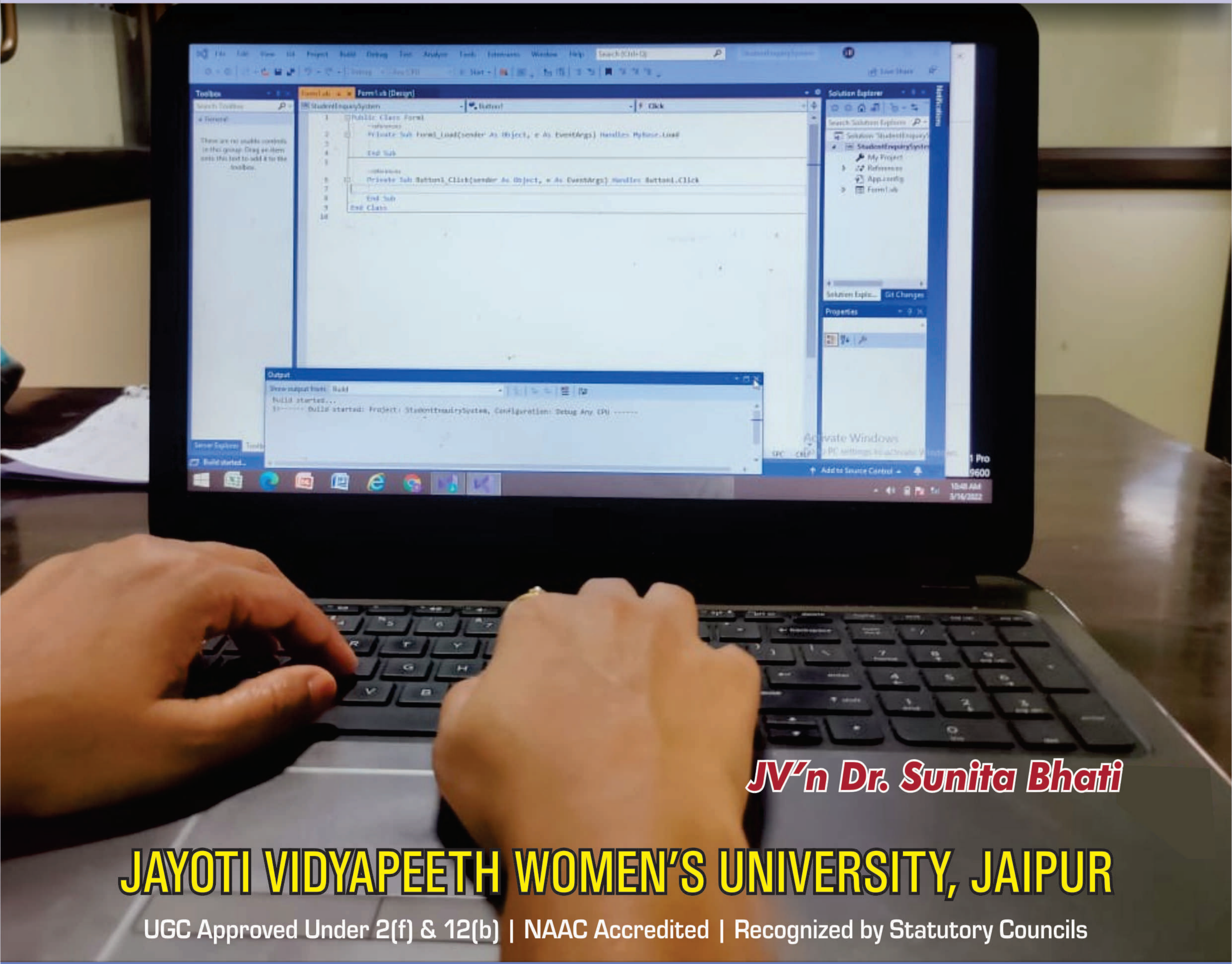
Women University Press
Jayoti Publication Desk
ISBN No. : 978-93-94024-89-2

. NET Programming

-LAB MANUAL



JV'n Dr. Sunita Bhati

JAYOTI VIDYAPEETH WOMEN'S UNIVERSITY, JAIPUR

UGC Approved Under 2(f) & 12(b) | NAAC Accredited | Recognized by Statutory Councils

Printed by :
JAYOTI PUBLICATION DESK

Published by :
Women University Press
Jayoti Vidyapeeth Women's University, Jaipur

Faculty of Education & Methodology

Title: .NET PROGRAMMING -LAB MANUAL
Author Name: Dr. Sunita Bhati
Published By: Women University Press

Publisher's Address: Jayoti Vidyapeeth Women's University, Jaipur
Vedant Gyan Valley,
Village-Jharna, Mahala Jobner Link Road, NH-8
Jaipur Ajmer Express Way,
Jaipur-303122, Rajasthan (India)

Printer's Detail: Jayoti Publication Desk

Edition Detail:

ISBN: 978-93-94024-89-2

Table of Contents

1. **Make a VB.NET program to print Hello World on the Console screen.**

```
Imports System
Module Module1
  Sub Main()
    Console.WriteLine("Hello World")
    Console.ReadKey()
  End Sub
End Module
```

Output:

Hello World

2. **Make a VB.NET program to show the use of variables and operators.**

```
Module variablesNdataypes
  Sub Main()
    Dim a As Short
    Dim b As Integer
    Dim c As Double

    a = 10
    b = 20
    c = a + b
    Console.WriteLine("a = {0}, b = {1}, c = {2}", a, b, c)
    Console.ReadLine()
  End Sub
End Module
```

Output:

```
a = 10, b = 20, c = 30
```

3. **Make a VB.NET program to show type conversion using various functions.**

```
Module DataTypes
  Sub Main()
    Dim n As Integer
    Dim da As Date
    Dim bl As Boolean = True
    n = 1234567
    da = Today

    Console.WriteLine(bl)
    Console.WriteLine(CSByte(bl))
    Console.WriteLine(CStr(bl))
    Console.WriteLine(CStr(da))
    Console.WriteLine(CChar(CStr(da)))
    Console.ReadKey()
  End Sub
End Module
```

Output:

```
True
-1
True
12/4/2012
1
```

4. **Make a VB.NET program to show the use of variables and constant.**

```
Module constantsNenum
  Sub Main()
    Const PI = 3.14149
    Dim radius, area As Single
    radius = 7
    area = PI * radius * radius
    Console.WriteLine("Area = " & Str(area))
    Console.ReadKey()
  End Sub
End Module
```

Output:

```
Area = 153.933
```

5. **Make a VB.NET program to show the use of variables and Enum data type.**

```
Module constantsNenum
  Enum Colors
    red = 1
    orange = 2
    yellow = 3
    green = 4
    azure = 5
    blue = 6
    violet = 7
  End Enum

  Sub Main()
    Console.WriteLine("The Color Red is : " & Colors.red)
    Console.WriteLine("The Color Yellow is : " & Colors.yellow)
    Console.WriteLine("The Color Blue is : " & Colors.blue)
    Console.WriteLine("The Color Green is : " & Colors.green)
    Console.ReadKey()
  End Sub
End Module
```

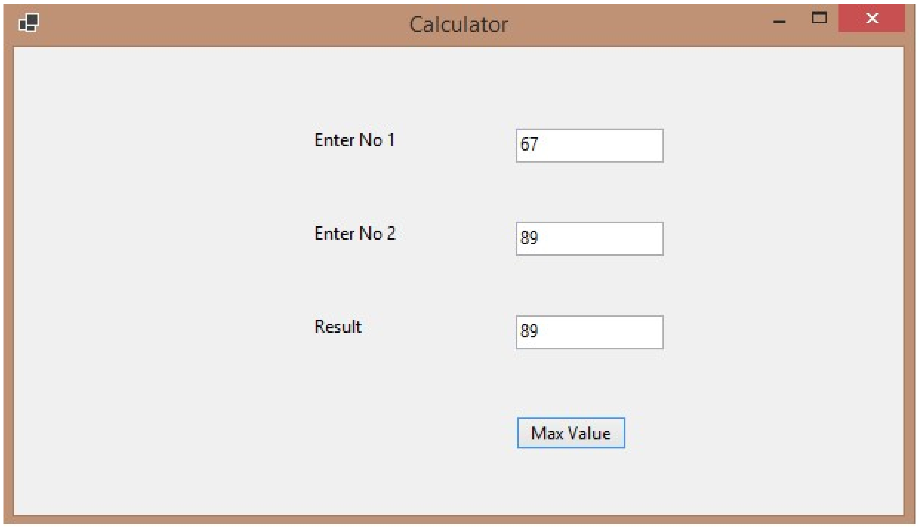
Output:

```
The Color Red is: 1
The Color Yellow is: 3
The Color Blue is: 6
The Color Green is: 4
```

6. **Make a VB.NET Window based program to calculate greater number between two numbers.**

```
Public Class Form1
   Private Sub btnMax_Click(sender As Object, e As EventArgs) Handles
btnMax.Click
       Dim a As Integer = Integer.Parse(txtNo1.Text)
       Dim b As Integer = Integer.Parse(txtNo2.Text)
       If a > b Then
           txtRes.Text = CStr(a)
       End If
       If b > a Then
           txtRes.Text = CStr(b)
       End If
   End Sub
End Class
```

Output:



Enter No 1 67

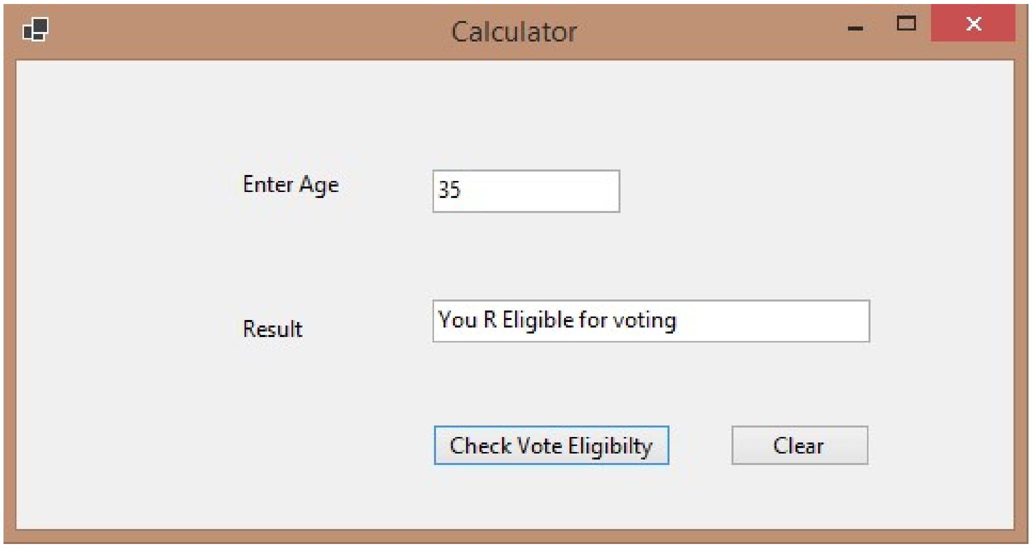
Enter No 2 89

Result 89

7. Make a VB.NET Window based program to check eligibility for voting.

```
Public Class Form1

    Private Sub btnChkId_Click(sender As Object, e As EventArgs) Handles
btnChkId.Click
        Dim age As Integer = Integer.Parse(txtNo1.Text)

        If age >= 18 Then
            txtRes.Text = "You R Eligible for voting"
        Else
            txtRes.Text = "You R not Eligible for voting"
        End If
    End Sub

    Private Sub btnClr_Click(sender As Object, e As EventArgs) Handles
btnClr.Click
        txtNo1.Text = ""
        txtRes.Text = ""
    End Sub

End Class
```

Output:



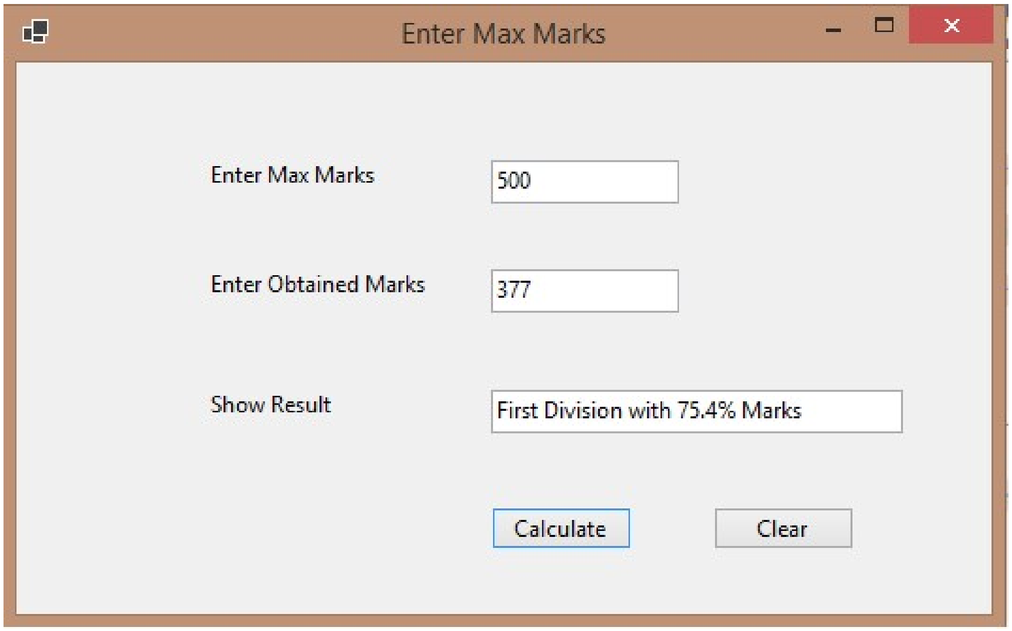
Enter Age 35

Result You R Eligible for voting

8. Make a VB.NET Window based program to calculate division and percentage.

```
Public Class Form1

    Private Sub btnCalculate_Click(sender As Object, e As EventArgs) Handles
btnCalculate.Click
        Dim maxmarks As Integer = Integer.Parse(txtMaxMarks.Text)
        Dim obtmarks As Integer = Integer.Parse(txtObtMarks.Text)
        Dim percentage As Single = (obtmarks * 100) / maxmarks
        If percentage >= 60 And percentage <= 100 Then
            txtResult.Text = "First Division with " + CStr(percentage) + "% Marks"
        ElseIf percentage >= 48 And percentage < 60 Then
            txtResult.Text = "Second Division with " + CStr(percentage) + "% Marks"
        ElseIf percentage >= 36 And percentage < 48 Then
            txtResult.Text = "Third Division with " + CStr(percentage) + "% Marks"
        Else
            txtResult.Text = "Fail with " + CStr(percentage) + "% Marks"
        End If
    End Sub
    Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles
btnClear.Click
        txtMaxMarks.Text = ""
        txtObtMarks.Text = ""
        txtResult.Text = ""
    End Sub
End Class
```

Output:



Enter Max Marks

Enter Max Marks 500

Enter Obtained Marks 377

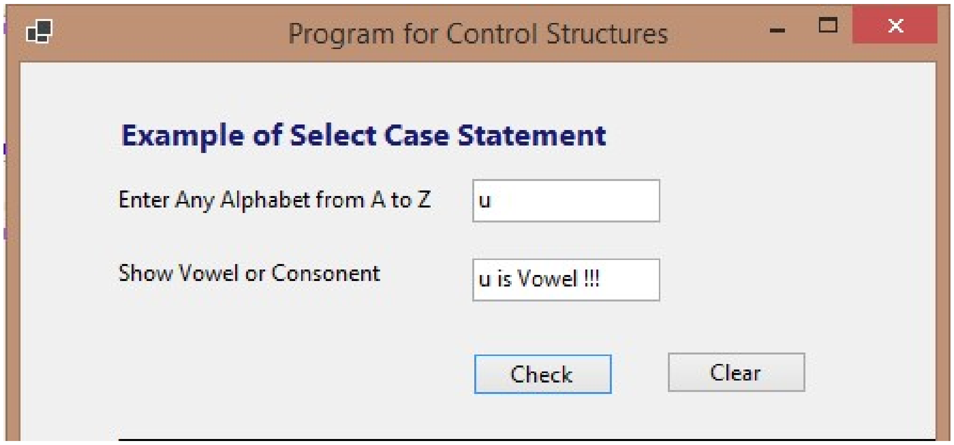
Show Result First Division with 75.4% Marks

Select Case Statement

9. **Make a VB.NET Window based program to check an entered character is Vowel or Consonant.**

```
Public Class Form1

    Private Sub btnChk_Click(sender As Object, e As EventArgs) Handles btnChk.Click
        Dim x As Char = CChar(txtChar.Text)
        Select Case x
            Case "a"
                txtShow.Text = x + " is Vowel !!!"
            Case "e"
                txtShow.Text = x + " is Vowel !!!"
            Case "i"
                txtShow.Text = x + " is Vowel !!!"
            Case "o"
                txtShow.Text = x + " is Vowel !!!"
            Case "u"
                txtShow.Text = x + " is Vowel !!!"
            Case Else
                txtShow.Text = x + " is Consonent !!!"
        End Select
    End Sub
    Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
        txtChar.Text = ""
        txtShow.Text = ""
    End Sub


End Class
```

Output:



Program for Control Structures

Example of Select Case Statement

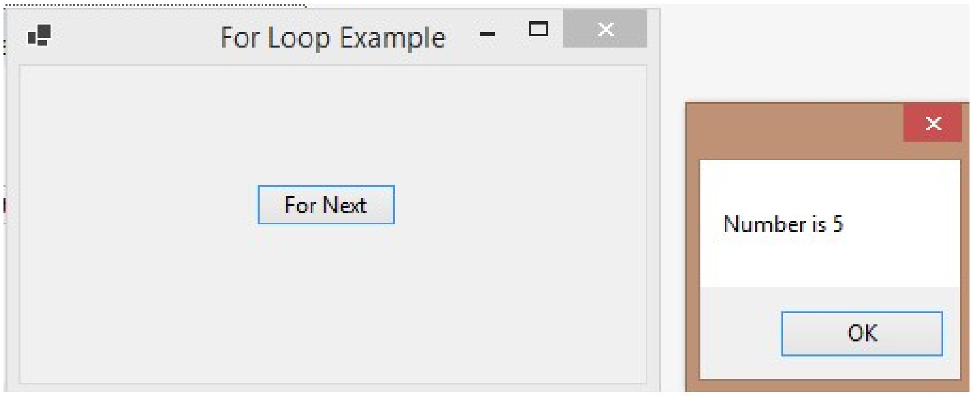
Enter Any Alphabet from A to Z: u

Show Vowel or Consonent: u is Vowel !!!

10. Print number 1 to 5 in message box using For loop.

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        For i As Integer = 1 To 5
            MessageBox.Show("Number is " & i)
        Next
    End Sub
End Class
```

Output:

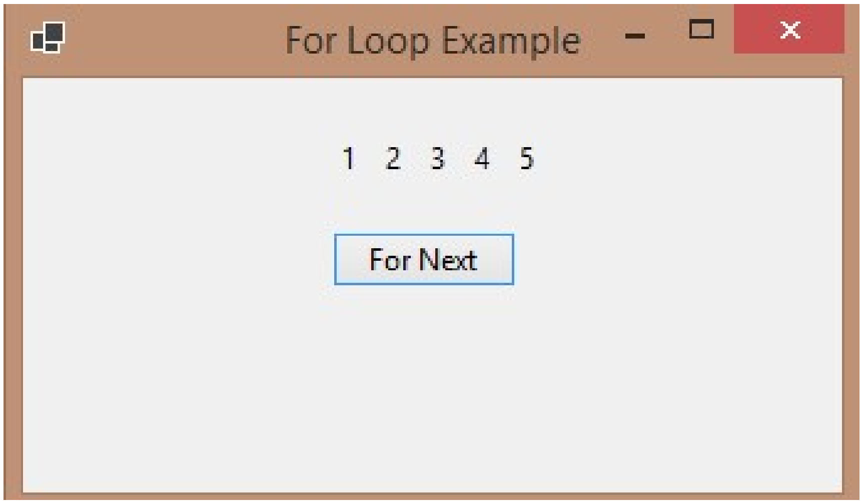


Number is 5

11. Print numbers from 1 to 5 in on a Label.

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        lblShow.Text = ""
        For i As Integer = 1 To 5
            lblShow.Text = lblShow.Text + "    " + CStr(i)
        Next
    End Sub
End Class
```

Output:



```
1  2  3  4  5
```

Nested For Next Loop

12. **Print the following pyramid in a richtextbox.**

1

1 2

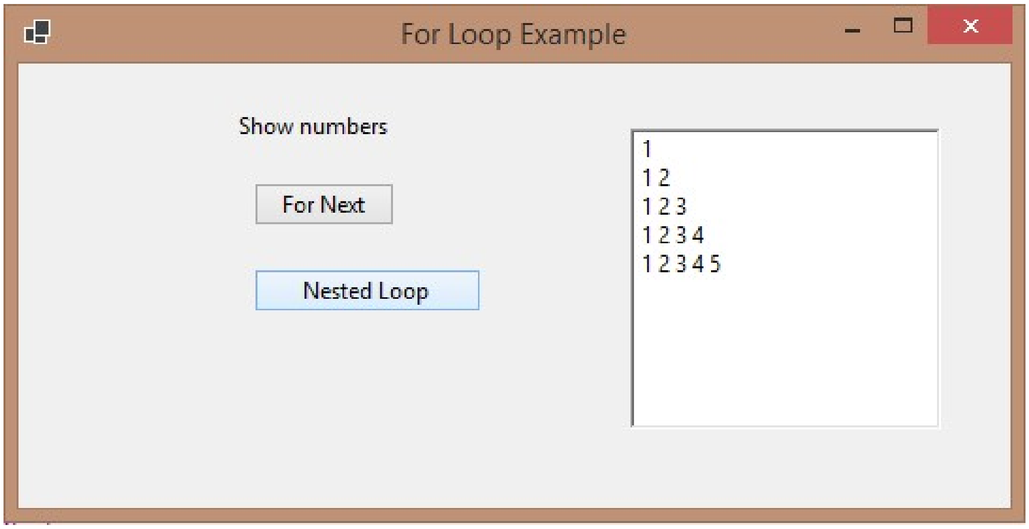
1 2 3

1 2 3 4

1 2 3 4 5

```
Public Class Form1

    Private Sub btnNestedLoop_Click(sender As Object, e As EventArgs) Handles
btnNestedLoop.Click
        For i As Integer = 1 To 5

            For k As Integer = 1 To i

                rtbShow.Text = rtbShow.Text + " " + CStr(k)
            Next
            rtbShow.Text = rtbShow.Text + vbCrLf

        Next
    End Sub
End Class
```

Output:



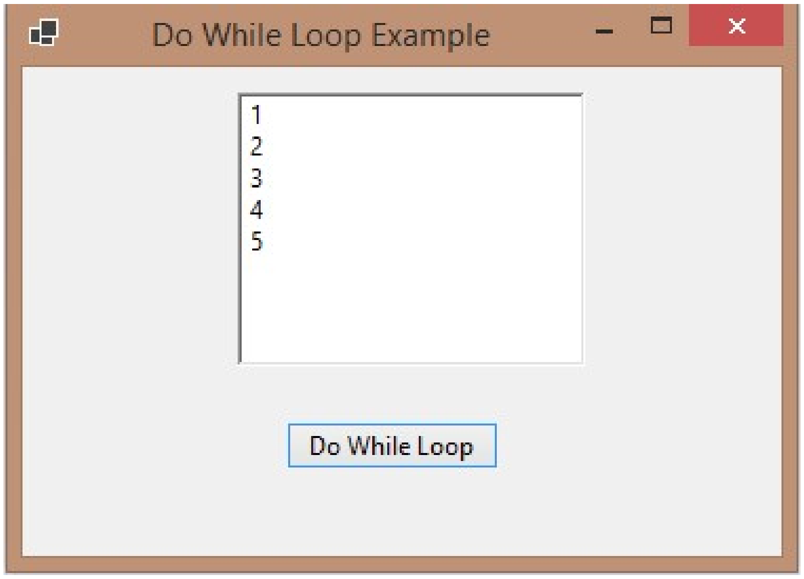
```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

Do While Loop

13. Print 1 to 5 numbers using Do While loop.

```
Public Class Form1
    Private Sub btnDowhileloop_Click(sender As Object, e As EventArgs) Handles
btnDowhileloop.Click
        Dim i As Integer = 0
        Do
            i += 1
            rtbShow.Text = rtbShow.Text + " " + CStr(i) + vbCrLf
        Loop While (i < 5)
    End Sub
End Class
```

Output:

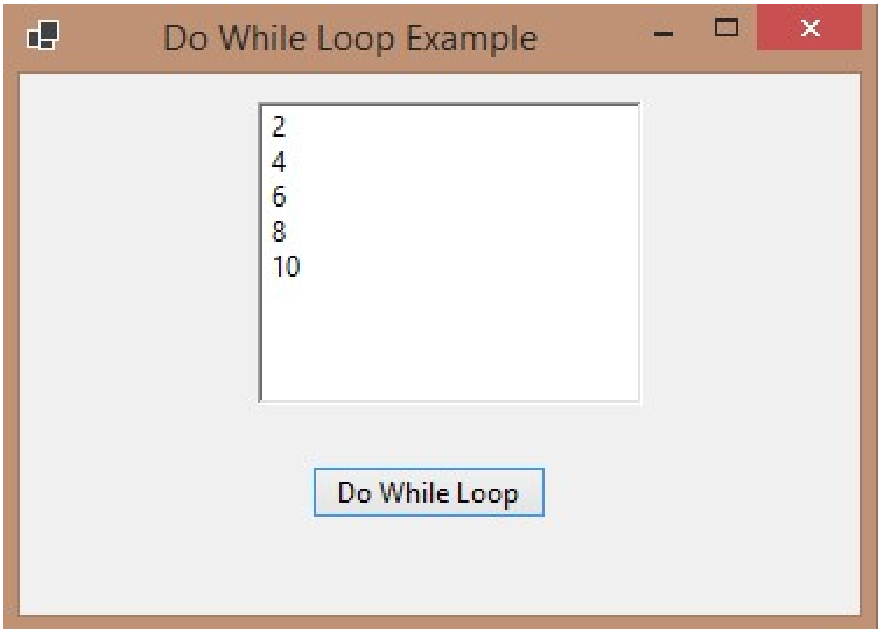

Do While Loop Example

```
1
2
3
4
5
```

14. Print even numbers between 1 to 10 (including 10) using Do While loop.

```
Public Class Form1
    Private Sub btnDowhileloop_Click(sender As Object, e As EventArgs) Handles
btnDowhileloop.Click
        Dim i As Integer = 0

        Do
            i += 2
            rtbShow.Text = rtbShow.Text + " " + CStr(i) + vbCrLf
        Loop While (i < 10)

    End Sub
End Class
```

Output:



```
2
4
6
8
10
```

Nested Do While Loop

15. **Print the following pyramid in a richtextbox.**

1

2 2

3 3 3

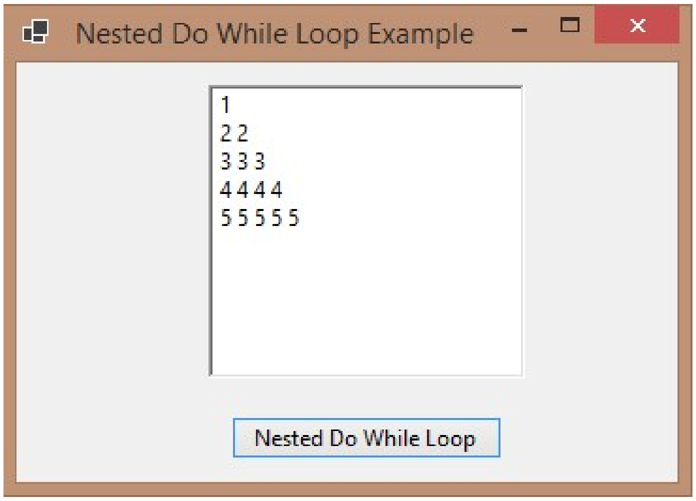
4 4 4 4

5 5 5 5 5

```
Public Class Form1
    Private Sub btnNestedDoWhile_Click(sender As Object, e As EventArgs) Handles
btnNestedDoWhile.Click
    Dim i As Integer = 0
    Dim j As Integer = 0

    Do
      i += 1
      Do
```

```
                j += 1
                rtbShow.Text = rtbShow.Text + " " + CStr(i)
            Loop While (j < i)
            j = 0
            rtbShow.Text = rtbShow.Text + vbCrLf
        Loop While (i < 5)

    End Sub
End Class
```

Output:



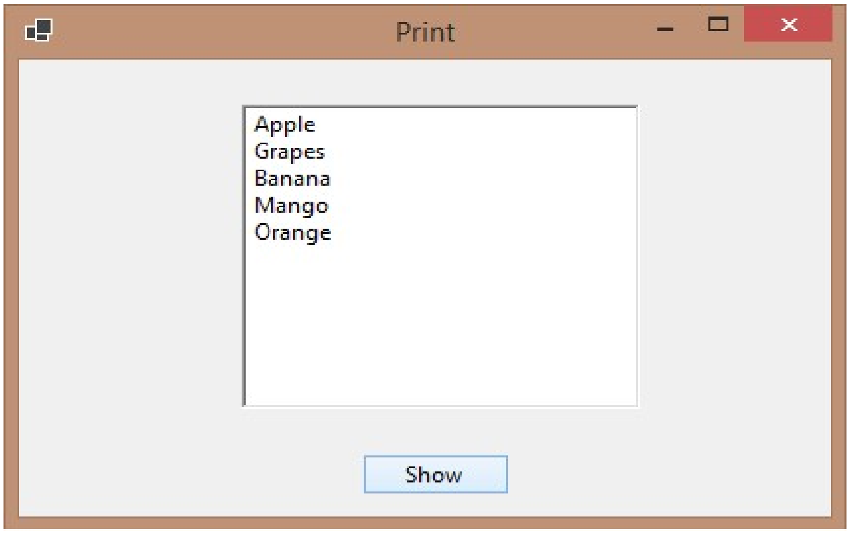
```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

For Each Loop

16. **Program to show the values of a string array into a richtextbox using For Each Loop.**

```
Public Class Form1
    Private Sub btnShow_Click(sender As Object, e As EventArgs) Handles btnShow.Click
        Dim str() As String = {"Apple", "Grapes", "Banana", "Mango", "Orange"}

        Dim fruit As String
        For Each fruit In str
            rtbShow.Text = rtbShow.Text + " " + fruit + vbCrLf

        Next

    End Sub
End Class
```

Output:

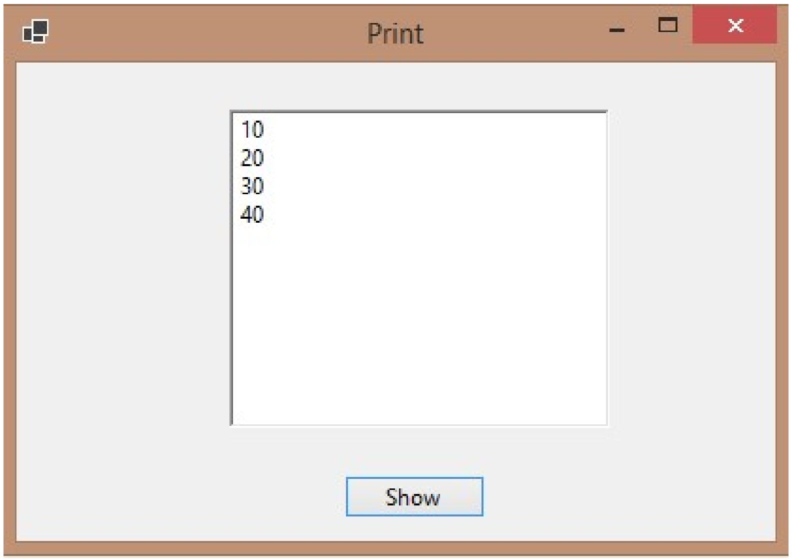


```
Apple
Grapes
Banana
Mango
Orange
```

17. Program to show the values of an integer array into richtextbox using For Each Loop.

```
Public Class Form1
    Private Sub btnShow_Click(sender As Object, e As EventArgs) Handles btnShow.Click
        Dim num() As Integer = {10, 20, 30, 40}

        Dim i As Integer
        For Each i In num
            rtbShow.Text = rtbShow.Text + " " + CStr(i) + vbCrLf

        Next

    End Sub

End Class
```

Output:



```
10
20
30
40
```

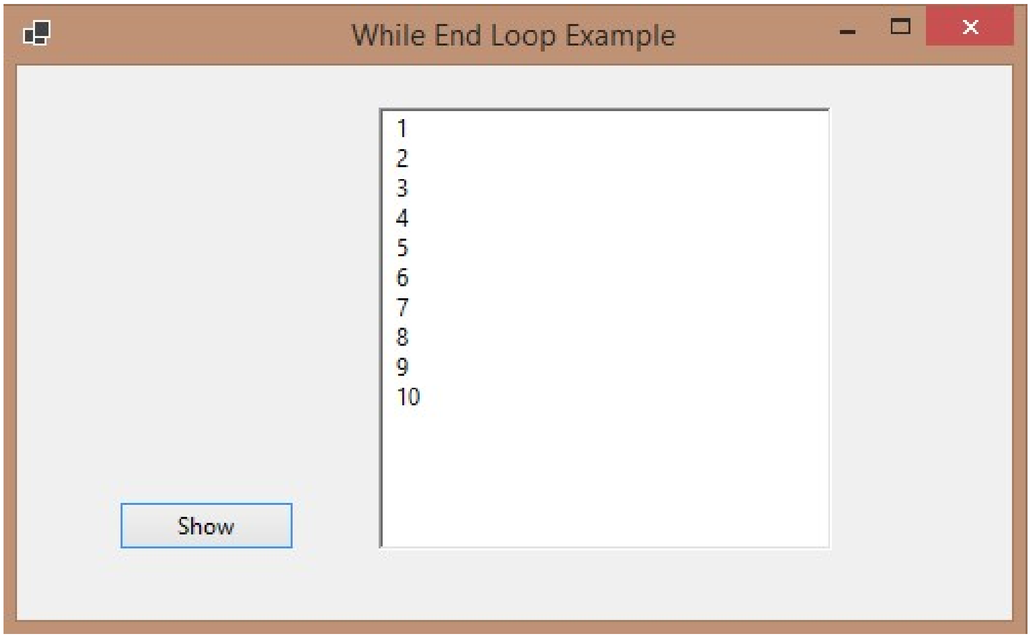
While End While Loop

18. **Write a simple program to print the number from 1 to 10 using While End loop in VB.NET.**

```
Public Class Form1

Private Sub btnOutput_Click(sender As Object, e As EventArgs) Handles btnOutput.Click
        Dim i As Integer = 0
        While i < 10
```

```
            i += 1
            rtbShowOutput.Text = rtbShowOutput.Text + "  " + CStr(i) + vbCrLf

        End While
End Sub

End Class
```

Output:



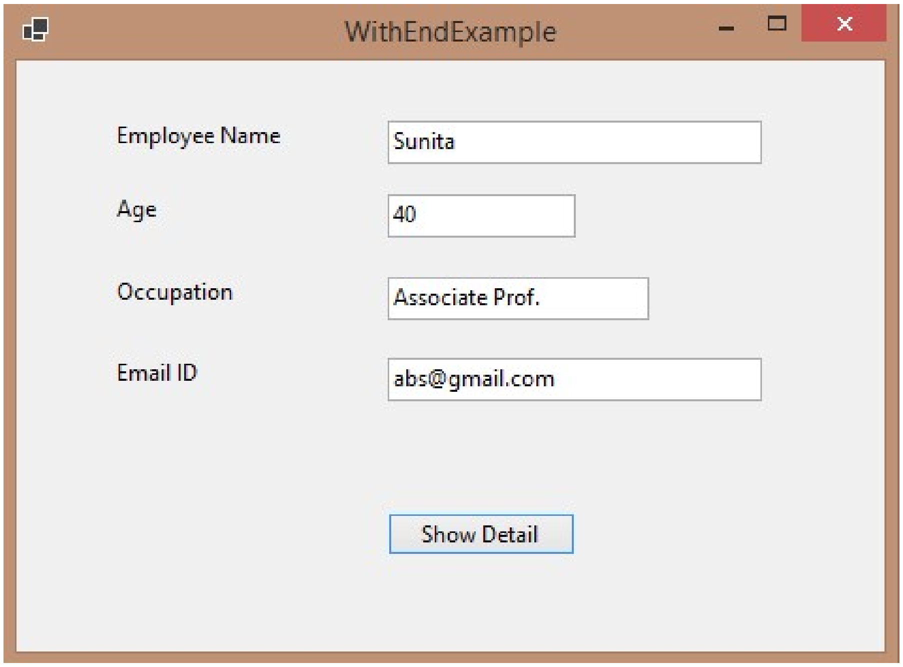
```
1
2
3
4
5
6
7
8
9
10
```

19. Write a program to understand the uses of With End statement in VB.NET.

```
Public Class With End Example
    Public Property name As String
    Public Property age As Integer
    Public Property occupation As String
    Public Property e-mail ID As String

    Private Sub btn Show Emp Data_Click(sender As Object, e As Event Args) Handles btn
Show Emp Data. Click
```

```
        Dim emp As New With End Example
        With emp
            .name = "Sunita"
            .age = 40
            .occupation = "Associate Prof."
            .e-mail ID = "abs@gmail.com"
        End With
        With emp
            Txt Name. Text = .name
            Txt Age. Text = CS tr(.age)
            Txt Occupation. Text = .occupation
            Txt Email. Text = .e-mail id

        End With

    End Sub
End Class
```

Output:



Employee Name Sunita

Age 40

Occupation Associate Prof.

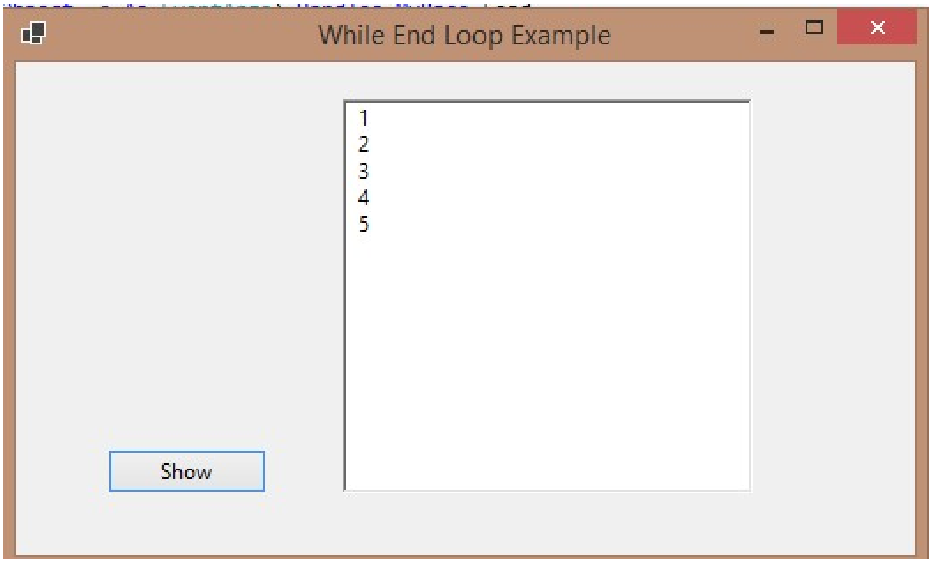
Email ID abs@gmail.com

Exit Statement

20. Write a simple program to use the Exit Statement in While End loop.

```
Public Class Form1

    Private Sub btn Output_Click(sender As Object, e As Event Args) Handles btn Output.
Click
        Dim i As Integer = 0
        While i < 10
            If i = 5 Then
                Exit While
            End If

            i += 1
            rtb Show Output. Text = rtb Show Output. Text + "  " + CS tr(i) + vb Cr Lf
        End While

    End Sub
End Class
```

Output:



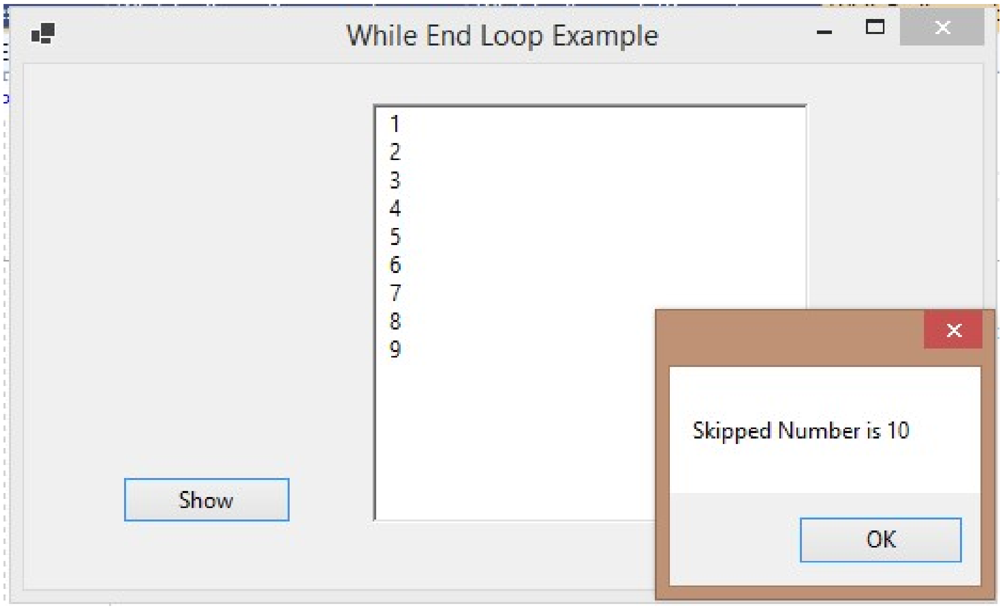
```
1
2
3
4
5
```

Continue Statement:

21.Write a VB.Net program to show the use of Continue Statement in While End loop.

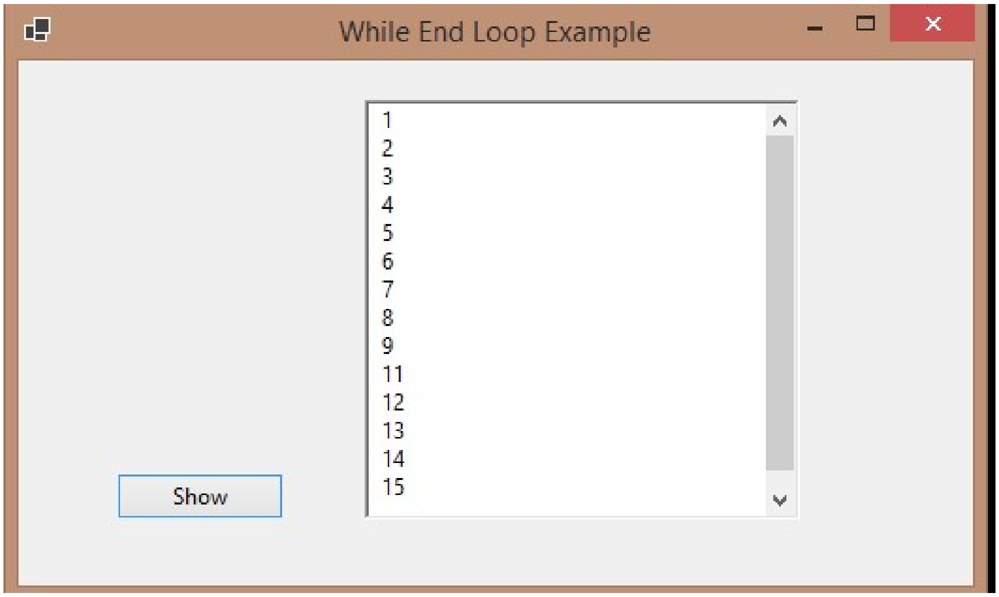
```
Public Class Form1

   Private Sub btn Output_ Click (sender As Object, e As Event Args) Handles btn Output.
Click
        Dim i As Integer = 0
        While i < 15
           If i = 9 Then

               Message Box. Show("Skipped Number is " + CS tr(i + 1))
               i += 1
               Continue While
           End If

           i += 1
           rtb Show Output. Text = rtb Show Output. Text + "  " + CS tr(i) + vb Cr Lf
        End While
   End Sub
End Class
```

Output:



```
1
2
3
4
5
6
7
8
9
```

Skipped Number is 10



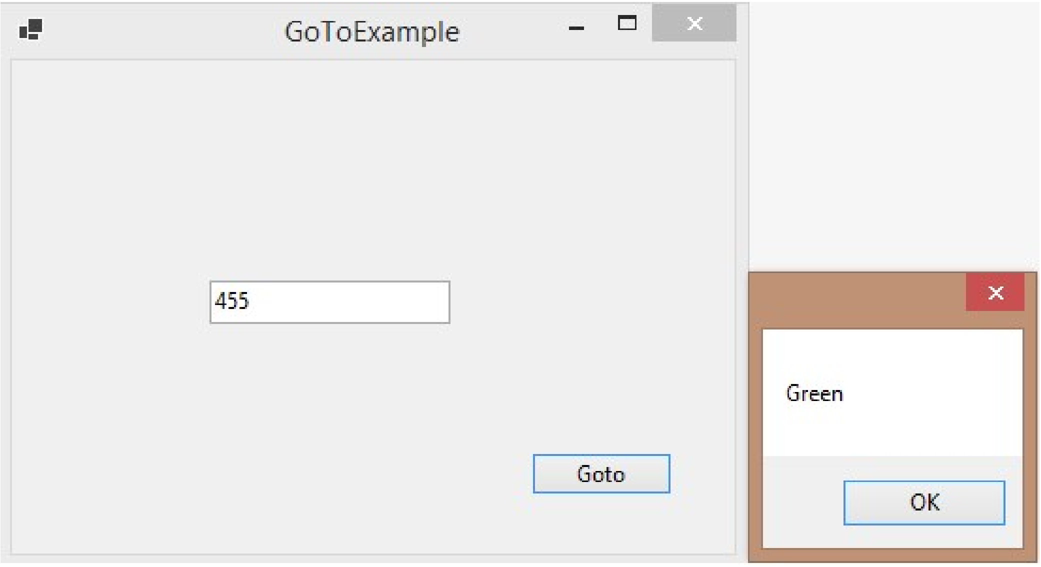
```
1
2
3
4
5
6
7
8
9
11
12
13
14
15
```

Go To Statement

22. Write a VB.Net program to show the use of GoTo Statement.

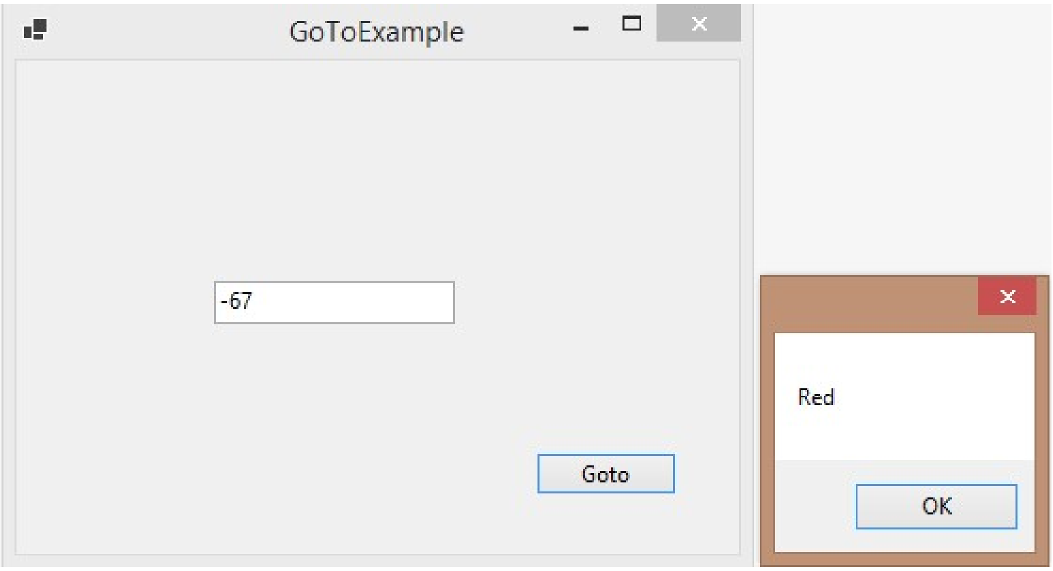
```
Public Class Go To Example
  Private Sub btn Goto_ Click(sender As Object, e As Event Args) Handles btn Goto. Click
    Dim x As Integer = Integer. Parse(txt No. Text)
    If x > 0 Then
      Go To Green
    Else
      Go To Red
    End If

Red:   Message Box. Show("Red")
    Exit Sub
Green:  Message Box. Show("Green")

   End Sub
End Class
```

Output:



455

Green



-67

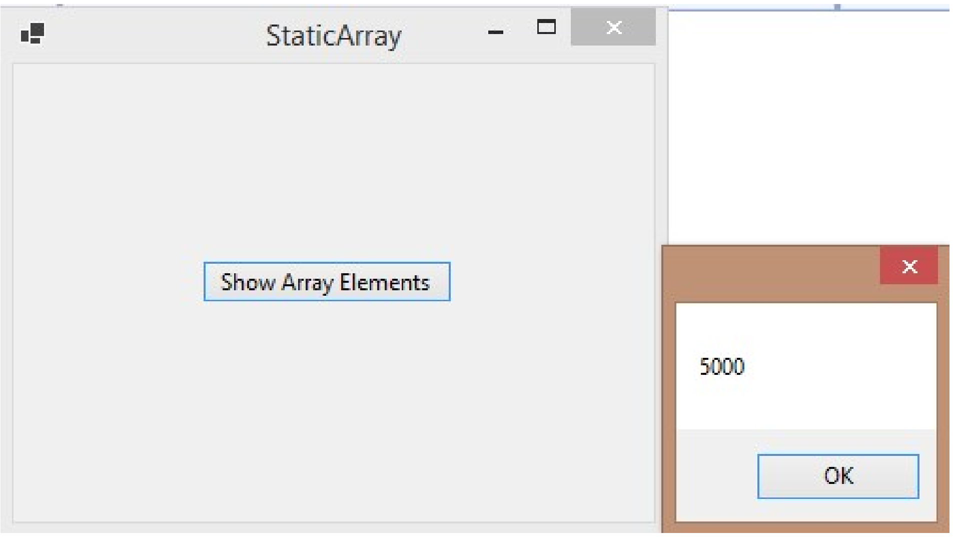
Red

Example of Static Array:

23. Write a VB.Net program to show the use of Static Array.

```
Public Class Static Array

  Private Sub Btn Show Array_ Click(sender As Object, e As Event Args) Handles Btn
Show Array. Click
      Dim num(5) As Integer
      Dim i As Integer
      Dim str As String
      For i = 0 To 4
          num(i) = (i + 1) * 1000
      Next
      For i = 0 To 4
          str = CS tr(num(i))
          Message Box. Show(str)
      Next

  End Sub
End Class
```

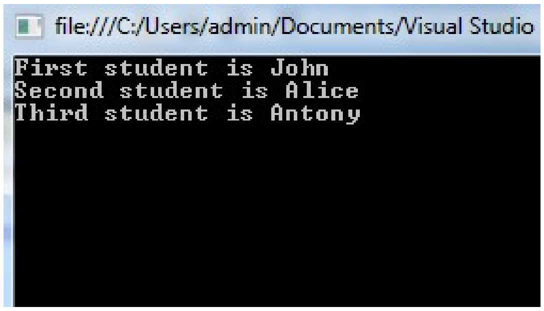
Output:



5000

24. Console Based Program of a Static Array.

```
Module Module1
    Sub Main()

       Dim students(0 to 2) As String

        students(0) = "John"

        students(1) = "Alice"

        students(2) = "Antony"

        Console. Write Line("First student is {0} ", students(0))
        Console. Write Line("Second student is {0} ", students(1))

        Console. Write Line("Third student is {0} ", students(2))

        Console. Read Key ()

    End Sub

End Module
```



```
First student is John
Second student is Alice
Third student is Antony
```

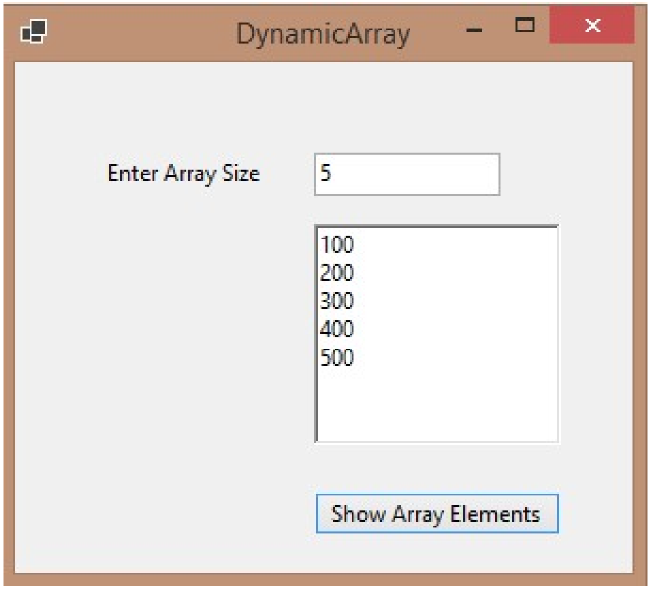
Example of Dynamic Array:

25. Write a VB.Net program to show the use of Dynamic array.

```
Public Class Dynamic Array
    Private Sub Btn Show Array_ Click(sender As Object, e As Event Args) Handles Btn
Show Array. Click
        Dim n As Integer = Integer. Parse(Txt Size. Text)
        Dim ary() As Integer
        Dim i As Integer

        Re Dim ary(n)
        For i = 0 To n - 1
            ary(i) = (i + 1) * 100
        Next
        For i = 0 To n - 1
```

```
            Rtb Show Array. Text = Rtb Show Array. Text + CStr (ary(i)) + vb CrLf
        Next
    End Sub

End Class
```

Output:



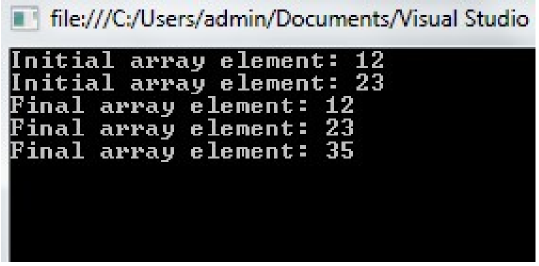
Enter Array Size 5

```
100
200
300
400
500
```

26. Console Based Program : Adding New Elements to an Array

```
Module Module1
    Sub Main()
        Dim nums() As Integer
        Re Dim nums(1)
        Nums(0) = 12
        nums(1) = 23
        For x = 0 To nu ms. Length - 1
            Console. Write Line("Initial array element: {0}", nu ms(x))
        Next
        Re Dim Preserve nums(2)
        nums(2) = 35
        For x = 0 To nums. Length - 1
            Console. Write Line("Final array element: {0}", nums(x))
        Next
        Console. Read Key()
    End Sub
End Module
```

Output:



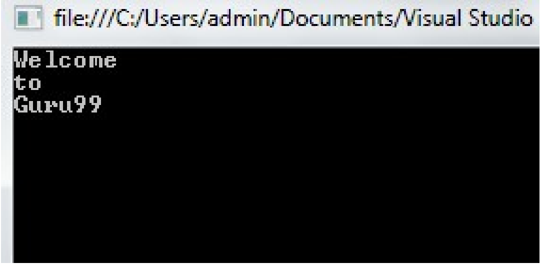
```
Initial array element: 12
Initial array element: 23
Final array element: 12
Final array element: 23
Final array element: 35
```

Array Functions:

27. Write a VB.Net program to show the use of Split Function.

```
Module Module1
    Sub Main()
        Dim my array() As String
        Dim str As String
        Dim x As Integer
        str = "Welcome, to, guru99"
        myarray = Split(str, ", ")
        For x = L Bound (myarray) To U Bound(myarray)
            Console. Write Line(myarray(x))
        Next
        Console.ReadKey()
    End Sub
End Module
```

Output:

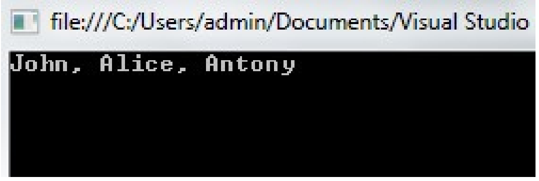


```
Welcome
to
Guru99
```

28. Write a VB.Net program to show the use of Join Function.

```
Module Module1
    Sub Main()
        Dim students(0 To 2) As String
        students(0) = "John"
        students(1) = "Alice"
        students(2) = "Antony"
        Dim classmates As String
        classmates = Join(students, ", ")
        Console. Write Line(classmates)
        Console. Read Key()
    End Sub
End Module
```

Output:



```
John, Alice, Antony
```

Procedures

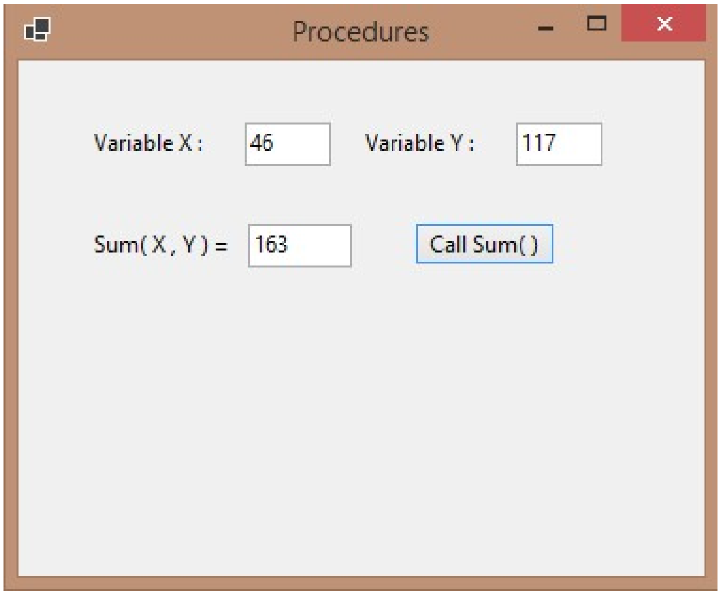
29. Console based program to show the use of procedure.

```
Module Example

    Sub Main()

        Display()



    End Sub

    Sub Display()
        Console. Write Line("Simple procedure")
    End Sub


End Module
```

30. Write a VB.Net window based program to show the use of procedure.

```
Public Class Procedures
    Private Sub Btn Sum_ Click(sender As Object, e As Event Args) Handles Btn Sum.Click
        Dim x As Integer = Integer. Parse(Txt X. Text)
        Dim y As Integer = Integer. Parse(Txt Y. Text)
        Sum(x, y)

    End Sub
    Sub Sum(By Val X As Integer, By Val Y As Integer)
        Txt Sum. Text = CS tr(X + Y)

    End Sub

End Class
```

Output:



Variable X : 46   Variable Y : 117

Sum(X , Y) = 163

Functions

31. Write a VB.Net console based program to show the use of built-in functions.

```
Module Example
   Sub Main()
      Console. Write Line(Math. Abs(-23))
      Console. Write Line(Math. Round(34.56))
      Console. Write Line ("Zet Code has {0} characters", _
         Len("Zet Code"))
   End Sub
End Module
```

32. Write a VB.Net console based program to create Addition function to find the sum of two numbers.

```
Module Example
   Dim x As Integer = 55
   Dim y As Integer = 32
   Dim result As Integer
   Sub Main()
      result = Addition(x, y)
      Console. Write Line(result)
    End Sub
    Function Addition(By Val k As Integer, _
                 By Val l As Integer) As Integer
```

```
        Return k+l

    End Function

 End Module
```

33. Write a VB.Net console based program to create Swap function to swap the values of two numbers.

```
Module Example

    Dim a As Byte = 4

    Dim b As Byte = 7

    Sub Main()

        Console. Write Line("Outside Swap procedure")

        Console. Write Line("a is {0}", a)

        Console. Write Line("b is {0}", b)

        Swap(a, b)

        Console. Write Line("Outside Swap procedure")

        Console. Write Line("a is {0}", a)

        Console. Write Line("b is {0}", b)

    End Sub

    Sub Swap(By Val a As Byte, By Val b As Byte)

        Dim temp As Byte

        temp = a

        a = b
        b = temp

        Console. Write Line("Inside Swap procedure")

        Console. Write Line("a is {0}", a)

        Console. Write Line("b is {0}", b)

    End Sub

End Module
```

34. Write a VB.Net console based program to create Swap function to swap the values of two numbers using call by reference method.

```
Module Example
    Dim a As Byte = 4
    Dim b As Byte = 7
    Sub Main()
        Console. Write Line("Outside Swap procedure")
        Console. Write Line("a is {0}", a)
        Console. Write Line("b is {0}", b)
        Swap(a, b)
        Console. Write Line("Outside Swap procedure")
        Console. Write Line("a is {0}", a)
        Console. Write Line("b is {0}", b)
    End Sub
    Sub Swap(By Ref a As Byte, By Ref b As Byte)
        Dim temp As Byte
        temp = a
        a = b
        b = temp
        Console. Write Line("Inside Swap procedure")
        Console. Write Line("a is {0}", a)
        Console. Write Line("b is {0}", b)
    End Sub
End Module
```

Recursion

35. Write a VB.Net console based program to find the factorial value of an entered number using Recursion function.

```
Module Example

    Sub Main()

        Console. Write Line(Factorial(4))

        Console. Write Line(Factorial(10))

    End Sub

    Function Factorial(By Val n As U Short) As U Short

        If (n=0)

            Return 1

        Else

            Return n * Factorial(n-1)

        End If

    End Function

 End Module
```

Overloading:

36. Write a VB.Net console based program to demonstrate the use of overloading.

```
Imports System

Class Adder

Overloads Public Sub Add(A as Integer, B as Integer)

    Console. Write line ("Adding Integers: " + Convert. To String(A + B))

End Sub
```

```
Overloads Public Sub Add(A as String, B as String)

        Console. Write line ("Adding Strings: " + a + b)

End Sub

Shared Sub Main()

        Dim Adder Obj as Adder

        Adder Obj=new Adder

        Adder Obj.Add(10,20)

        Adder Obj.Add("hello"," how are you")

    End Sub

End Class
```

Overriding:

37. Write a VB.Net console based program to demonstrate the use of overriding.

```
Class Human

   Over rid able Public Sub Speak()

        Console. Write line ("Speaking")

    End Sub

End Class


Class Indian

    Inherits Human

    Overrides Public Sub Speak()
```

```
        Console. Write line ("Speaking Hindi")

    End Sub

End Class


Class Main Class

    Shared Sub Main()

        Dim Tom as Human

        Tom=new Human

        Dim Tony as Indian

        Tony=new Indian

        Tom. Speak()

        Tony. Speak()

    End Sub

End Class
```

Shared Members of a VB.Net Class:

38. Write a VB.Net console based program to demonstrate the use of Shared Members.

```
Class Static Var
   Public Shared num As Integer
   Public Sub count()
      num = num + 1
   End Sub
   Public Shared Function get Num() As Integer
      Return num
   End Function
```

```
  Shared Sub Main()
    Dim s As Static Var = New Static Var()
    s. count()
    s. count()
    s. count()
    Console. Write Line("Value of variable num: {0}", Static Var. get Num())
    Console. Read Key()
  End Sub
End Class
```

Inheritance

39. Write a VB.Net console based program to demonstrate the use of Inheritance.

```
' Base class

Class Shape

  Protected width As Integer

  Protected height As Integer

  Public Sub set Width(By Val w As Integer)

    width = w

  End Sub

  Public Sub set Height(By Val h As Integer)

    height = h

  End Sub

End Class

' Derived class

Class Rectangle : Inherits Shape
```

```
  Public Function get Area() As Integer
    Return (width * height)
  End Function
End Class
Class Rectangle Tester
  Shared Sub Main()
    Dim rect As Rectangle = New Rectangle()
    rect. set Width(5)
    rect. Set Height(7)
    ' Print the area of the object.
    Console. Write Line("Total area: {0}", rect. Get Area())
    Console. Read Key()
  End Sub
End Class
```

Abstract class: -

40. Write a VB.Net console based program to demonstrate the use of Abstract Class.

```
Public Class Form1
   Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArg
s) Handles Button1.Click
      Dim a As New raj1
      a. ravi("Rahul")
   End Sub
   Public Must Inherit Class raj
      Public Must Override Sub ravi(By Val s As String)
   End Class
   Public Class raj1
      Inherits raj
      Public Overrides Sub ravi(By Val s As String)
         Msg Box("Hello:" & s)
      End Sub
   End Class
End Class
```

Interface:

41. Write a VB.Net console based program to demonstrate the use of Interface.

```
Public Class Form2
   Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArg
s) Handles Button1.Click
      Dim a As New Student
      a. sagar(25)
      a.amit("Abhi jeet")
      Msg Box(a.rahul(10, 20))
   End Sub
   Public Interface Person
```

```
        Sub sagar (By Val x As Integer)
        Sub amit (By Val s As String)
        Function rahul(By Val x As Integer, By Val y As Integer)
    End Interface
    Public Class Student
        Implements Person
        Public Sub amit(ByVal s As String) Implements Person.amit
            Msg Box("Hello:" & s)
        End Sub
        Public Function rahul(ByVal x As Integer, ByVal y As Integer) As Object Implements Person.rahul
            Dim z As Integer
            z = x + y
            Return z
        End Function
        Public Sub sagar(ByVal x As Integer) Implements Person.sagar
            Msg Box(x)
        End Sub
    End Class
End Class
```

Contact Us:
University Campus Address:

Jayoti Vidyapeeth Women's University
Vadaant Gyan Valley, Village-Jharna, Mahala Jobner Link Road,
Jaipur Ajmer Express Way, NH-8, Jaipur- 303122, Rajasthan (INDIA)
(Only Speed Post is Received at University Campus Address, No. any Courier Facility is available at Campus Address)

Pages	: 43
Book Price	: ₹ 150/-

JPD
Women University Press
Jayoti Publication Desk

Year & Month of Publication- 11/05/2022