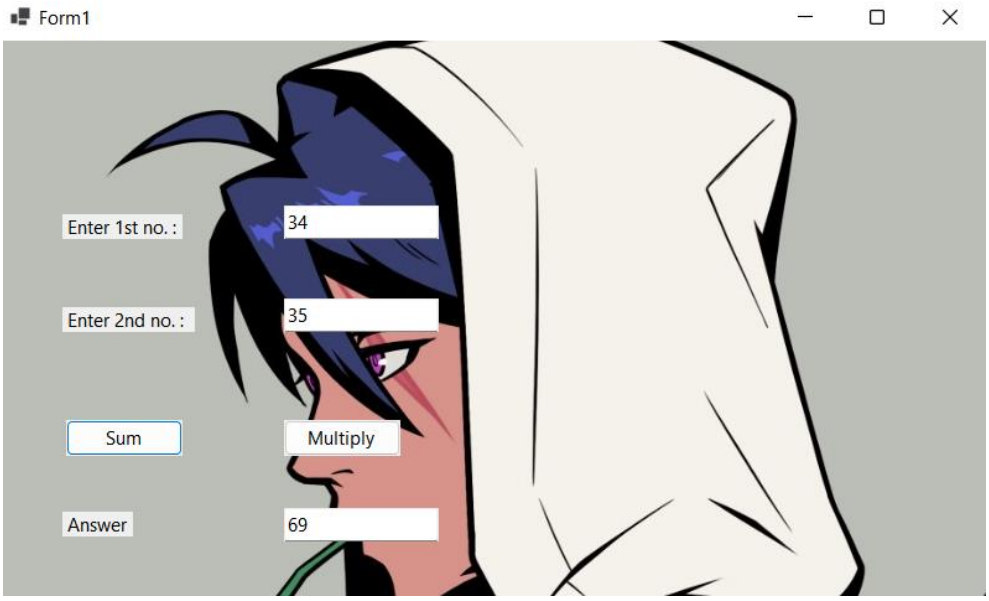
CODING:

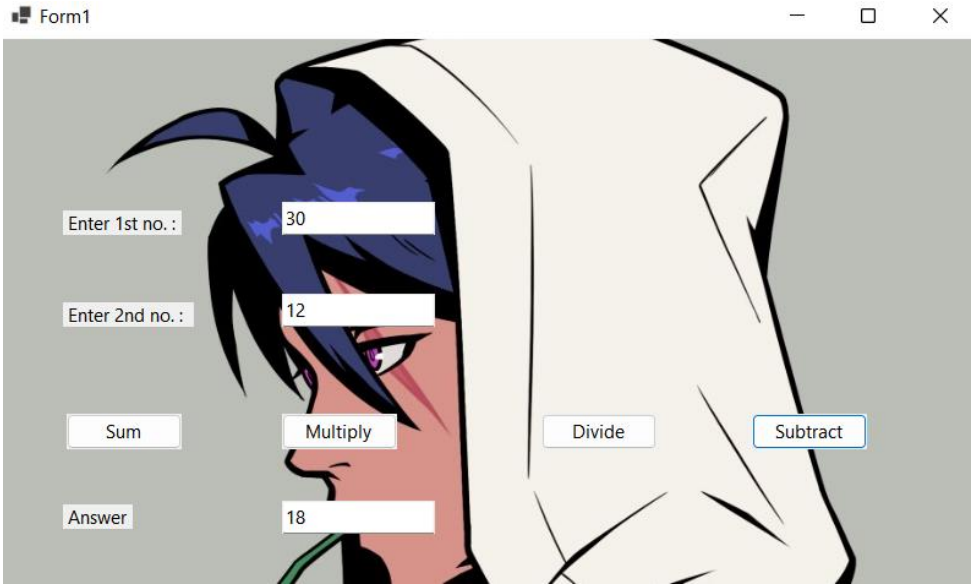
```
PublicClass Form1
PrivateSub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
Dim a AsInteger = CInt(TextBox1.Text)
Dim b AsInteger = CInt(TextBox2.Text)
Dim c AsInteger
        c = a * b
        TextBox3.Text = c
EndSub

PrivateSub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Dim a AsInteger = CInt(TextBox1.Text)
Dim b AsInteger = CInt(TextBox2.Text)
Dim c AsInteger
        c = a + b
        TextBox3.Text = c
EndSub

PrivateSub Label1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label1.Click
EndSub

PrivateSub Label2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label2.Click
EndSub

PrivateSub Label3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label3.Click
EndSub

PrivateSub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox1.TextChanged
EndSub

PrivateSub TextBox2_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox2.TextChanged
EndSub

PrivateSub TextBox3_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox3.TextChanged
EndSub

PrivateSub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
Dim a AsInteger = CInt(TextBox1.Text)
Dim b AsInteger = CInt(TextBox2.Text)
Dim c AsInteger
        c = a - b
        TextBox3.Text = c
EndSub
PrivateSub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
Dim a AsInteger = CInt(TextBox1.Text)
Dim b AsInteger = CInt(TextBox2.Text)
```

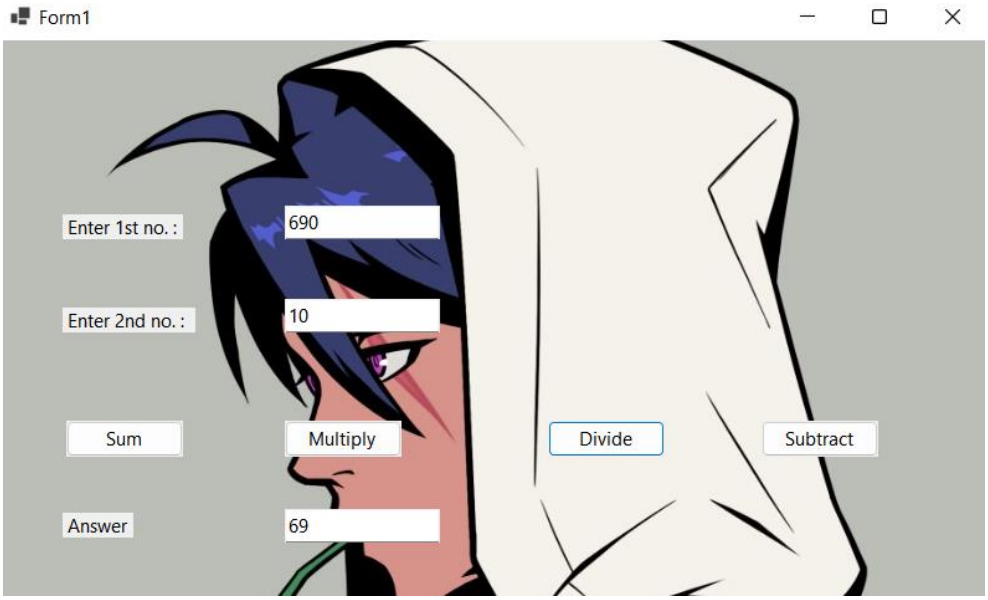
```
Dim c AsInteger
        c = a / b
        TextBox3.Text = c
EndSub
EndClass
```

OUTPUT:






Form1
Enter 1st no. :
30
Enter 2nd no. :
12
Sum
Multiply
Divide
Subtract
Answer
18


Form1
Enter 1st no. :
690
Enter 2nd no. :
10
Sum
Multiply
Divide
Subtract
Answer
69

PROGRAM-5

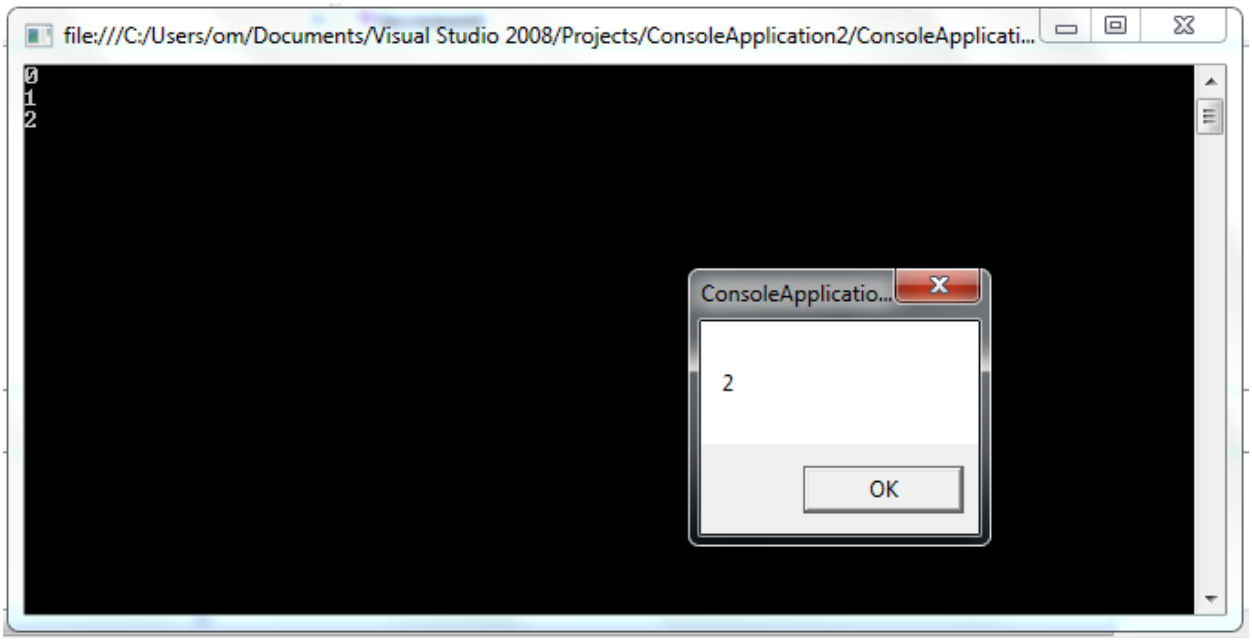
OBJECTIVE:

Write a program to access function in module one from another module.

CODING:

```
Module Module1

Sub Main()
'use module2
'it does not need to be created.
        module2.increment()
        module2.increment()
        module2.increment()
EndSub

EndModule
Module module2
Dim _value AsInteger
Sub increment()
'the value is shared.
        Console.WriteLine(_value)
        MsgBox(_value)
'change the value.
        _value += 1
EndSub
EndModule
```

OUTPUT:



PROGRAM-6

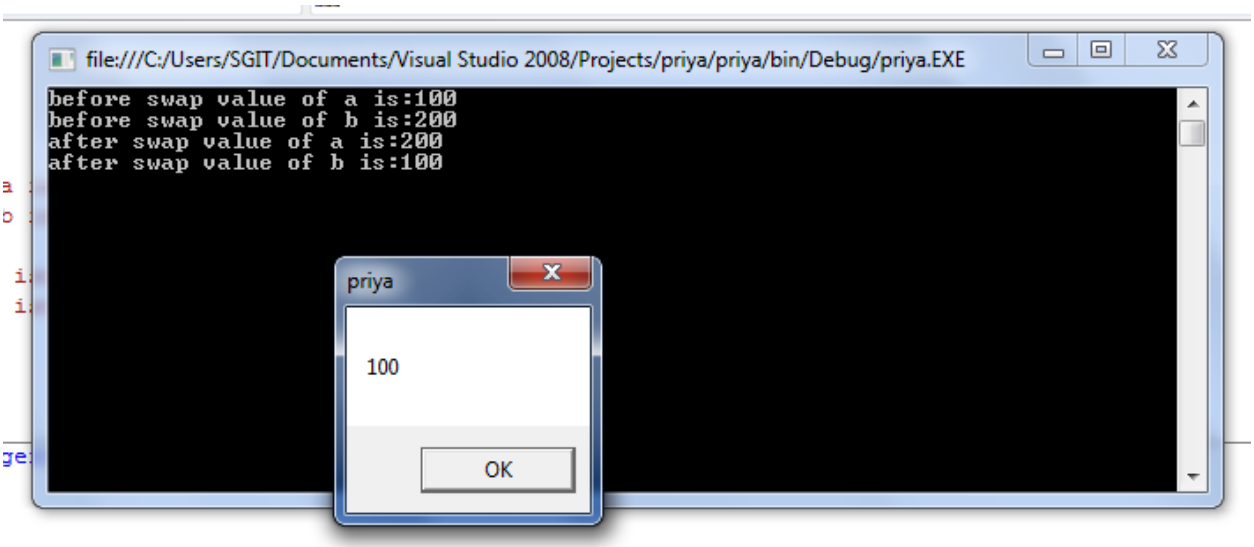
OBJECTIVE:

Write a program to swap two numbers using value of the parameters (pass by reference).

CODING:

```
Module Module1

Sub Main()
Dim a AsInteger = 100
Dim b AsInteger = 200
        Console.WriteLine("before swap value of a is:{0}", a)
        Console.WriteLine("before swap value of b is:{0}", b)
        swap(a, b)
        Console.WriteLine("after swap value of a is:{0}", a)
        Console.WriteLine("after swap value of b is:{0}", b)
        MsgBox(a)
        MsgBox(b)
EndSub
Sub swap(ByRef x AsInteger, ByRef y AsInteger)
Dim temp AsInteger
        temp = x
        x = y
        y = temp
EndSub
EndModule
```

OUTPUT:



PROGRAM-7

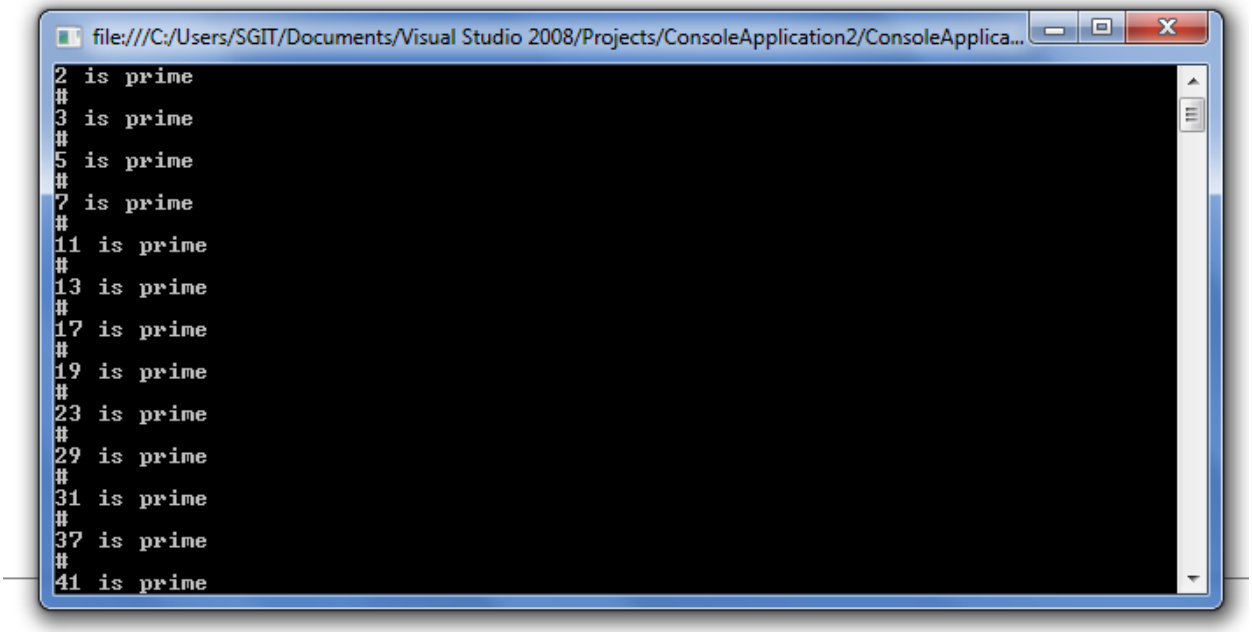
OBJECTIVE:

Write a program to find prime numbers from 2 to 100 using nesting of for loop.

CODING:

```
Module Module1

Sub Main()
Dim i, j AsInteger
For i = 2 To 100
For j = 2 To i
If ((i Mod j) = 0) Then
ExitFor
EndIf
Next j
If (j > (i \ j)) Then
                Console.WriteLine("#")
                Console.WriteLine("{0} is prime", i)
EndIf
Next i
        Console.ReadLine()

EndSub

EndModule
```

OUTPUT:



PROGRAM-8

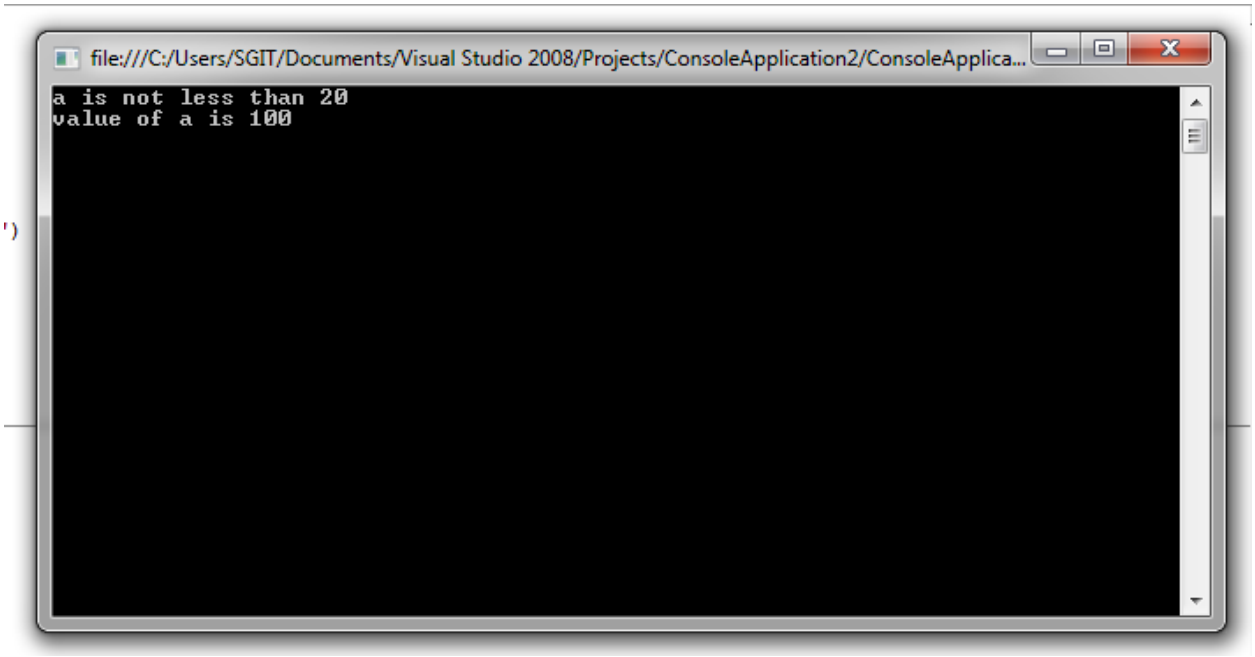
OBJECTIVE:

Write a program to check the number is less or more than a number using If & Else.

CODING:

```
Module module1

Sub Main()
Dim a AsInteger = 100
If (a < 20) Then
            Console.WriteLine("a is less than 20")
Else
            Console.WriteLine("a is not less than 20")
EndIf
        Console.WriteLine("value of a is {0}", a)
        Console.ReadLine()
EndSub
EndModule
```

OUTPUT:



PROGRAM-9

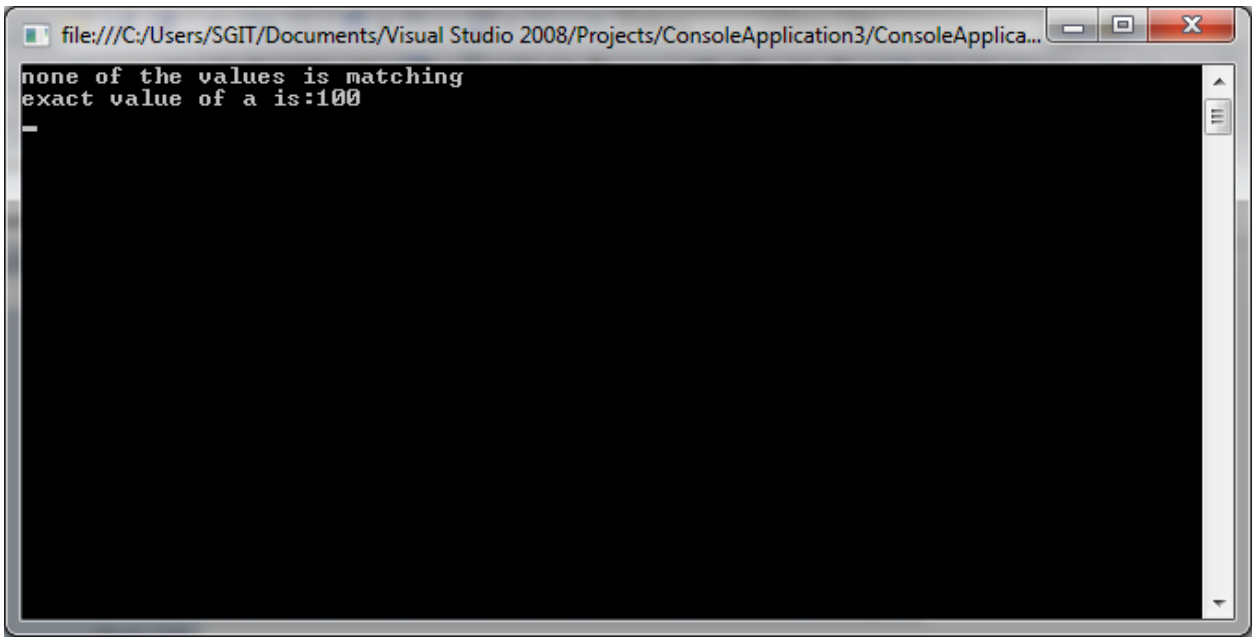
OBJECTIVE:

Write a program to check the number range using If, Else & ElseIf.

CODING:

```
Module Module1

Sub Main()
Dim a AsInteger = 100
If (a = 10) Then
            Console.WriteLine("value of a is 10")
ElseIf (a = 20) Then
            Console.WriteLine("value of a is 20")
ElseIf (a = 30) Then
            Console.WriteLine("value of a is 30")
Else
            Console.WriteLine("none of the values is matching")
EndIf
        Console.WriteLine("exact value of a is:{0}", a)
        Console.ReadLine()
EndSub

EndModule
```

OUTPUT:



PROGRAM-10

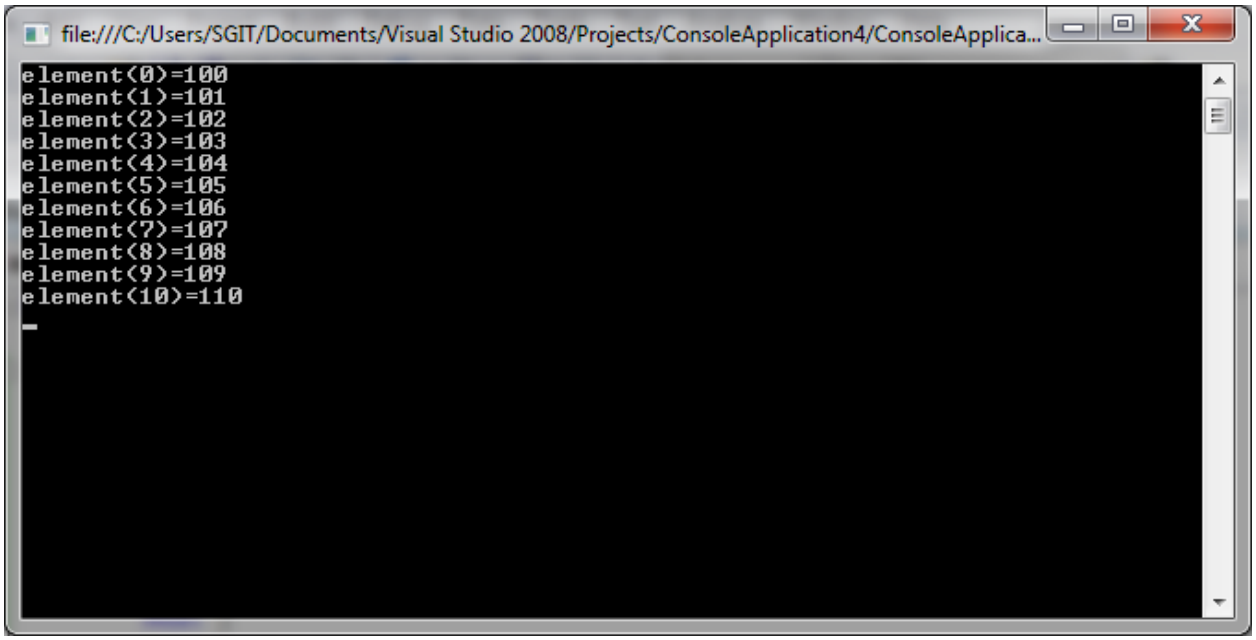
OBJECTIVE:

Write a program to print the values of an array of size[10].

CODING:

```
Module Module1

Sub Main()
Dim n(10) AsInteger
Dim i, j AsInteger
For i = 0 To 10
            n(i) = i + 100
Next i
For j = 0 To 10
            Console.WriteLine("element({0})={1}", j, n(j))
Next j
        Console.ReadKey()
EndSub

EndModule
```

OUTPUT:



PROGRAM-11

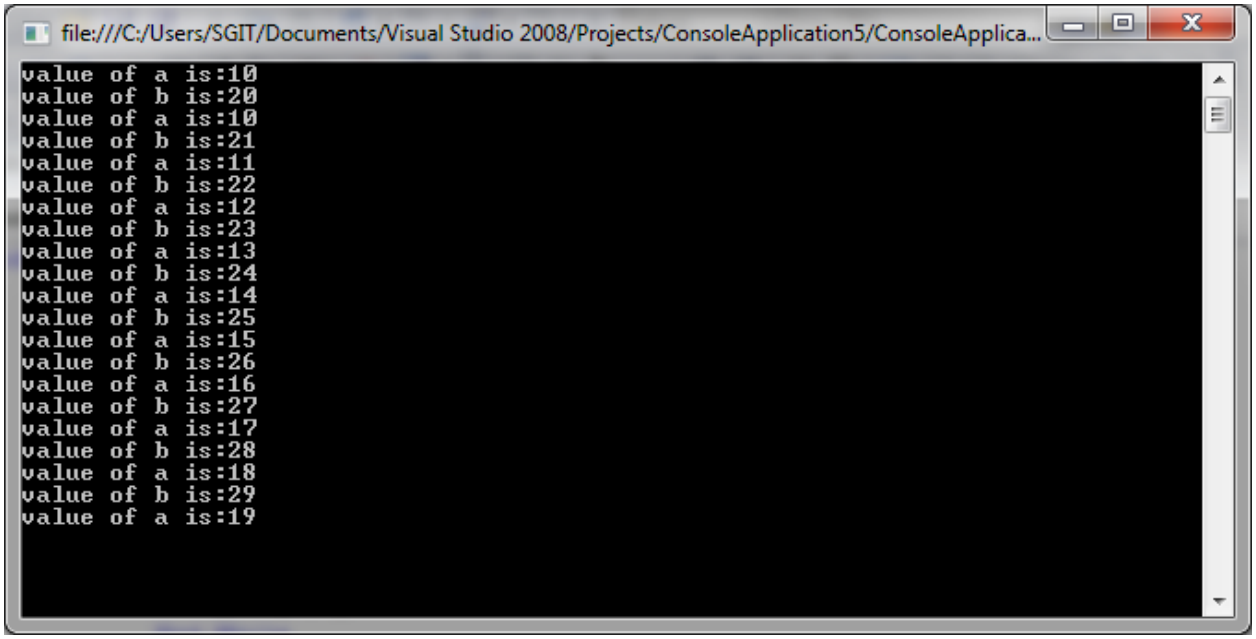
OBJECTIVE:

Write a program to print the value of two numbers alternatively using while loop.

CODING:

```
Module Module1

Sub Main()
Dim a AsInteger = 10
Dim b AsInteger = 20
While a < 20
            Console.WriteLine("value of a is:{0}", a)
While b < 30
                Console.WriteLine("value of b is:{0}", b)
                b = b + 1
                Console.WriteLine("value of a is:{0}", a)
                a = a + 1
EndWhile
EndWhile
        Console.ReadLine()

EndSub

EndModule
```

OUTPUT:



PROGRAM-12

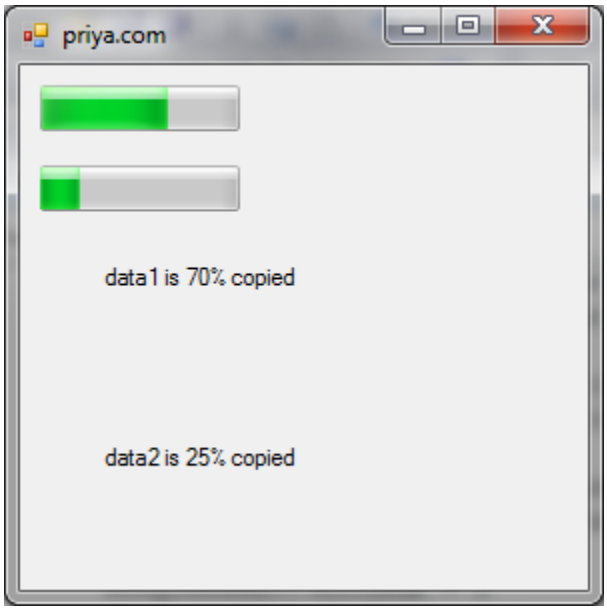
OBJECTIVE:

Write a program to print process status of progress bar.

CODING:

```
PublicClass Form1
PrivateSub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) HandlesMyBase.Load
Dim ProgressBar1 As ProgressBar
Dim ProgressBar2 As ProgressBar
        ProgressBar1 = New ProgressBar()
        ProgressBar2 = New ProgressBar()
        ProgressBar1.Location = New Point(10, 10)
        ProgressBar2.Location = New Point(10, 50)
        ProgressBar1.Minimum = 0
        ProgressBar1.Maximum = 200
        ProgressBar1.Value = 130
        ProgressBar2.Minimum = 0
        ProgressBar2.Maximum = 200
        ProgressBar2.Value = 40
Me.Controls.Add(ProgressBar1)
Me.Controls.Add(ProgressBar2)
Me.Text = "priya.com"
        Label1.Text = "data1 is 70% copied"
        Label2.Text = "data1 is 25% copied"
EndSub
    PrivateSub Label1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label1.Click
    EndSub
PrivateSub Label2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label2.Click
EndSub
EndClass
```

OUTPUT:



PROGRAM-13

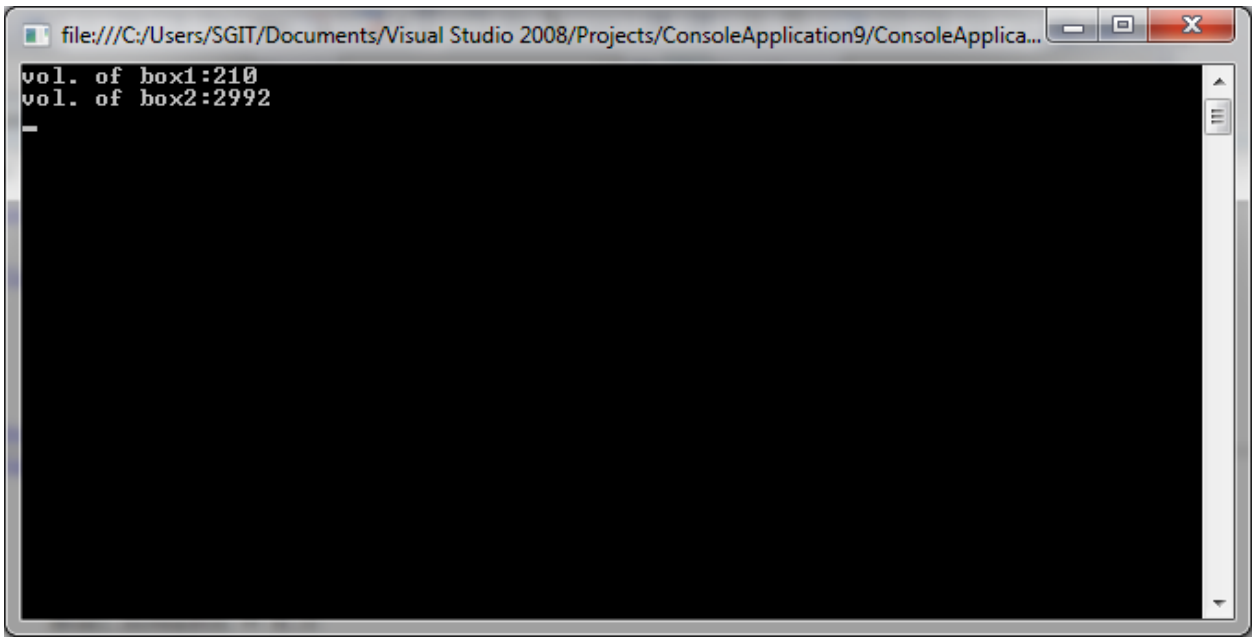
OBJECTIVE:

Write a program to make a class a object.

CODING:

```
Module Module1
Class box
Public length AsDouble
Public breadth AsDouble
Public height AsDouble
EndClass
Sub Main()
Dim Box1 As box = New box()
Dim Box2 As box = New box()
Dim volume AsDouble = 0.0
        Box1.length = 5.0
        Box1.breadth = 6.0
        Box1.height = 7.0
        Box2.length = 11.0
        Box2.breadth = 16.0
        Box2.height = 17.0
        volume = Box1.length * Box1.height * Box1.breadth
        Console.WriteLine("vol. of box1:{0}", volume)
        volume = Box2.length * Box2.height * Box2.breadth
        Console.WriteLine("vol. of box2:{0}", volume)
        Console.ReadKey()
EndSub
EndModule
```

OUTPUT:



PROGRAM-14

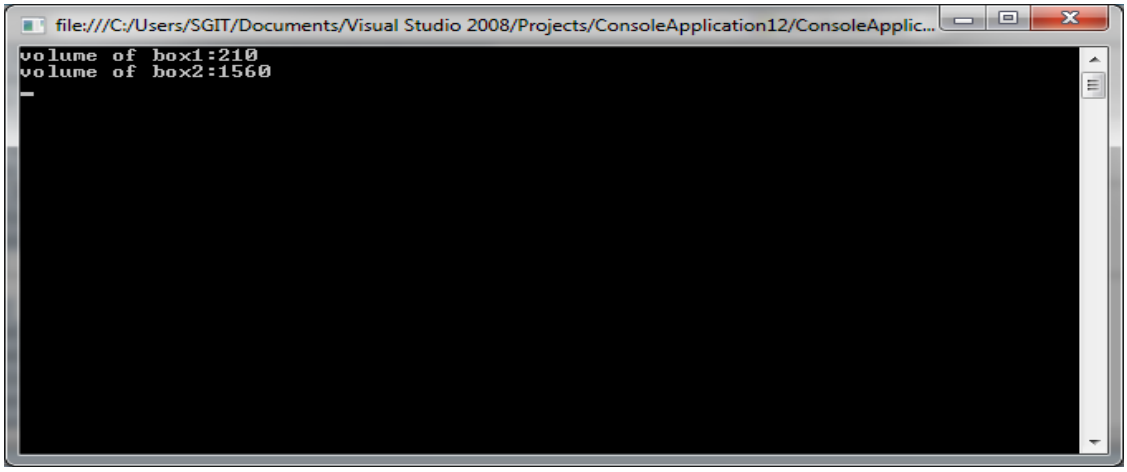
OBJECTIVE:

Write a program to make a class a object.

CODING:

```
Module Module1

Class Box
Public length AsDouble
Public breadth AsDouble
Public height AsDouble
PublicSub setLength(ByVal len AsDouble)
            length = len
EndSub
PublicSub setBreadth(ByVal bre AsDouble)
            breadth = bre
EndSub
PublicSub setHeight(ByVal hei AsDouble)
            height = hei
EndSub
PublicFunction getVolume() AsDouble
Return length * breadth * height
EndFunction
EndClass

Sub Main()
Dim Box1 As Box = New Box()
Dim Box2 As Box = New Box()
Dim volume AsDouble = 0.0
        Box1.setLength(6.0)
        Box1.setBreadth(7.0)
        Box1.setHeight(5.0)
        Box2.setLength(12.0)
        Box2.setBreadth(13.0)
        Box2.setHeight(10.0)
        volume = Box1.getVolume()
        Console.WriteLine("volume of box1:{0}", volume)
        volume = Box2.getvolume()
        Console.WriteLine("volume of box2:{0}", volume)
        Console.ReadKey()


EndSub

EndModule
```

OUTPUT:


file:///C:/Users/SGIT/Documents/Visual Studio 2008/Projects/ConsoleApplication12/ConsoleApplic...
volume of box1:210
volume of box2:1560

PROGRAM-15

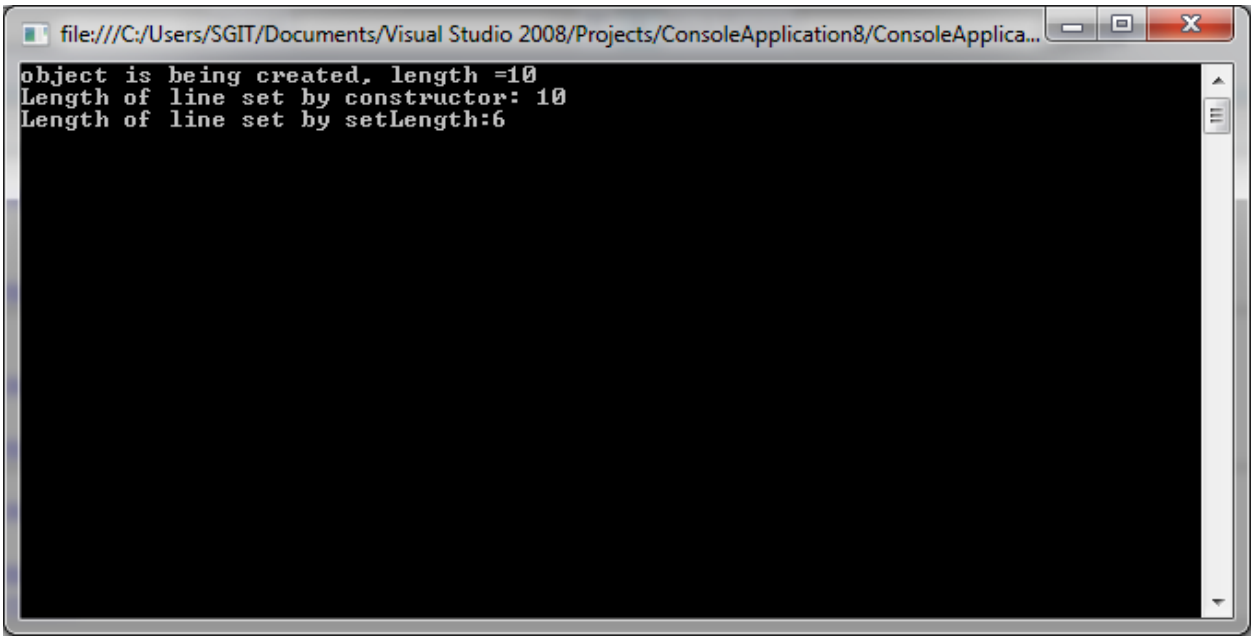
OBJECTIVE:

Write a program to call a function using parameterized constructor.

CODING:

```
Module Module1
Class Line
Private length AsDouble
        PublicSubNew(ByVal len AsDouble)
            Console.WriteLine("object is being created, length ={0}", len)
            length = len
EndSub
        PublicSub setLength(ByVal len AsDouble)
            length = len
EndSub
        PublicFunction getLength() AsDouble
Return length
EndFunction
EndClass
      SharedSub Main()
Dim line As Line = New Line(10.0)
        Console.WriteLine("Length of line set by constructor: {0}",
line.getLength())
  line.setLength(6.0)
        Console.WriteLine("Length of line set by setLength:{0}",
line.getLength())
        Console.ReadKey()
EndSub
EndModule
```

OUTPUT:



PROGRAM-16

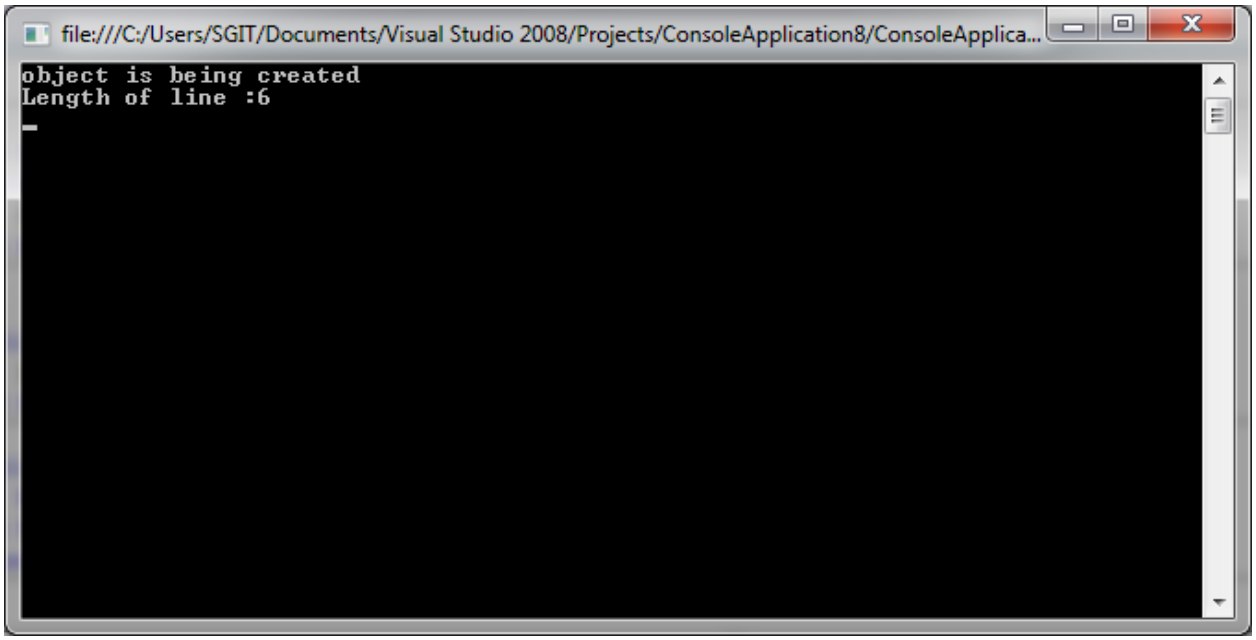
OBJECTIVE:

Write a program to call a function using destructor.

CODING:

```
Module Module1
Class Line
Private length AsDouble
PublicSubNew()
            Console.WriteLine("object is being created")
EndSub
ProtectedOverridesSub Finalize()
            Console.WriteLine("object is being deleted")
EndSub
        PublicSub setLength(ByVal len AsDouble)
            length = len
EndSub
        PublicFunction getLength() AsDouble
Return length
EndFunction
EndClass
    Sub Main()
Dim line As Line = New Line()
        line.setLength(6.0)
        Console.WriteLine("Length of line :{0}", line.getLength())
        Console.ReadKey()
EndSub
EndModule
```

OUTPUT:



PROGRAM-17

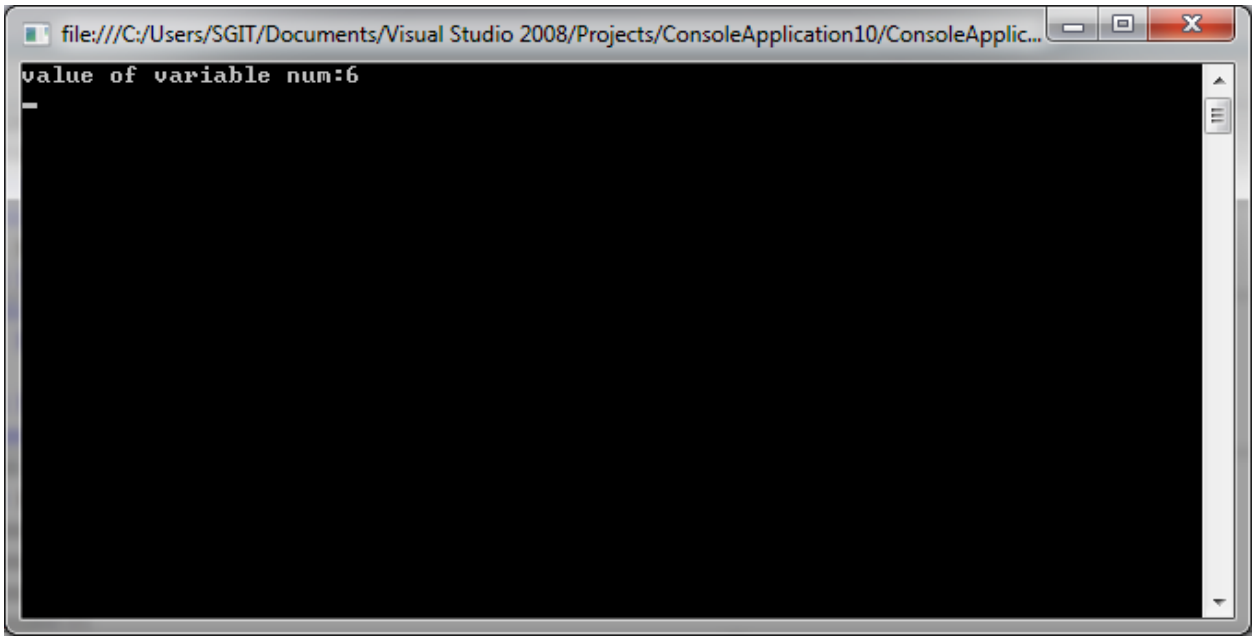
OBJECTIVE:

Write a program to define a class member as static using shared members.

CODING:

```
Module Module1
Class staticvar
PublicShared num AsInteger
PublicSub count()
            num = num + 1
EndSub
PublicSharedFunction getnum() AsInteger
Return num
EndFunction
EndClass
Sub Main()
Dim s As staticvar = New staticvar()
        s.count()
        s.count()
        s.count()
        s.count()
        s.count()
        s.count()
        Console.WriteLine("value of variable num:{0}", staticvar.getnum())
        Console.ReadKey()

EndSub

EndModule
```

OUTPUT:



PROGRAM-18

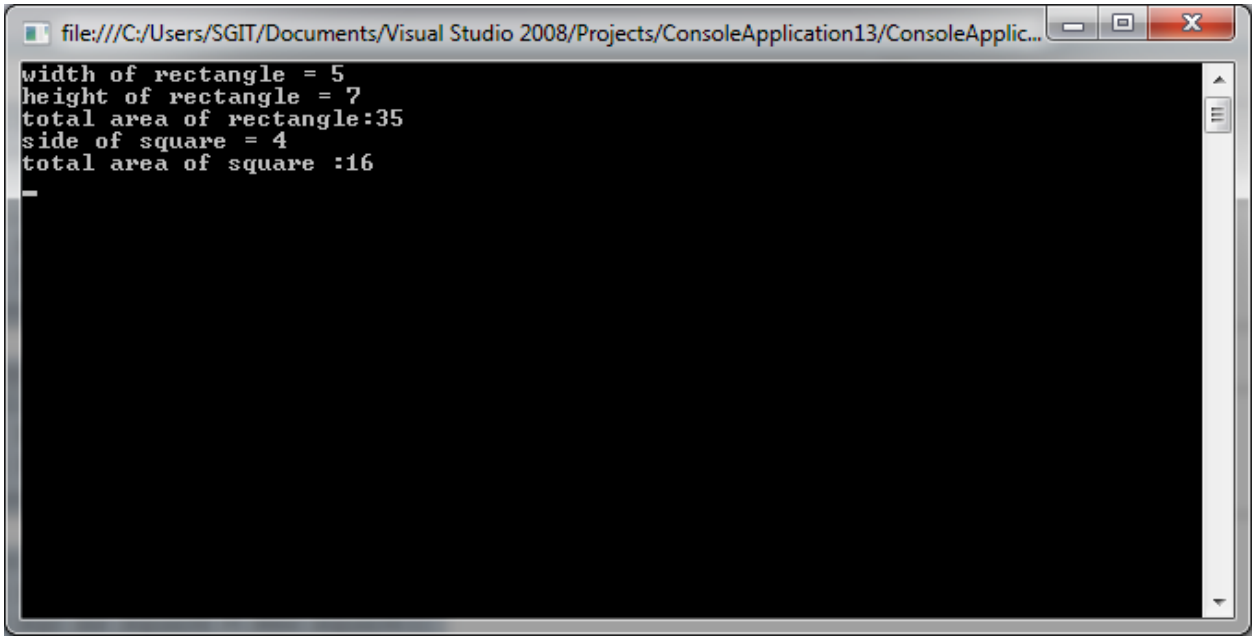
OBJECTIVE:

Write a program for multiple inheritance and acess specifiers .

CODING:

```
Module Module1
    Class shape
Protected width AsInteger
Protected height AsInteger
Protected side AsInteger
        PublicSub setWidth(ByVal w AsInteger)
            width = w
EndSub
PublicSub setHeight(ByVal h AsInteger)
            height = h
EndSub
PublicSub setSize(ByVal s AsInteger)
            side = s

EndSub
EndClass
Class rectangle : Inherits shape
PublicFunction getarea() AsInteger
Return (width * height)
EndFunction
EndClass

Class square : Inherits shape
PublicFunction getdata() AsInteger
Return (side * side)
EndFunction
EndClass

Sub Main()
Dim rect As rectangle = New rectangle()
Dim sqr As square = New square()
        rect.setWidth(5)
        rect.setHeight(7)
        sqr.setSize(4)
        Console.WriteLine("width of rectangle = 5")
        Console.WriteLine("height of rectangle = 7")

        Console.WriteLine("total area of rectangle:{0}", rect.getarea())
        Console.WriteLine("side of square = 4")
        Console.WriteLine("total area of square :{0}", sqr.getdata())
        Console.ReadKey()
EndSub

EndModule
```

OUTPUT:


file:///C:/Users/SGIT/Documents/Visual Studio 2008/Projects/ConsoleApplication13/ConsoleApplic...
width of rectangle = 5
height of rectangle = 7
total area of rectangle:35
side of square = 4
total area of square :16

PROGRAM-19

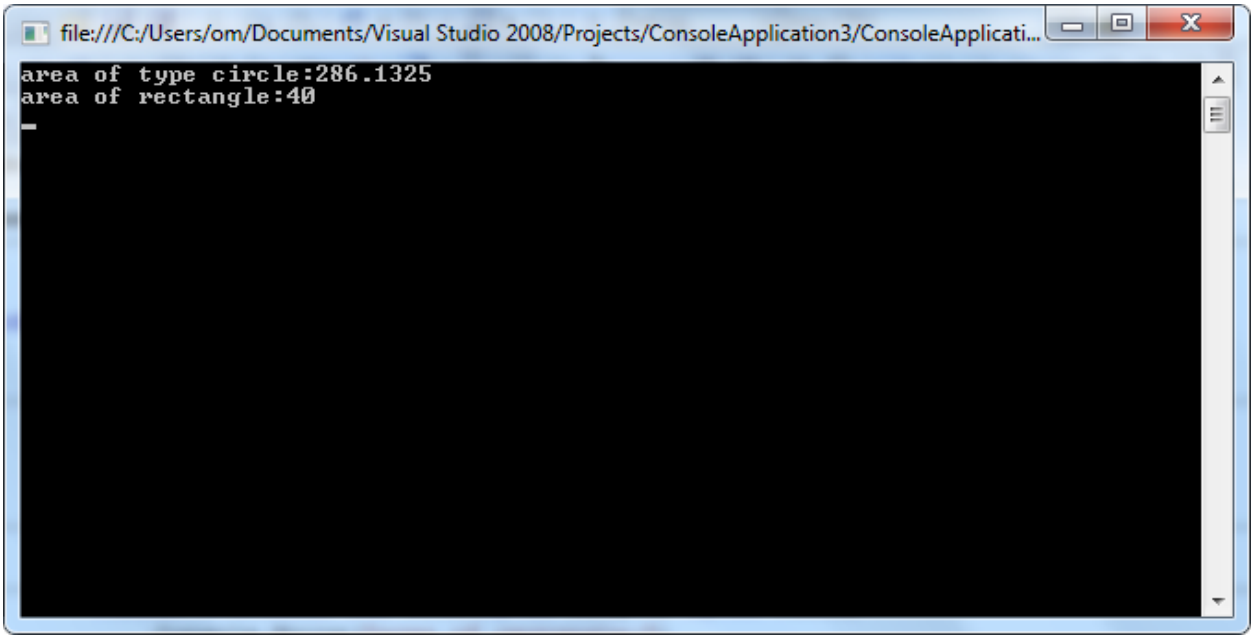
OBJECTIVE:

Write a program to show the method overloading property of vb.net .

CODING:

```
Module Module1
Class overload
Dim r AsDouble
PublicOverloadsSub area(ByVal r)
            Console.Write("area of type circle:")
            Console.WriteLine(3.14 * r * r * r)
EndSub
Dim length AsInteger
Dim width AsInteger
PublicOverloadsSub area(ByVal length, ByVal width)
            Console.Write("area of rectangle:")
            Console.WriteLine(length * width)
EndSub
EndClass
Sub Main()
Dim o AsNew overload()
        o.area(4.5)
        o.area(5, 8)
        Console.ReadLine()
EndSub
EndModule
```

OUTPUT:



PROGRAM-20

OBJECTIVE:

Write a program to show the method overriding property of vb.net .

CODING:

```
Module Module1
Class c1
OverridableSub hi()
            Console.WriteLine("old method hi")
EndSub
EndClass

Class c2 : Inherits c1
OverridesSub hi()
            Console.WriteLine("new and improved hi")
EndSub
EndClass
Sub Main()
Dim old AsNew c1()
Dim improved AsNew c2()
        old.hi()
        improved.hi()
        Console.ReadLine()

EndSub

EndModule
```

OUTPUT: