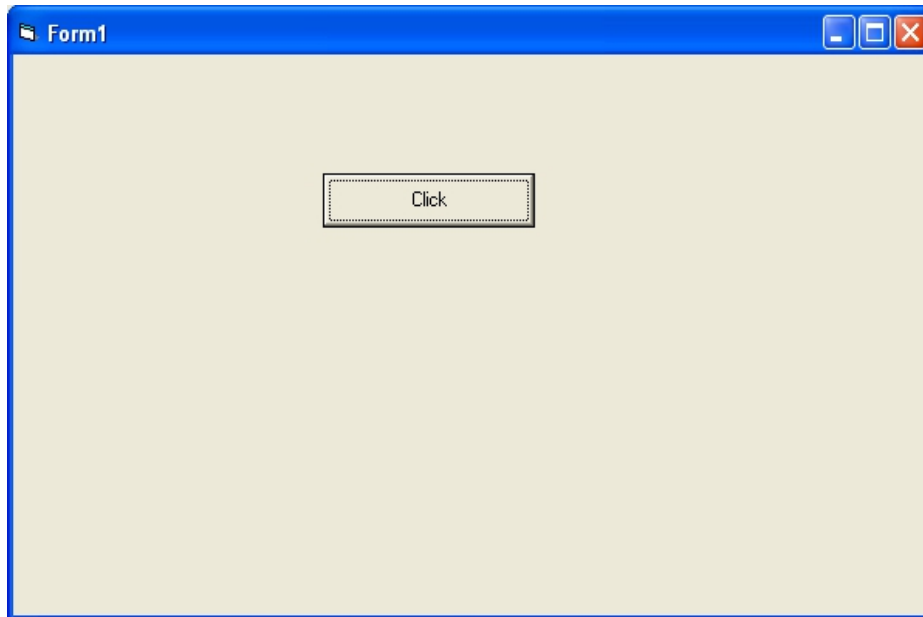


Program to print Hello World in a Message Box on clicking on a button

```
Private Sub Command1_Click()  
MsgBox ("Hello World")  
End Sub
```



Program to add,subtract,multiply and divide two numbers

```
Private Sub Command1_Click()  
Dim a As Integer  
Dim b As Integer  
Dim sum As Integer  
a = CInt(Text1.Text)  
b = CInt(Text2.Text)  
sum = a + b  
Label3.Caption = "Sum of Two Numbers is " & sum  
End Sub
```

```
Private Sub Command2_Click()  
Dim a As Integer  
Dim b As Integer  
Dim diff As Integer  
a = CInt(Text1.Text)  
b = CInt(Text2.Text)  
diff = a - b  
Label3.Caption = "Difference of Two Numbers is " & diff  
End Sub
```

```
Private Sub Command3_Click()  
Dim a As Integer  
Dim b As Integer
```

```
Dim diff As Integer
a = CInt(Text1.Text)
b = CInt(Text2.Text)
product = a * b
Label3.Caption = "Product of Two Numbers is " & product
End Sub
```

```
Private Sub Command4_Click()
Dim a As Integer
Dim b As Integer
Dim quotient As Integer
a = CInt(Text1.Text)
b = CInt(Text2.Text)
quotient = a / b
Label3.Caption = "Quotient of Two Numbers is " & quotient
End Sub
```

```
Private Sub Command5_Click()
Dim a As Integer
Dim b As Integer
Dim remainder As Integer
a = CInt(Text1.Text)
b = CInt(Text2.Text)
remainder = a Mod b
Label3.Caption = "Remainder of Two Numbers is " & remainder
End Sub
```

The screenshot shows a Windows application window titled "Form1". The window has a blue title bar with standard minimize, maximize, and close buttons. The main area is a light beige color. At the top, there are two text boxes: "Enter first Number" containing the value "20" and "Enter second number" containing the value "10". Below these are five buttons: "Add", "Subtract", "Multiply", "Quotient", and "Remainder". The "Add" button is highlighted with a dashed border. At the bottom of the window, a label displays the text "Sum of Two Numbers is 30".

Program to find area and perimeter of square

```
Private Sub Command1_Click()  
Dim side As Integer  
Dim area As Integer  
Dim perimeter As Integer  
side = CInt(Text1.Text)  
area = side * side  
Label2.Caption = "Area of square is " & area  
End Sub
```

```
Private Sub Command2_Click()  
Dim side As Integer  
Dim area As Integer  
Dim perimeter As Integer  
side = CInt(Text1.Text)  
perimeter = 4 * side  
Label2.Caption = "Perimeter of square is " & perimeter  
End Sub
```

The screenshot shows a Windows application window titled "Form2". Inside the window, there is a text box labeled "Enter side of square" with the value "10" entered. Below the text box are two buttons: "Calculate Area" and "Calculate Perimeter". At the bottom of the window, a label displays the result: "Area of square is 100".

Form2

Enter side of square

10

Calculate Area

Calculate Perimeter

Perimeter of square is 40

Program to find area and perimeter of rectangle
length and breadth are entered by user in textboxes

```
Private Sub Command1_Click()  
Dim length As Integer  
Dim breadth As Integer  
Dim area As Integer  
length = CInt(Text1.Text)  
breadth = CInt(Text2.Text)  
area = length * breadth  
Label3.Caption = "Area of Rectangle is " & area  
End Sub
```

```
Private Sub Command2_Click()  
Dim length As Integer  
Dim breadth As Integer  
Dim perimeter As Integer  
length = CInt(Text1.Text)  
breadth = CInt(Text2.Text)  
perimeter = 2 * (length + breadth)  
Label3.Caption = "Perimeter of Rectangle is " & perimeter  
End Sub
```

Form2

Enter side of square

10

Calculate Area

Calculate Perimeter

Area of square is 100

Form2

Enter side of square

10

Calculate Area

Calculate Perimeter

Perimeter of square is 40

program to find volume of box
width, depth and height are entered by user in textboxes

```
Private Sub Command1_Click()  
Dim width As Integer  
Dim depth As Integer  
Dim height As Integer  
Dim volume As Integer  
width = CInt(Text1.Text)  
depth = CInt(Text2.Text)  
height = CInt(Text3.Text)  
volume = width * depth * height  
Label4.Caption = "Volume of Box is " & volume  
End Sub
```

The screenshot shows a Windows application window titled "Form4". It contains three text boxes for input: "Enter Width of Box" with the value "10", "Enter Depth of Box" with the value "20", and "Enter Height of Box" with the value "30". Below these is a button labeled "Volume of Box". At the bottom left, a label displays the result: "Volume of Box is 6000".

program to find area and circumference of circle
radius is entered by user in textbox

```
Private Sub Command1_Click()  
Dim radius As Double  
Dim area As Double  
radius = Cdbl(Text1.Text)  
area = 3.14 * radius * radius  
Label2.Caption = "Area of Circle is " & area  
End Sub
```

```
Private Sub Command2_Click()  
Dim radius As Double  
Dim circumference As Double  
radius = Cdbl(Text1.Text)  
circumference = 2 * 3.14 * radius  
Label2.Caption = "Circumference of Circle is " & circumference  
End Sub
```

Form5

Enter Radius of Circle

Calculate Area of Circle Calculate Circumference of Circle

Area of Circle is 314

Form5

Enter Radius of Circle

Calculate Area of Circle Calculate Circumference of Circle

Circumference of Circle is 62.8

program to find simple interest based on principal amount , rate and time

```

Private Sub Command1_Click()
Dim principal As Double
Dim rate As Double
Dim time As Double
Dim si As Double
principal = Cdbl(Text1.Text)
rate = Cdbl(Text2.Text)
time = Cdbl(Text3.Text)
si = principal * rate * time
Label4.Caption = "Simple Interest is " & si
End Sub

```

Form6

Principal Amount: 1000

Rate of Interest in percentage: 3

Time in Years: 5

Calculate Simple Interest

Simple Interest is 15000

program to find 10% discount based on price entered by user in textbox

```
Private Sub Command1_Click()  
Dim price As Double  
Dim discount As Double  
price = Cdbl(Text1.Text)  
discount = price * 0.1  
Label2.Caption = "Discount is " & discount  
End Sub
```

Form7

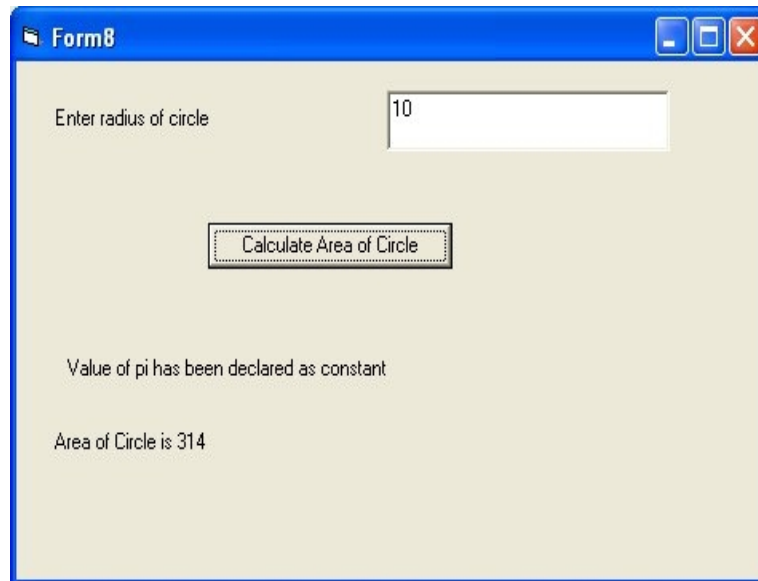
Enter price of product: 1000

Calculate Discount

Discount is 100

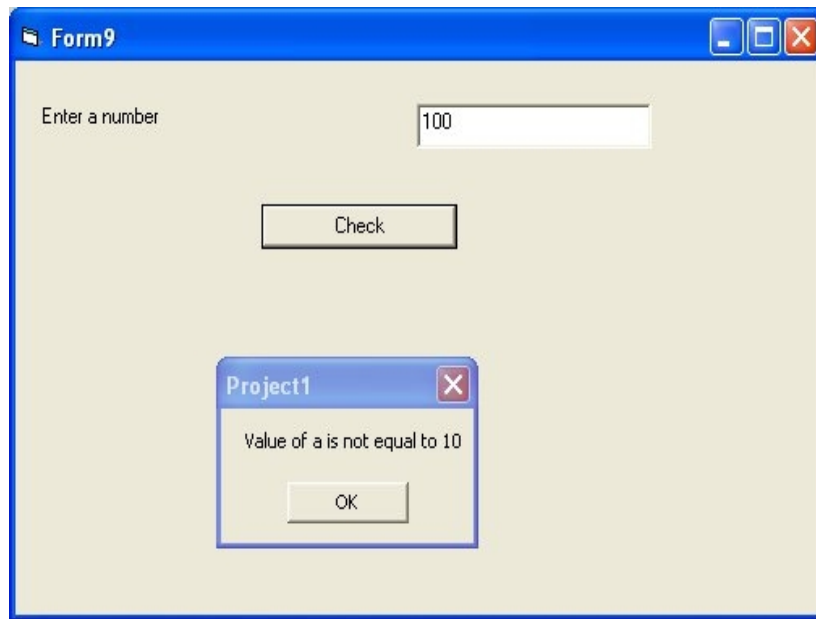
program to declare a constant pi with value as 3.14

```
Private Sub Command1_Click()  
Const pi As Double = 3.14  
Dim radius As Double  
Dim area As Double  
radius = Cdbl(Text1.Text)  
area = pi * radius * radius  
Label3.Caption = "Area of Circle is " & area  
End Sub
```



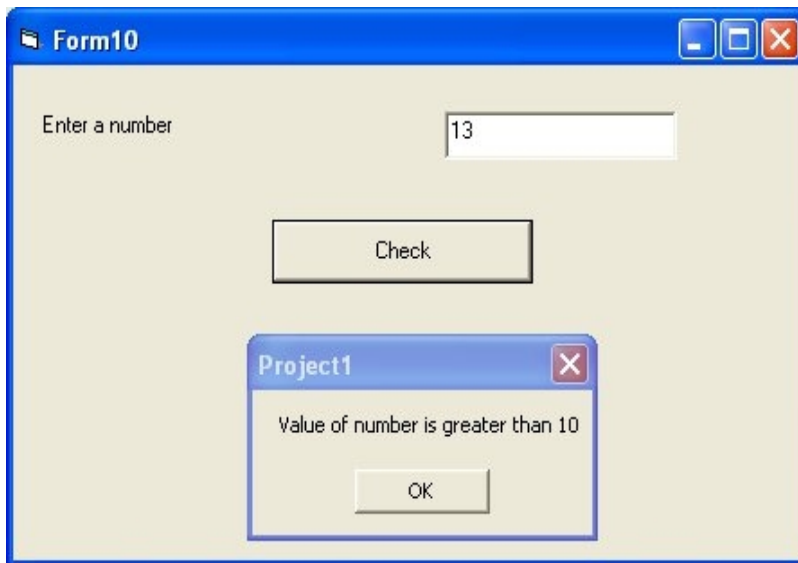
program to demonstrate if statement

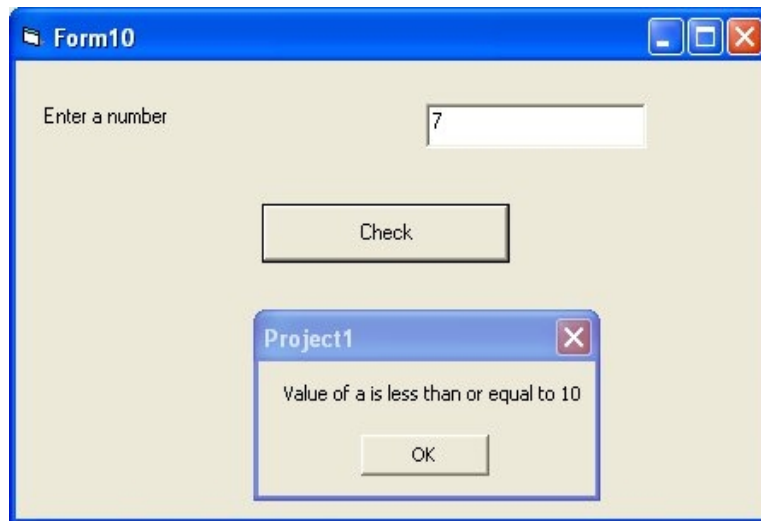
```
Private Sub Command1_Click()  
Dim a As Integer  
a = CInt(Text1.Text)  
If a = 10 Then  
MsgBox ("Value of a is 10")  
Else  
MsgBox ("Value of a is not equal to 10")  
End If  
End Sub
```



program to check whether number is greater than 10 or not

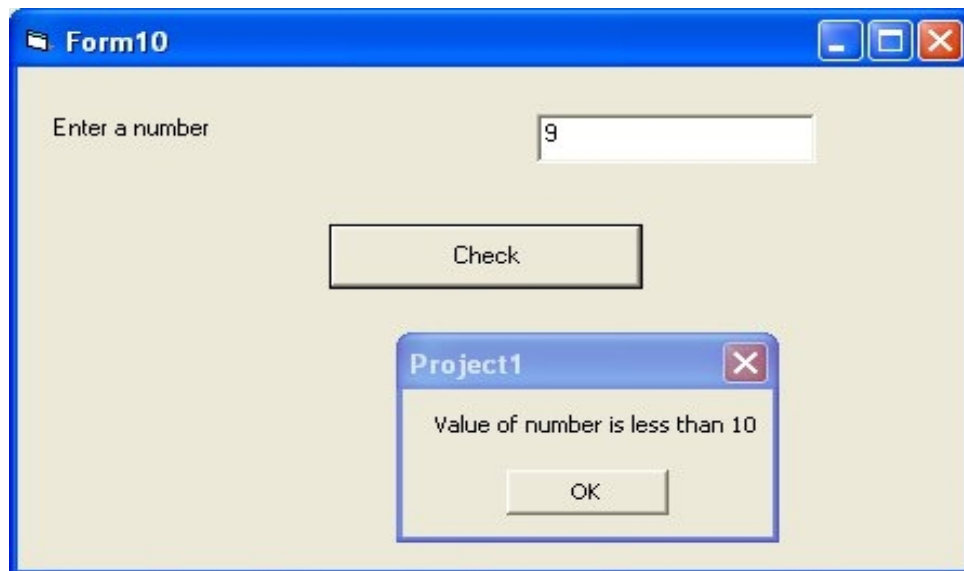
```
Private Sub Command1_Click()  
Dim a As Integer  
a = CInt(Text1.Text)  
If a > 10 Then  
MsgBox ("Value of number is greater than 10")  
Else  
MsgBox ("Value of a is less than or equal to 10")  
End If  
End Sub
```





program to check whether a is greater than 10 or less than 10

```
Private Sub Command1_Click()  
Dim a As Integer  
a = CInt(Text1.Text)  
If a < 10 Then  
MsgBox ("Value of number is less than 10")  
Else  
MsgBox ("Value of a is greater than or equal to 10")  
End If  
End Sub
```



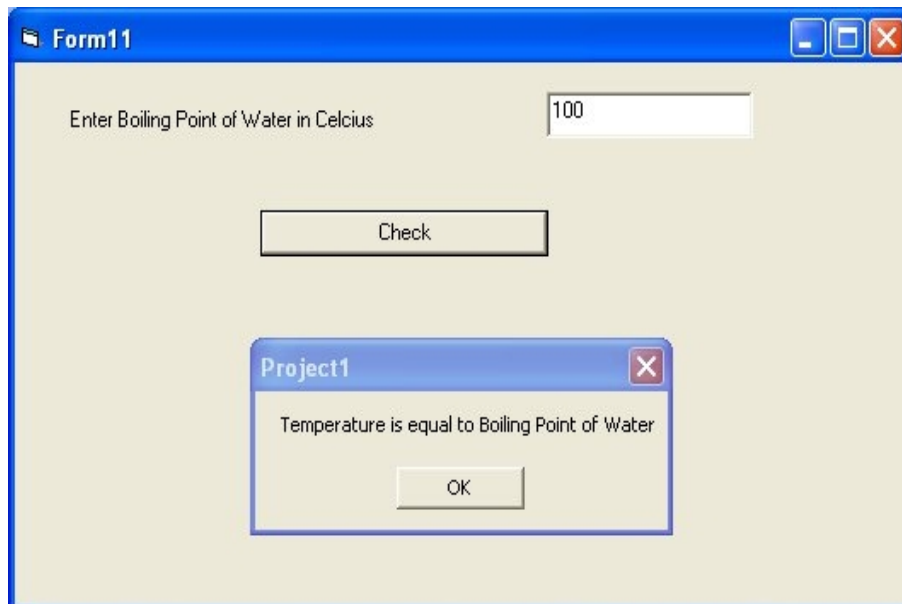
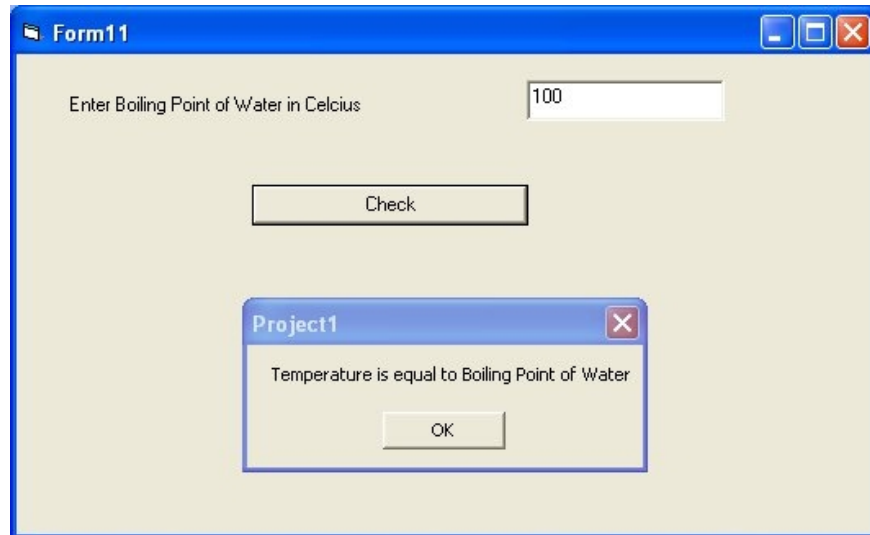
program to input temperature from user and check whether it is boiling point of water is not

```
Private Sub Command1_Click()  
Dim a As Integer  
a = CInt(Text1.Text)
```

```

If a < 10 Then
MsgBox ("Value of number is less than 10")
Else
MsgBox ("Value of a is greater than or equal to 10")
End If
End Sub

```



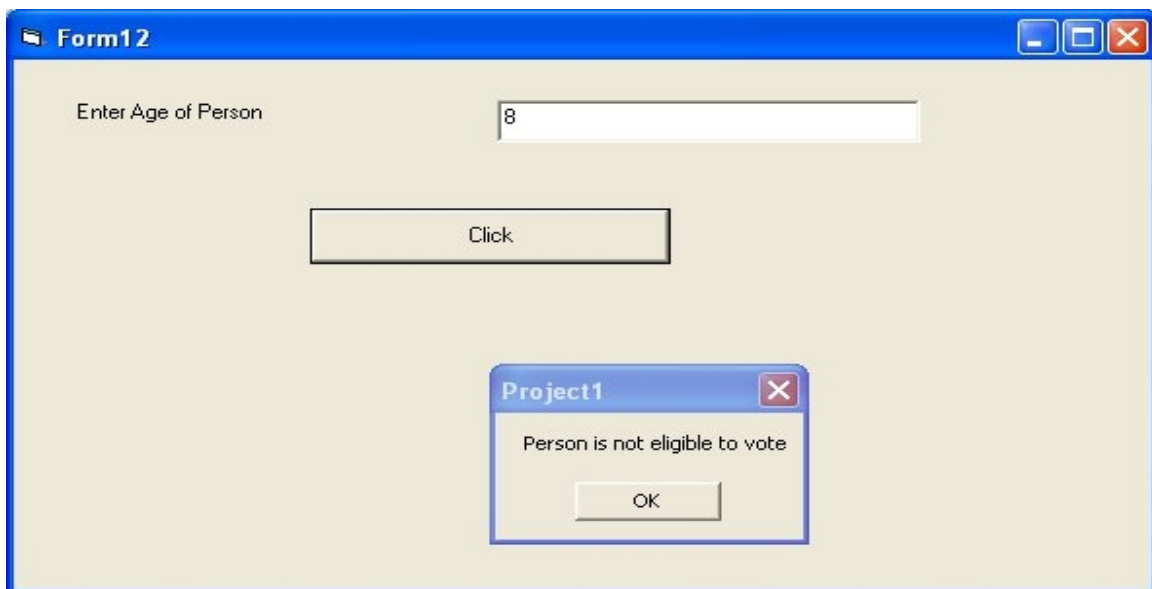
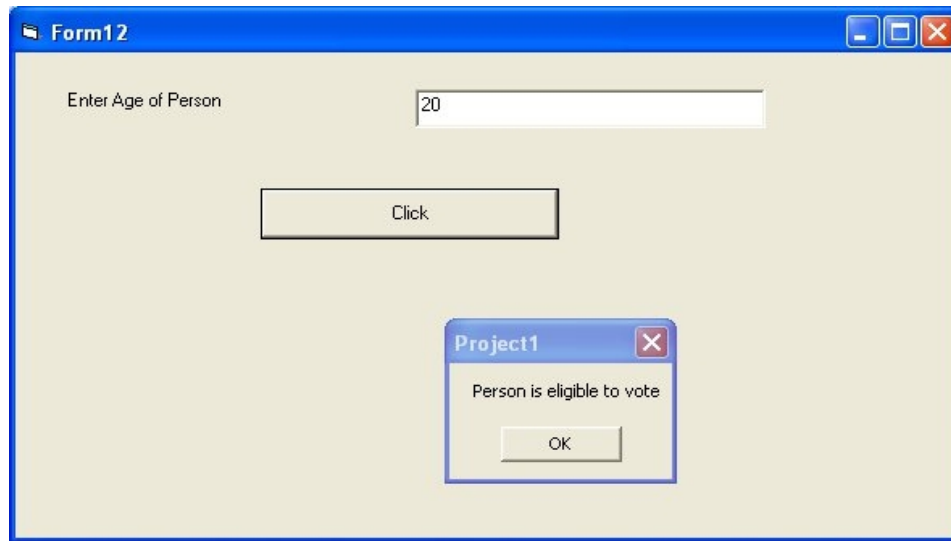
program to calculate check eligibility of person to vote based on age

```

Private Sub Command1_Click()
Dim age As Integer
age = CInt(Text1.Text)
If age >= 18 Then
MsgBox ("Person is eligible to vote")
Else
MsgBox ("Person is not eligible to vote")

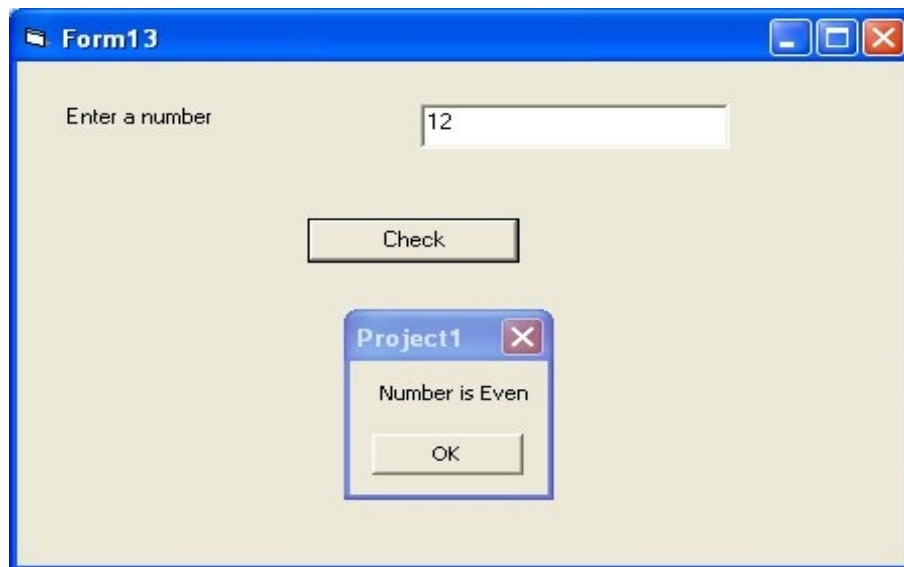
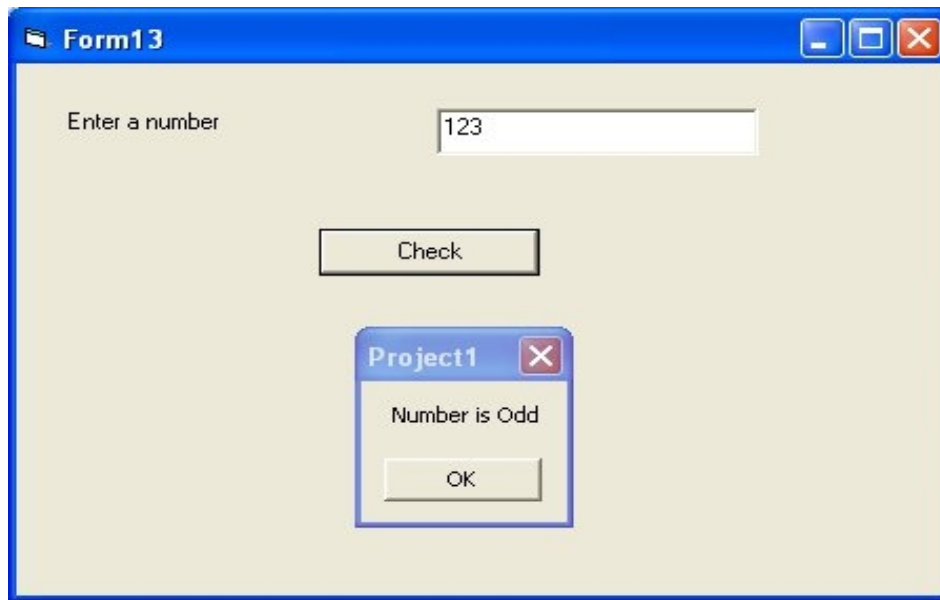
```

End If
End Sub



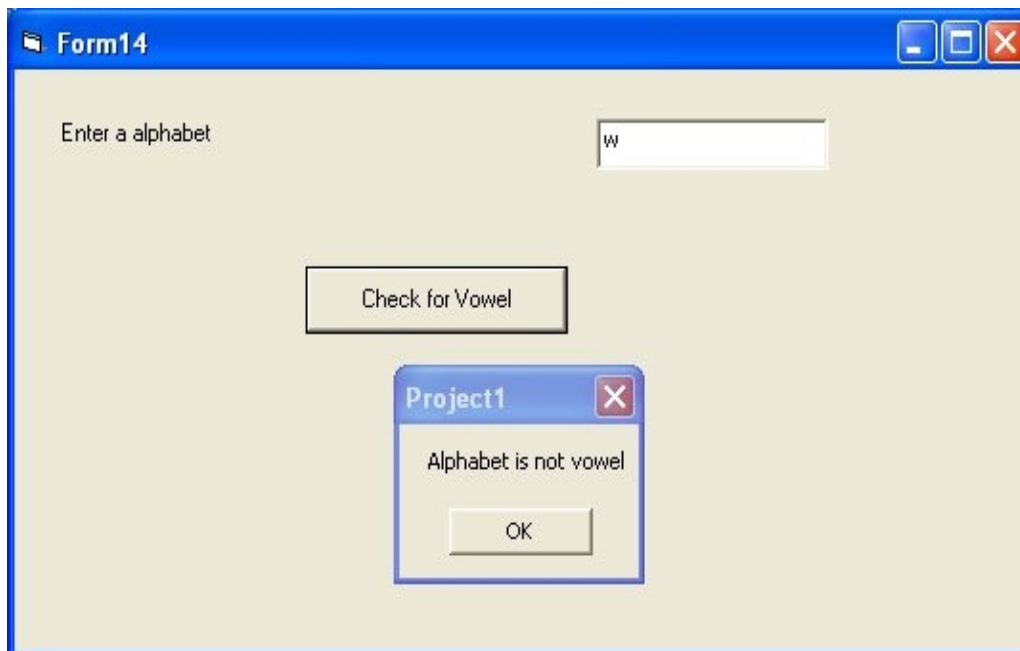
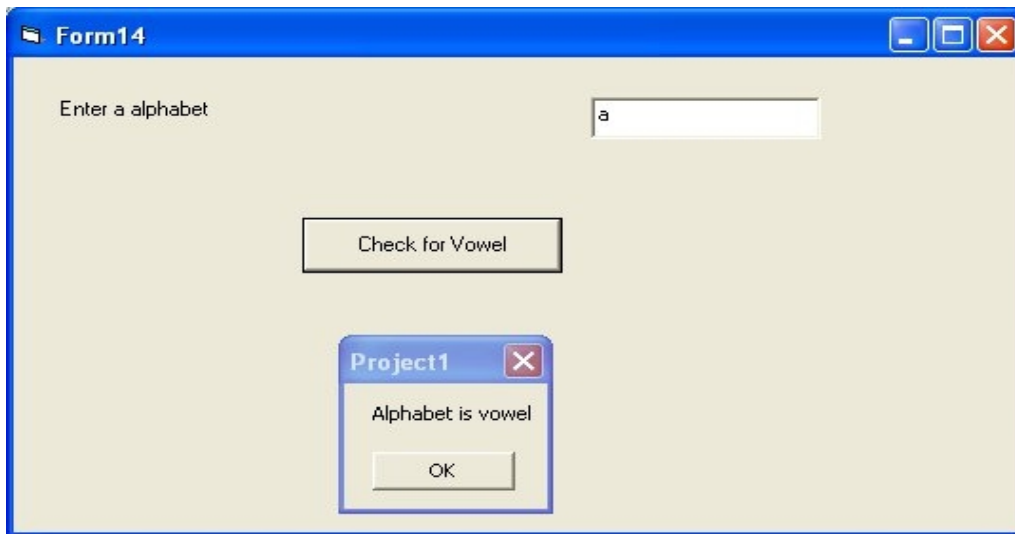
program to check whether number is even or odd

```
Private Sub Command1_Click()  
Dim a As Integer  
a = CInt(Text1.Text)  
If a Mod 2 = 0 Then  
MsgBox ("Number is Even")  
Else  
MsgBox ("Number is Odd")  
End If  
End Sub
```



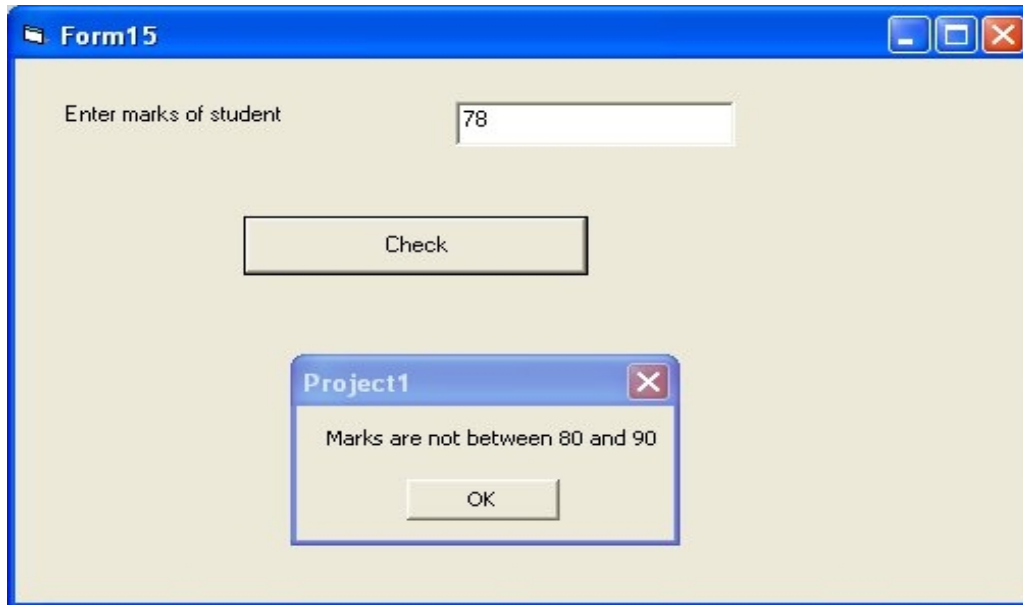
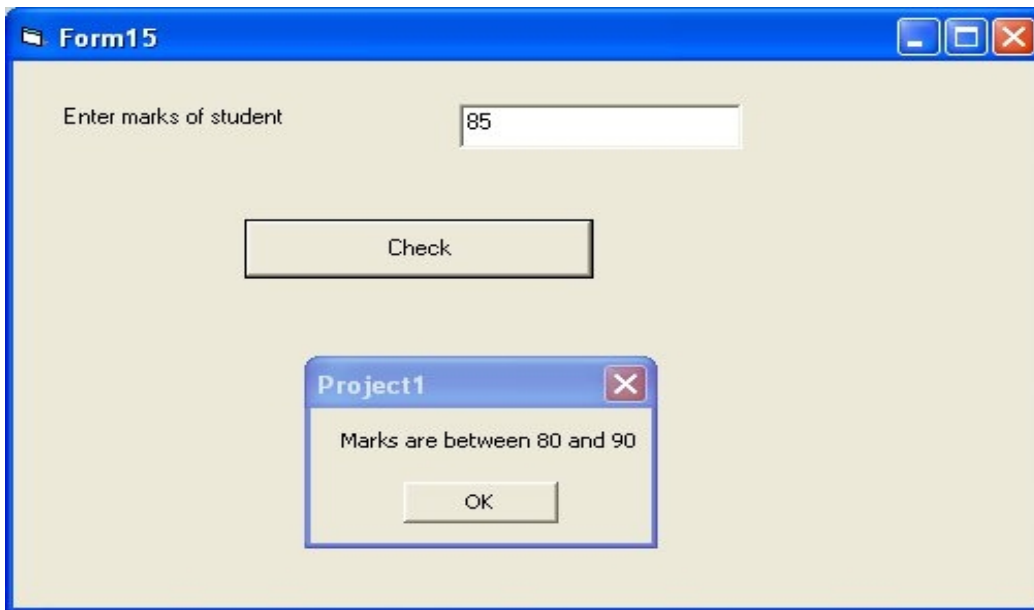
program to check an alphabet is vowel or not

```
Private Sub Command1_Click()  
Dim a As String  
a = Text1.Text  
If a = "a" Or a = "e" Or a = "i" Or a = "o" Or a = "u" Then  
MsgBox ("Alphabet is vowel")  
Else  
MsgBox ("Alphabet is not vowel")  
End If  
End Sub
```



program to enter marks from user and check whether marks are between 80 and 90 using and operator

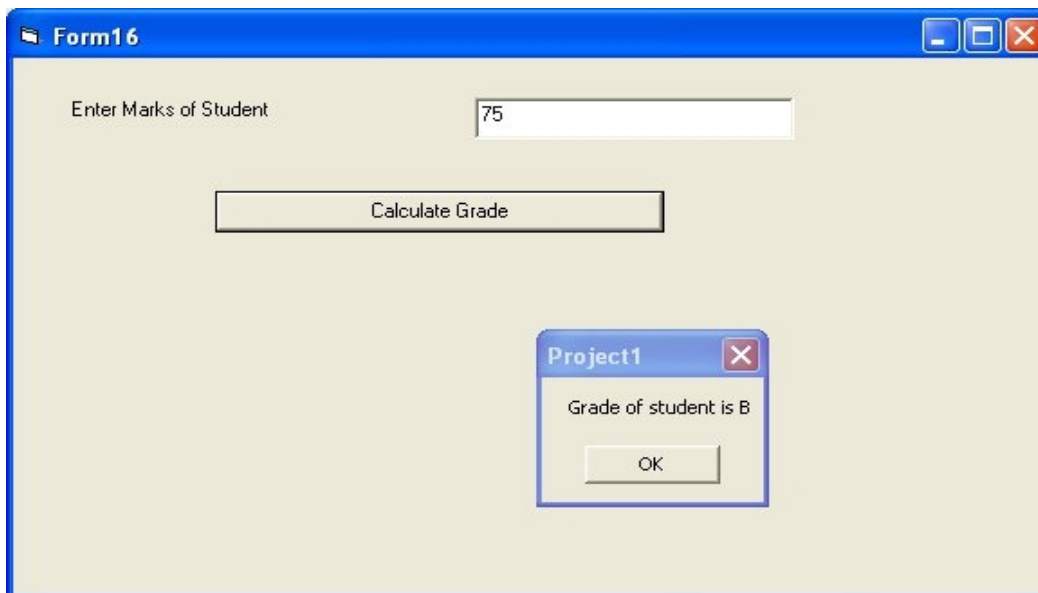
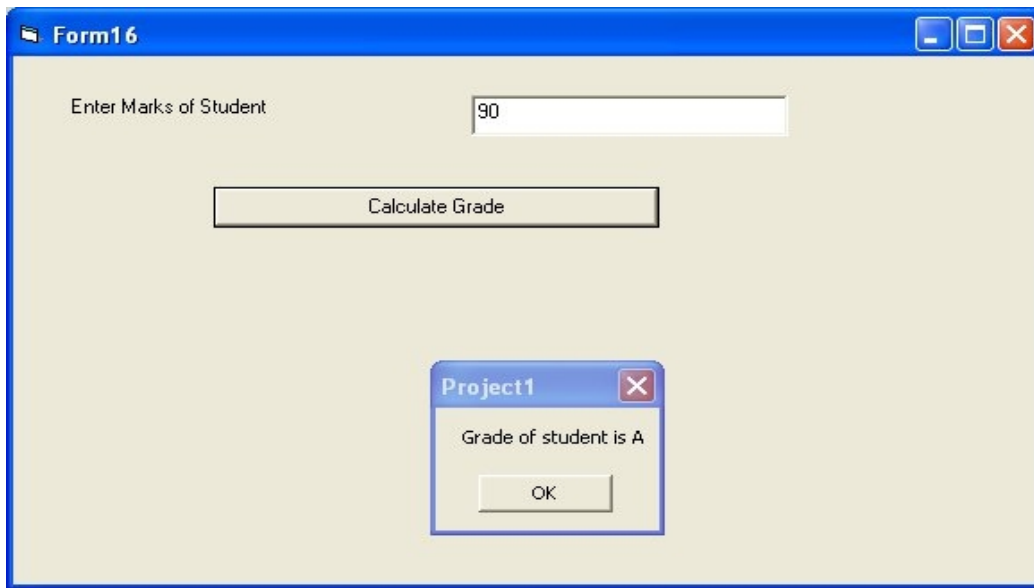
```
Private Sub Command1_Click()  
Dim marks As Integer  
marks = CInt(Text1.Text)  
If marks >= 80 And marks <= 90 Then  
MsgBox ("Marks are between 80 and 90")  
Else  
MsgBox ("Marks are not between 80 and 90")  
End If  
End Sub
```



program to calculate grade of student based on marks

Marks	grade
80 to 100	A
70 to 79	B
60 to 69	C
Less than 60	D

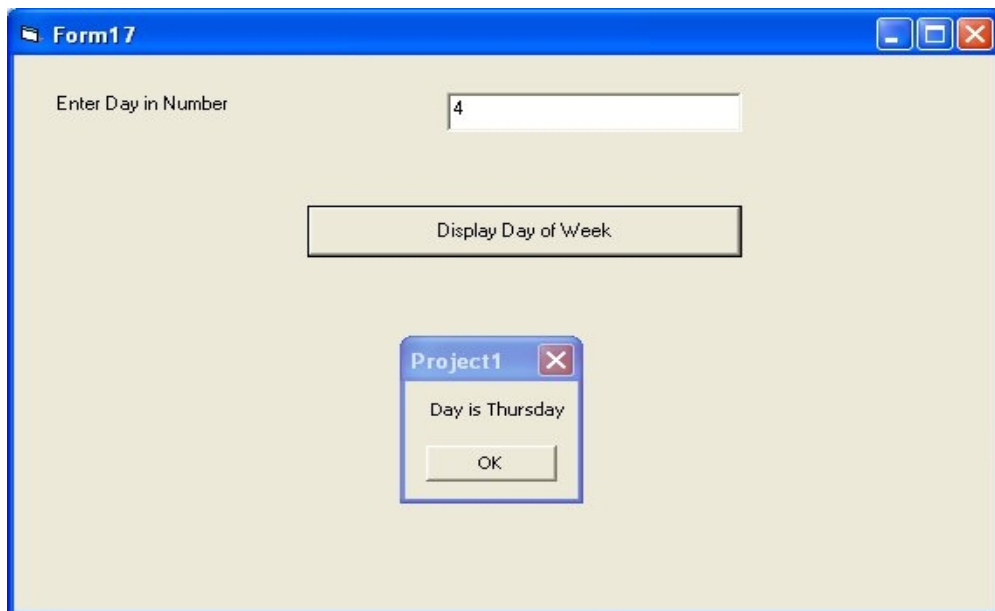

```
Private Sub Command1_Click()  
Dim marks As Integer  
Dim grade As String  
marks = CInt(Text1.Text)  
If marks >= 80 And marks <= 100 Then  
grade = "A"  
ElseIf marks >= 70 And marks < 80 Then  
grade = "B"  
ElseIf marks >= 60 And marks < 70 Then  
grade = "C"  
Else  
grade = "D"  
End If  
MsgBox ("Grade of student is " & grade)  
End Sub
```

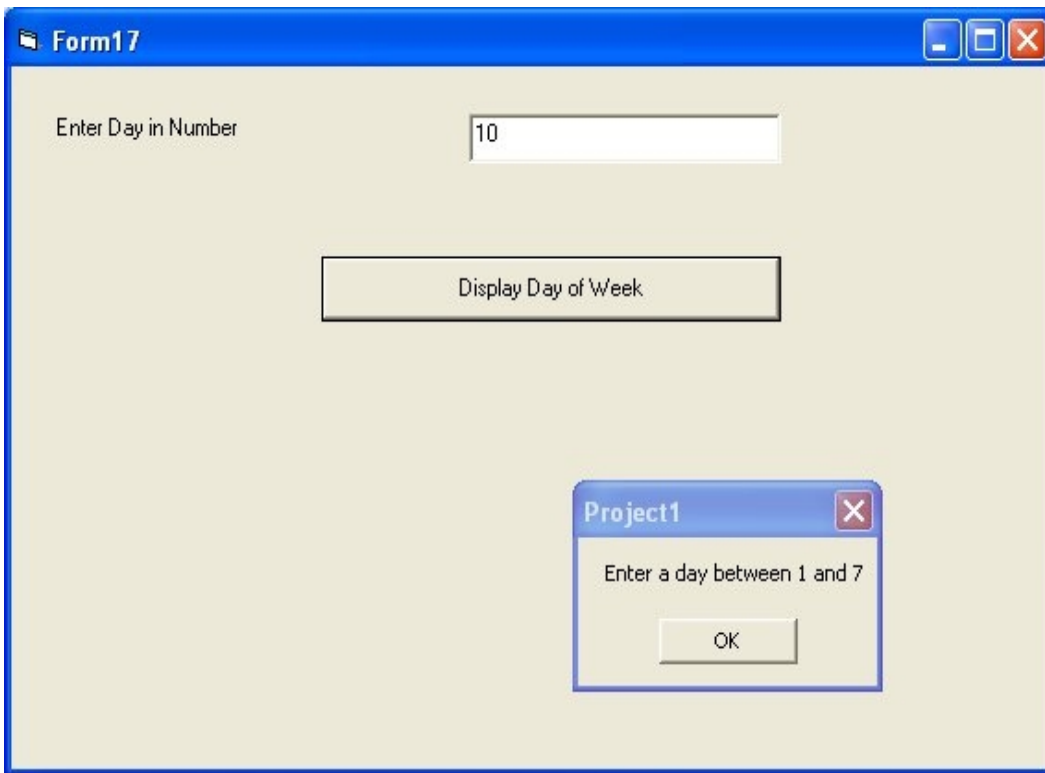


program to demonstrate select case statement

programs asks user for day in number and displays day in words in a messagebox

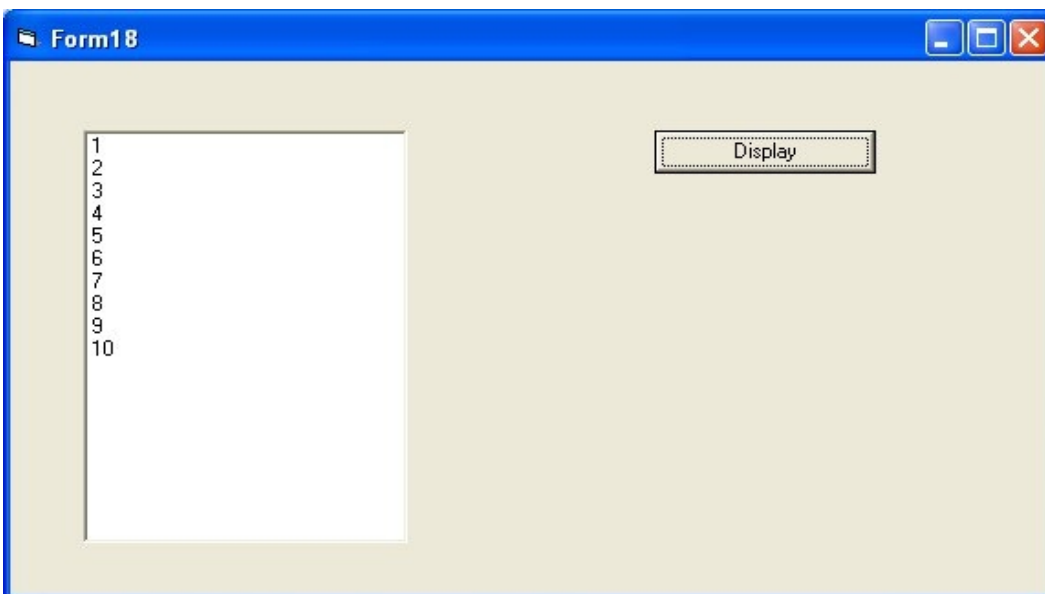
```
Private Sub Command1_Click()  
Dim a As Integer  
a = CInt(Text1.Text)  
Select Case a  
Case 1  
MsgBox "Day is Monday"  
Case 2  
MsgBox "Day is Tuesday"  
Case 3  
MsgBox "Day is Wednesday"  
Case 4  
MsgBox "Day is Thursday"  
Case 5  
MsgBox "Day is Friday"  
Case 6  
MsgBox "Day is Saturday"  
Case 7  
MsgBox "Day is Sunday"  
Case Else  
MsgBox "Enter a day between 1 and 7"  
End Select  
End Sub
```





program to demonstrate for loop to print numbers from 1 to 10 in a ListBox

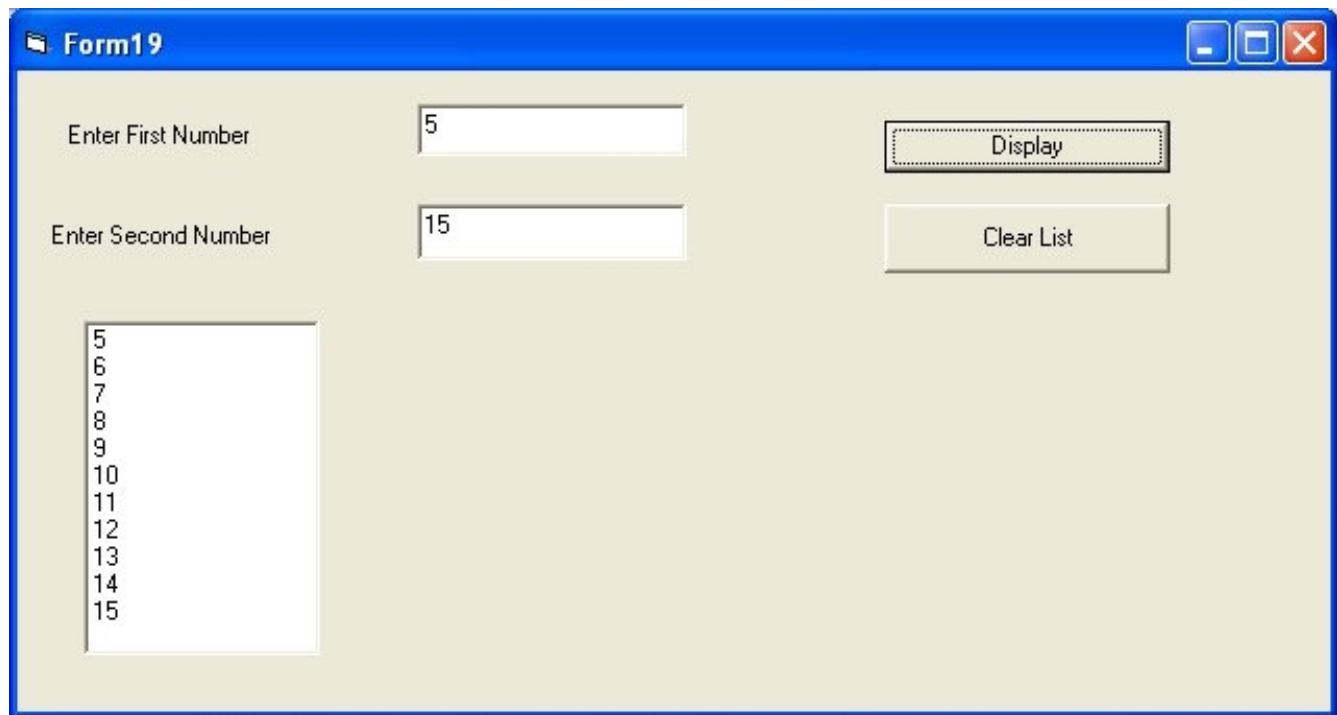
```
Private Sub Command1_Click()  
Dim i As Integer  
For i = 1 To 10  
List1.AddItem (i)  
Next  
End Sub
```



program to input two numbers and display list of all numbers between these two numbers

```
Private Sub Command1_Click()  
Dim a As Integer  
Dim b As Integer  
Dim i As Integer  
a = CInt(Text1.Text)  
b = CInt(Text2.Text)  
For i = a To b  
List1.AddItem (i)  
Next  
End Sub
```

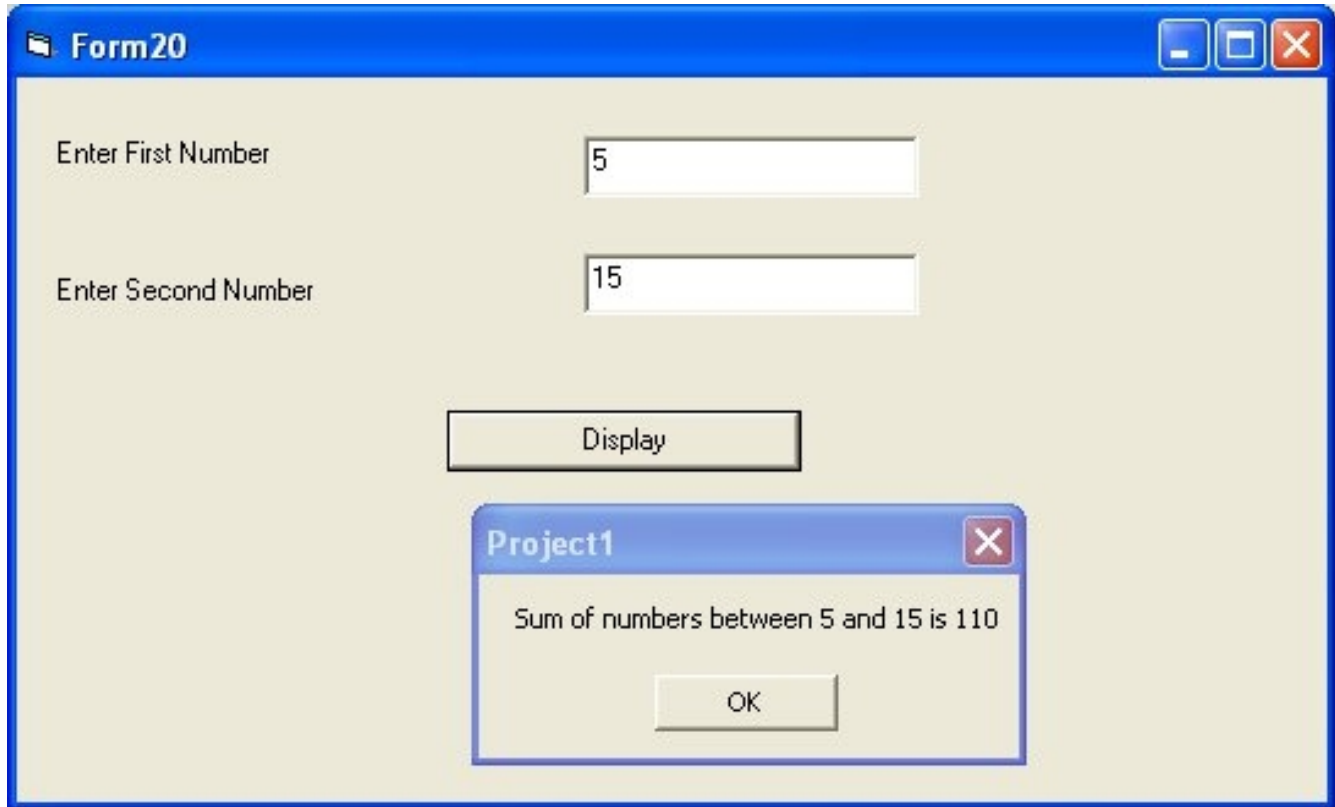
```
Private Sub Command2_Click()  
List1.Clear  
End Sub
```



program to display sum between two numbers and display sum of numbers between them
we are assuming number 1 is less than number 2

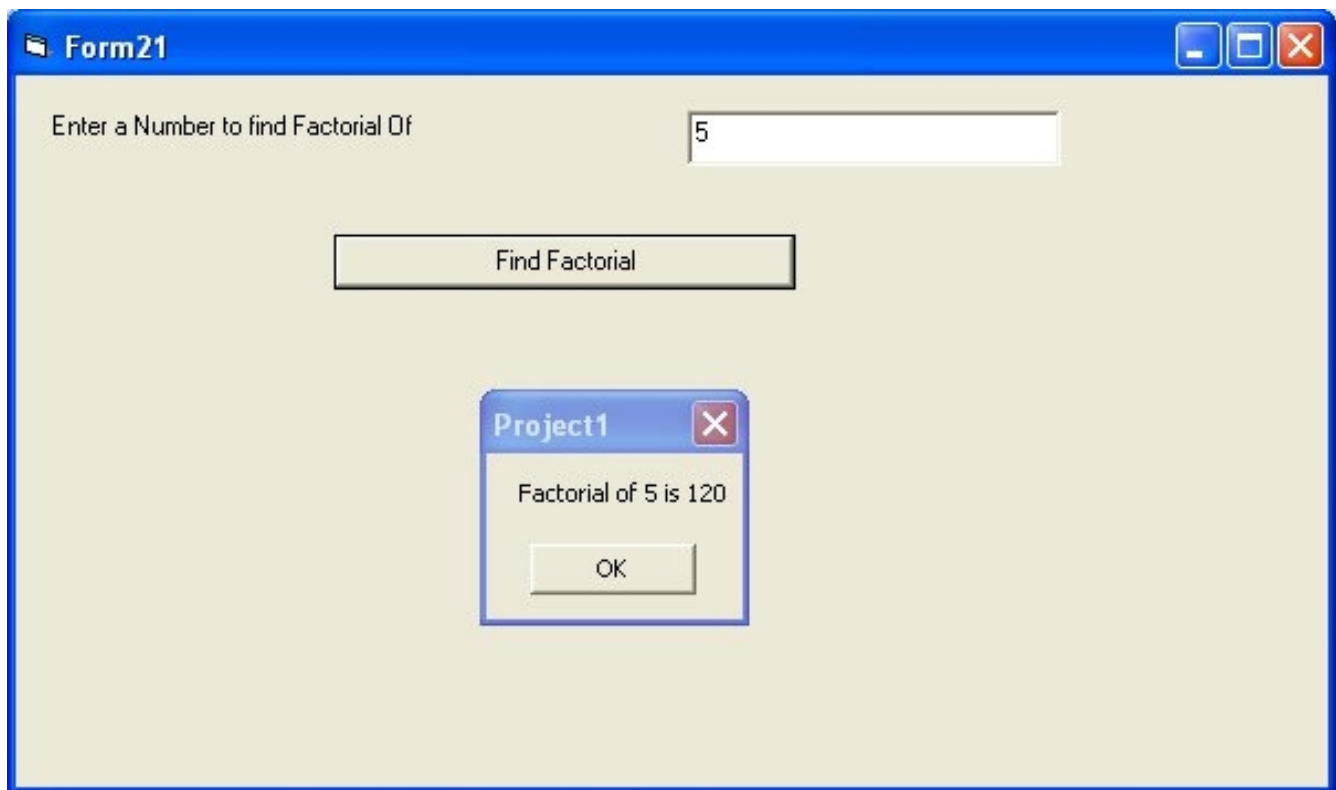
```
Private Sub Command1_Click()  
Dim a As Integer  
Dim b As Integer  
Dim i As Integer  
Dim sum As Integer  
sum = 0  
a = CInt(Text1.Text)  
b = CInt(Text2.Text)
```

```
For i = a To b
sum = sum + i
Next
MsgBox ("Sum of numbers between " & a & " and " & b & " is " & sum)
End Sub
```



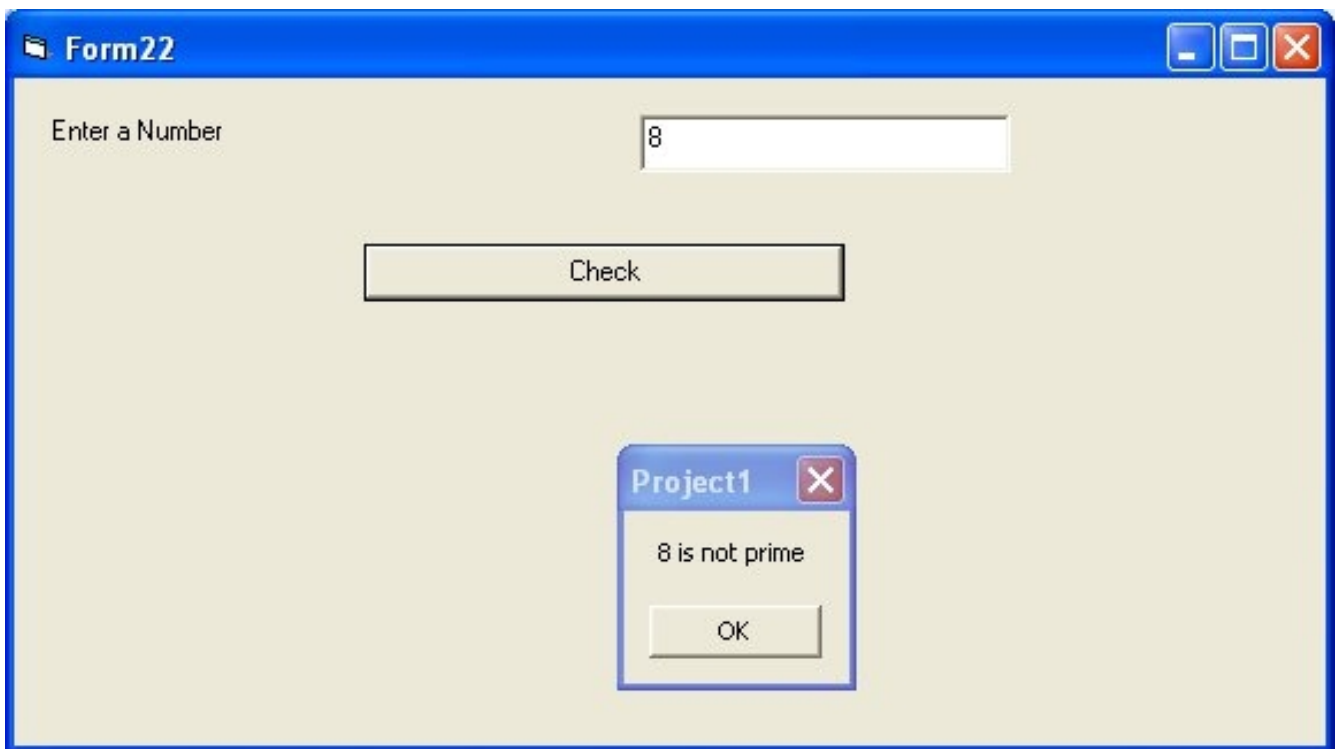
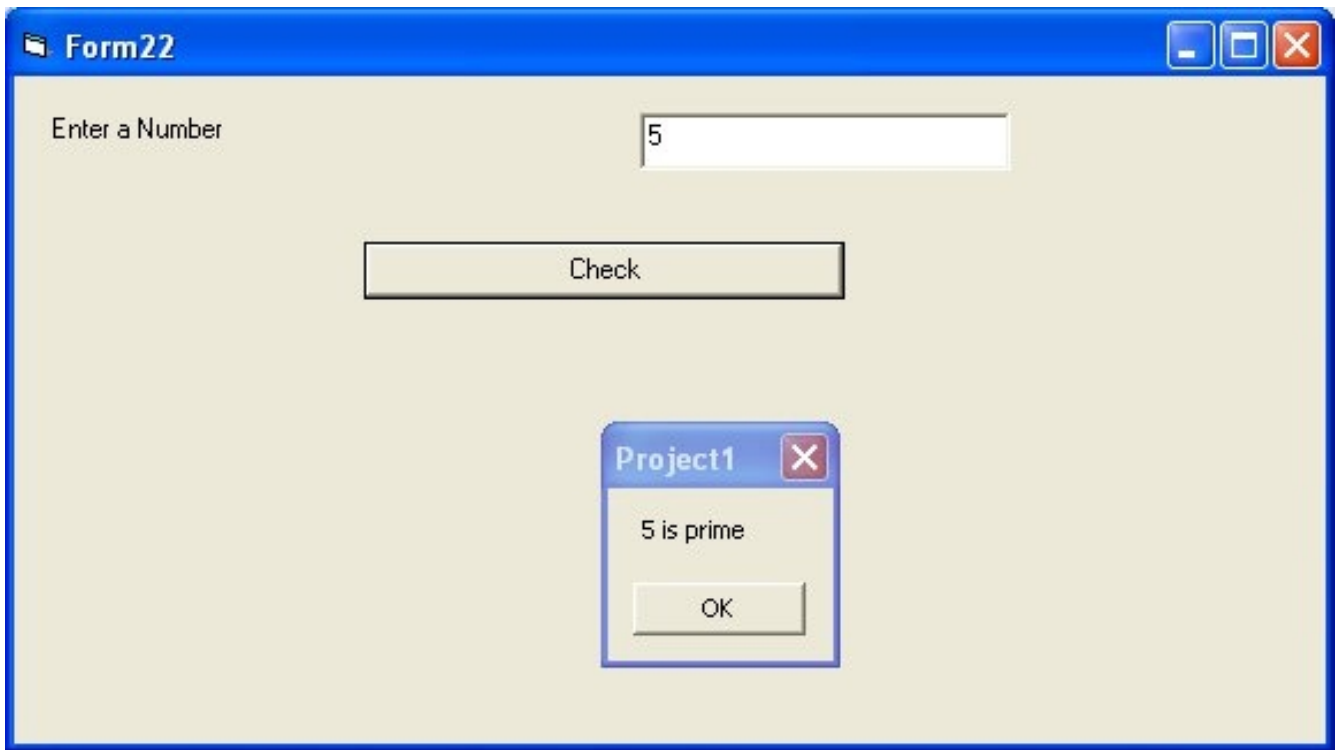
program to find factorial of a number

```
Private Sub Command1_Click()
Dim i As Integer
Dim a As Integer
Dim fact As Integer
fact = 1
a = CInt(Text1.Text)
For i = 1 To a
fact = fact * i
Next
MsgBox ("Factorial of " & a & " is " & fact)
End Sub
```



program to find whether number is prime or not

```
Private Sub Command1_Click()  
Dim a As Integer  
Dim i As Integer  
Dim prime As Boolean  
prime = True  
a = CInt(Text1.Text)  
For i = 2 To a / 2  
If a Mod i = 0 Then  
prime = False  
Exit For  
End If  
Next  
If prime = True Then  
MsgBox (a & " is prime")  
Else  
MsgBox (a & " is not prime")  
End If  
End Sub
```



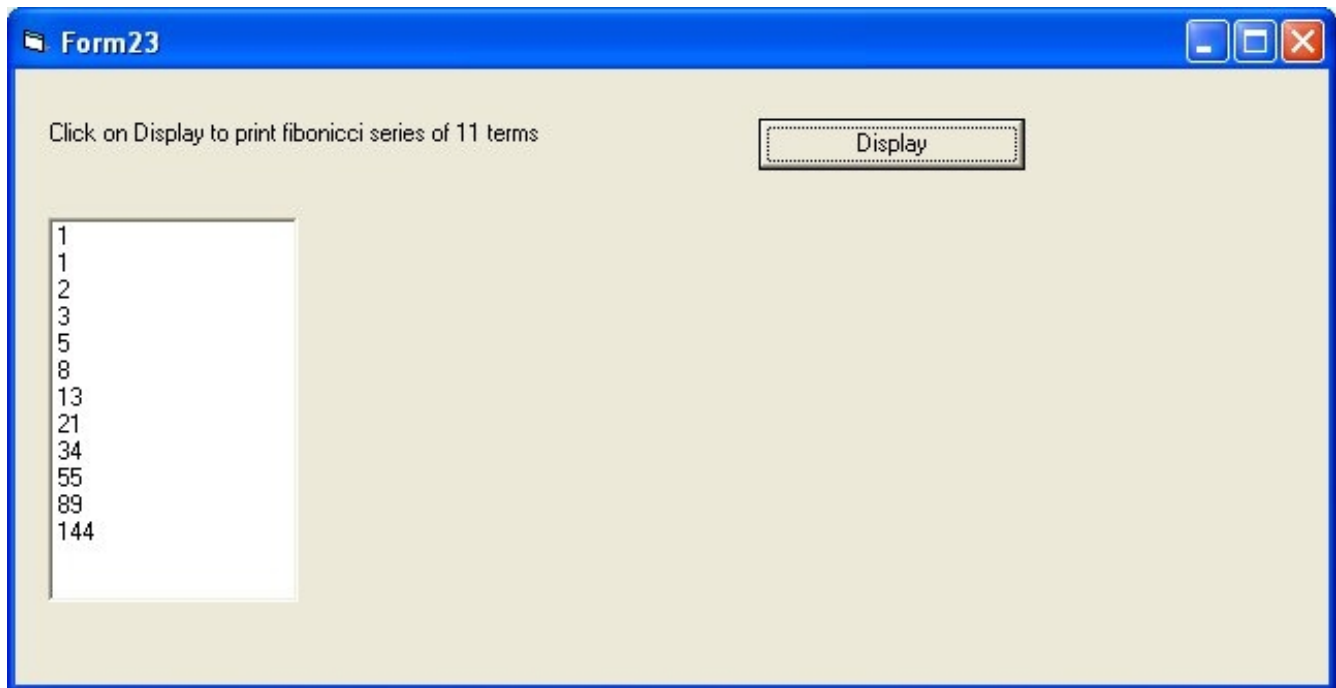
program to print fibonacchi series in a listbox

```
Private Sub Command1_Click()  
Dim i As Integer  
Dim a As Integer  
Dim b As Integer
```

```

Dim c As Integer
a = 1
b = 1
List1.AddItem (a)
List1.AddItem (b)
For i = 1 To 10
c = a + b
List1.AddItem (c)
a = b
b = c
Next
End Sub

```

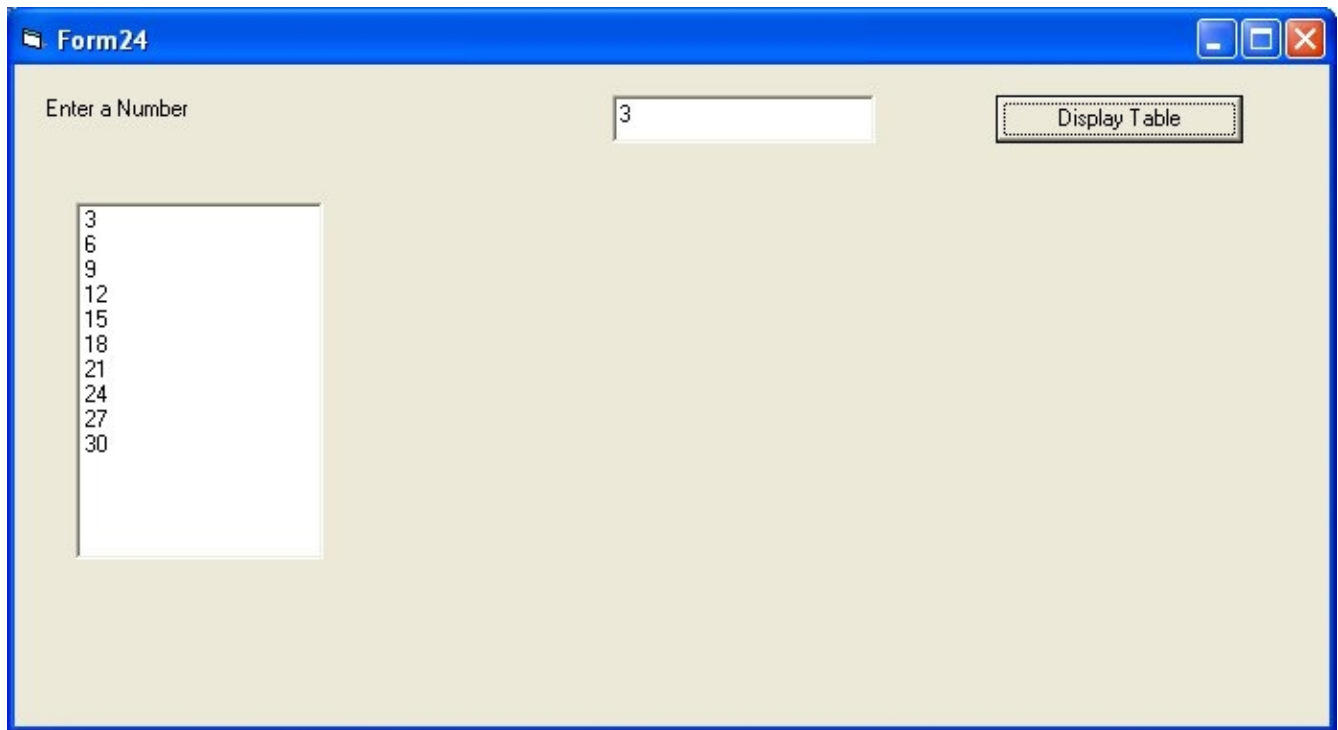


program to display table of a number from 1 to 10 in a listbox

```

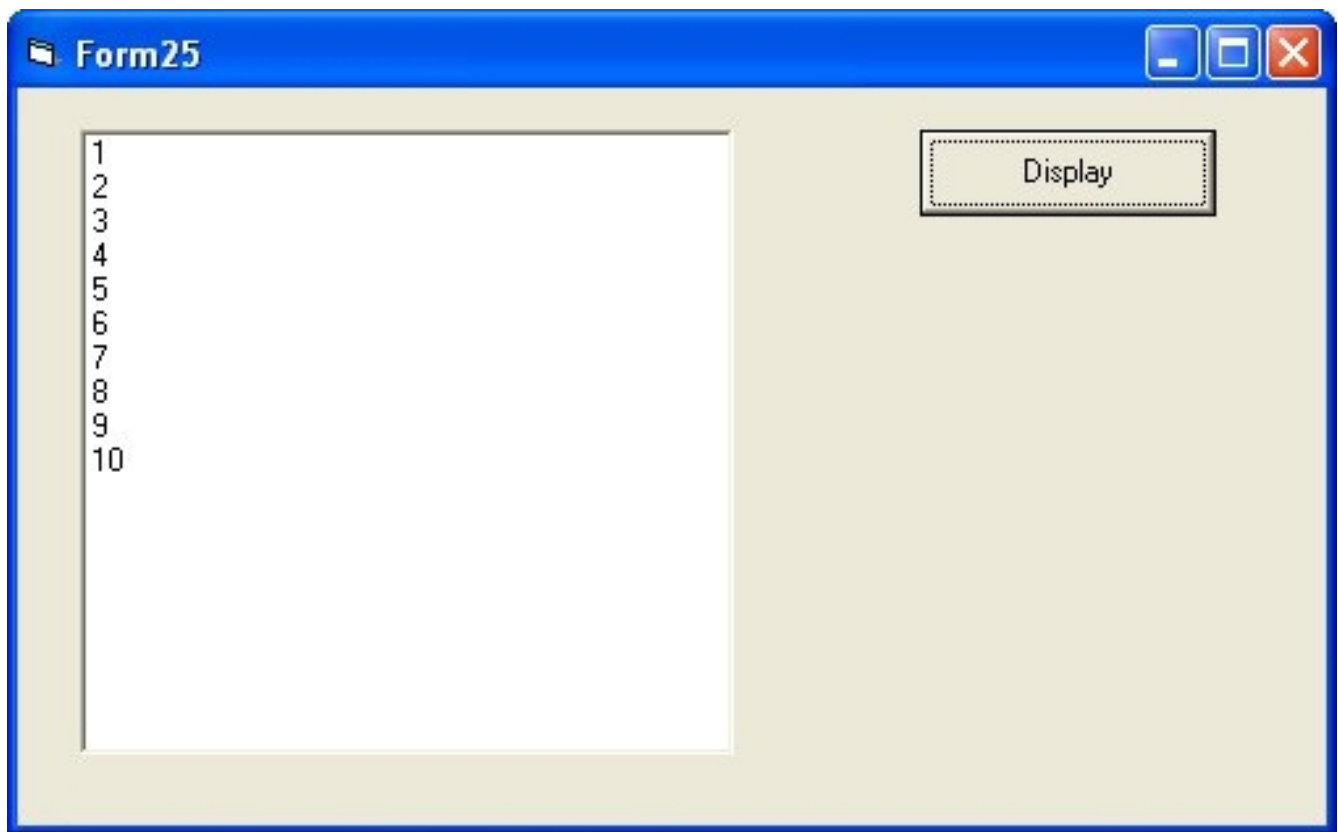
Private Sub Command1_Click()
Dim i As Integer
Dim a As Integer
a = CInt(Text1.Text)
For i = 1 To 10
List1.AddItem (a * i)
Next
End Sub

```

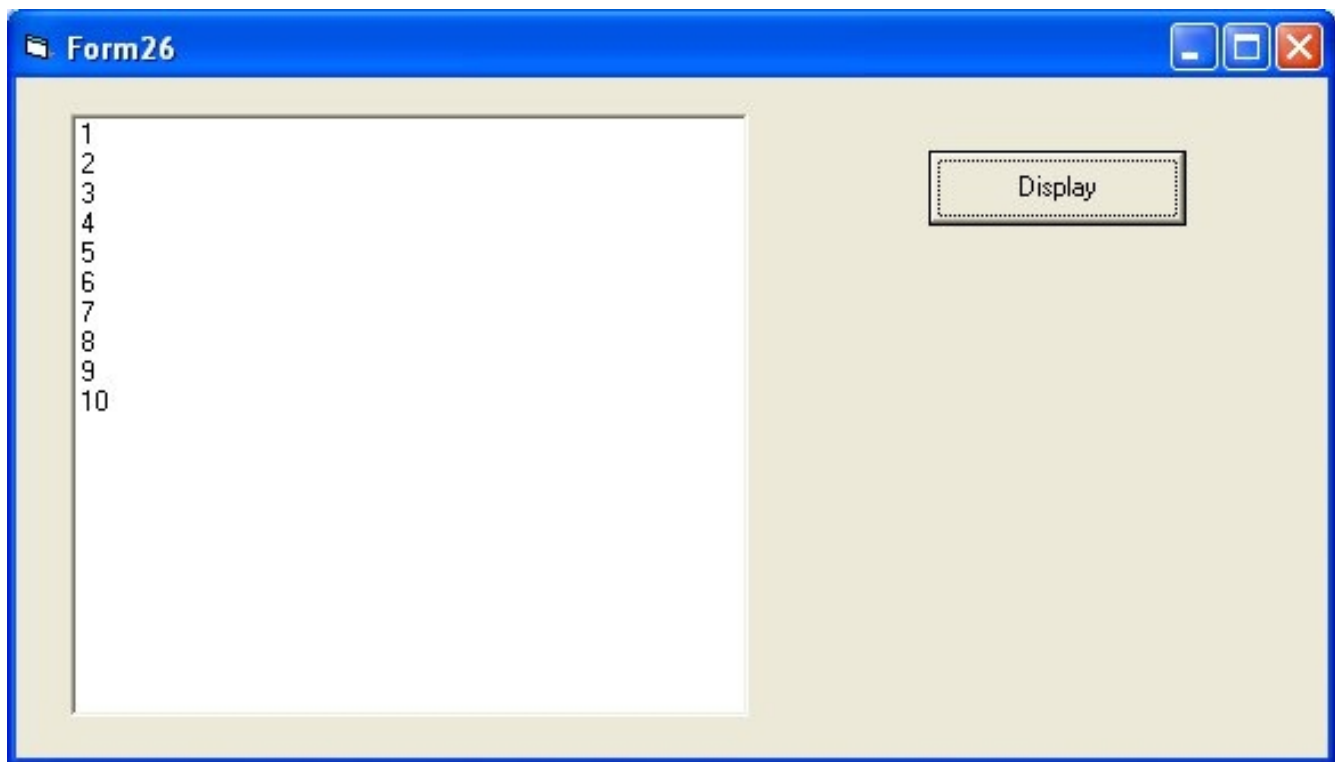
program to demonstrate while loop to display numbers from 1 to 10 using while wend loop

```
Private Sub Command1_Click()  
Dim i As Integer  
i = 1  
While i <= 10  
List1.AddItem (i)  
i = i + 1  
Wend  
End Sub
```



program to demonstrate do while loop statement

```
Private Sub Command1_Click()  
Dim number As Integer  
number = 1  
Do While number <= 10  
List1.AddItem (number)  
number = number + 1  
Loop  
End Sub
```



Do...Loop While Statement

The Do...Loop While statement first executes the statements and then test the condition after each execution. The following program block illustrates the structure:

```
Dim number As Long
number = 0
Do
number = number + 1
Loop While number < 201
```

Do Until...Loop Statement

Unlike the Do While...Loop and While...Wend repetition structures, the Do Until... Loop structure tests a condition for falsity. Statements in the body of a Do Until...Loop are executed repeatedly as long as the loop-continuation test evaluates to False.

An example for Do Until...Loop statement. The coding is typed inside the click event of the command button

```
Dim number As Long
number=0
Do Until number > 1000
number = number + 1
Print number
Loop
```

With Statement

With...End With statement

When properties are set for objects or methods are called, a lot of coding is included that acts on the same object. It is easier to read the code by implementing the With...End With statement. Multiple properties can be set and multiple methods can be called by using the With...End With statement. The code is executed more quickly and efficiently as the object is evaluated only once. The concept can be clearly understood with following example.

```
With Text1
.Font.Size = 14
.Font.Bold = True
.ForeColor = vbRed
.Height = 230
.Text = "Hello World"
End With
```

program to add an item to listbox on click of button1
and get selected item from listbox on click of button2

```
Private Sub Command1_Click()
List1.AddItem (Text1.Text)
End Sub
```

```
Private Sub Command2_Click()
Label1.Caption = List1.List(List1.ListIndex)
End Sub
```

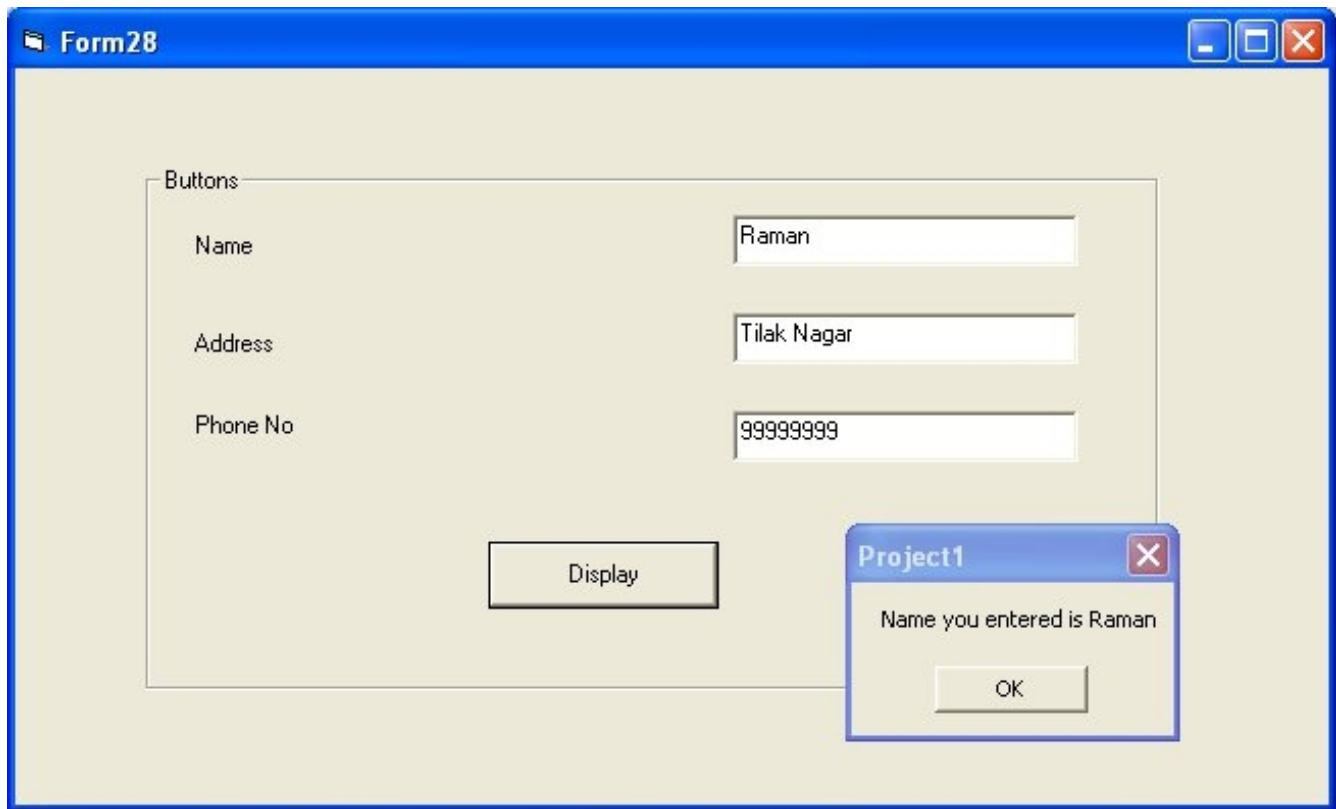
Examples of Controls on Visual Basic 6.0 Toolbox

Frame Control

```
Private Sub Command1_Click()
MsgBox ("Name you entered is " & Text1.Text)
End Sub
```

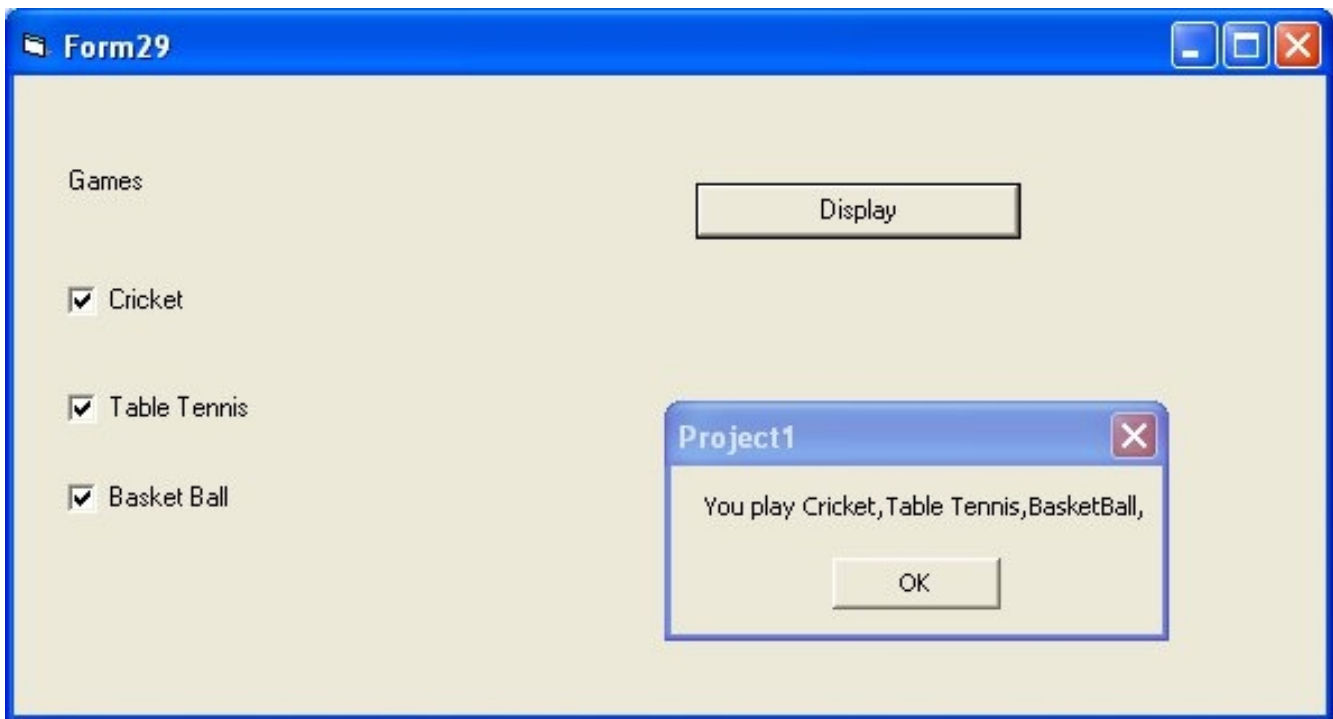
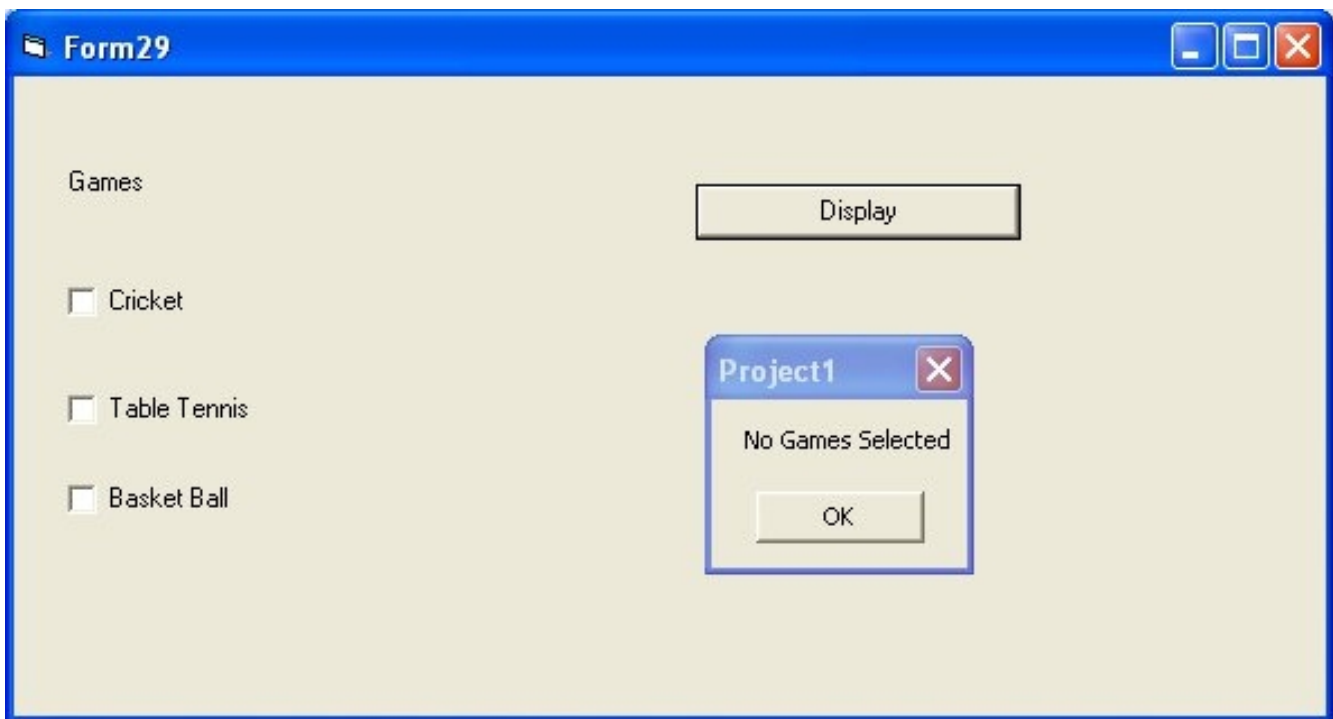
Name on Frame can be changed through Caption Property

```
Frame1.Caption= "Buttons"
```



Example of ComboBox

```
Private Sub Command1_Click()  
Dim a As String  
a = ""  
If Check1.Value = vbChecked Then  
a = a & "Cricket,"  
End If  
If Check2.Value = vbChecked Then  
a = a & "Table Tennis,"  
End If  
If Check3.Value = vbChecked Then  
a = a & "BasketBall,"  
End If  
If a = "" Then  
MsgBox ("No Games Selected")  
Else  
MsgBox ("You play " & a)  
End If  
End Sub
```



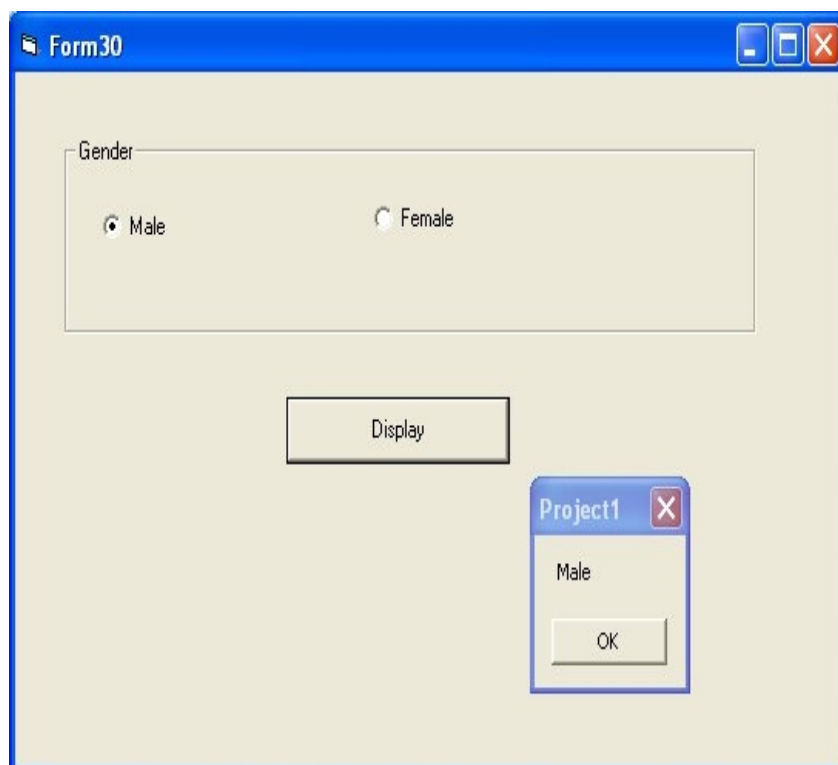
vbChecked is a constant in Visual Basic and a constant starts with vb in Visual Basic

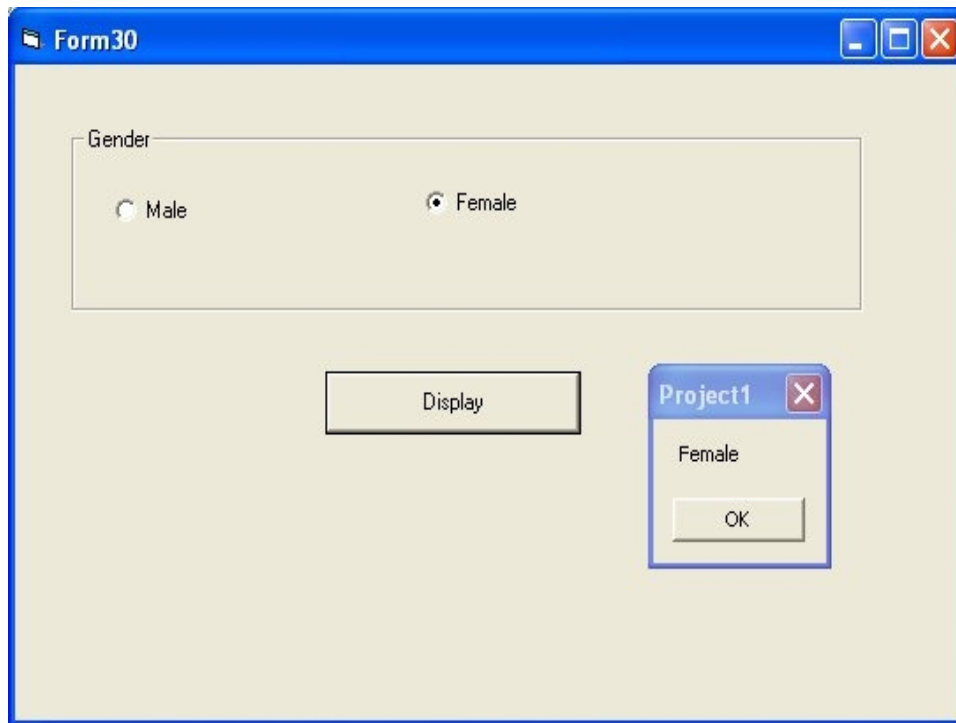
Radio Buttons or Option Buttons in Visual Basic 6.0

If you want to place two radio buttons in form you can use a frame

only a single option button can be selected in a frame

```
Private Sub Command1_Click()  
If Option1.Value = True Then  
MsgBox ("Male")  
Else  
MsgBox ("Female")  
End If  
End Sub
```

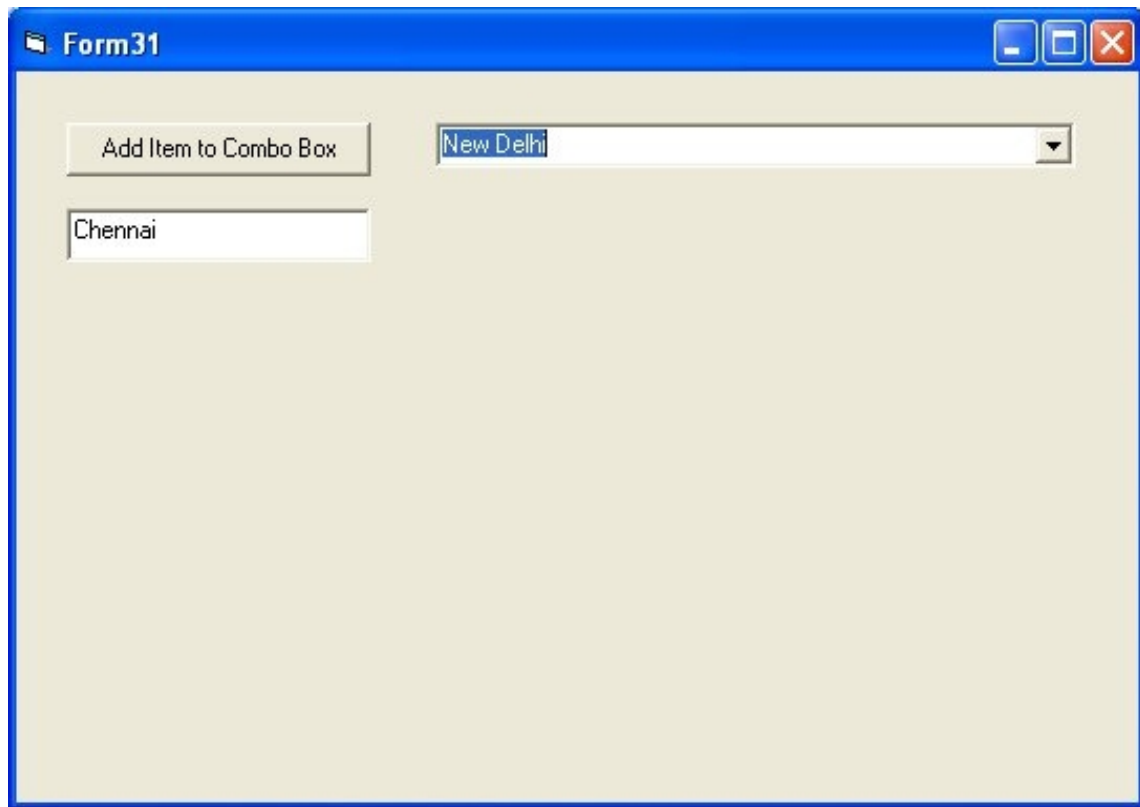




if you want to select a radio button or an option button by default, you can use value property to be selected as True.

Example of Combo Box Control

```
Private Sub Command1_Click()  
Combo1.AddItem (Text1.Text)  
MsgBox (Text1.Text & " added to ComboBox")  
End Sub
```

Horizontal Scroll Bar

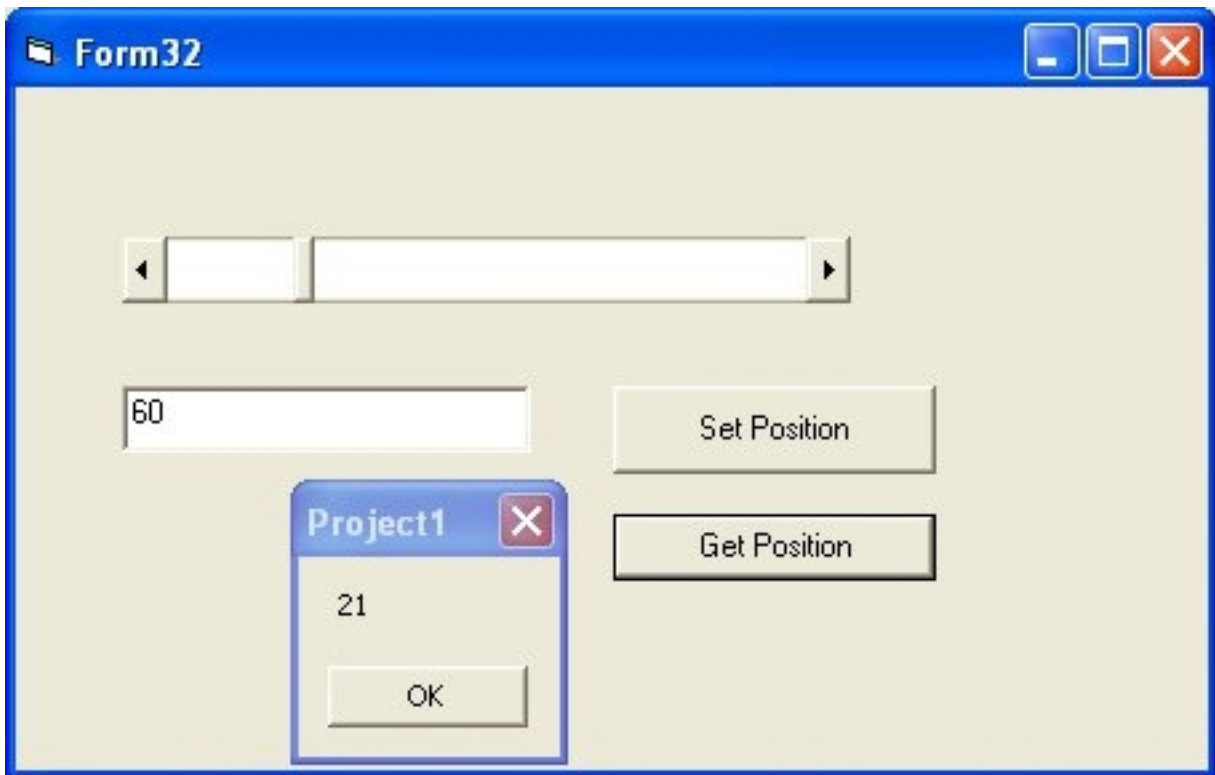
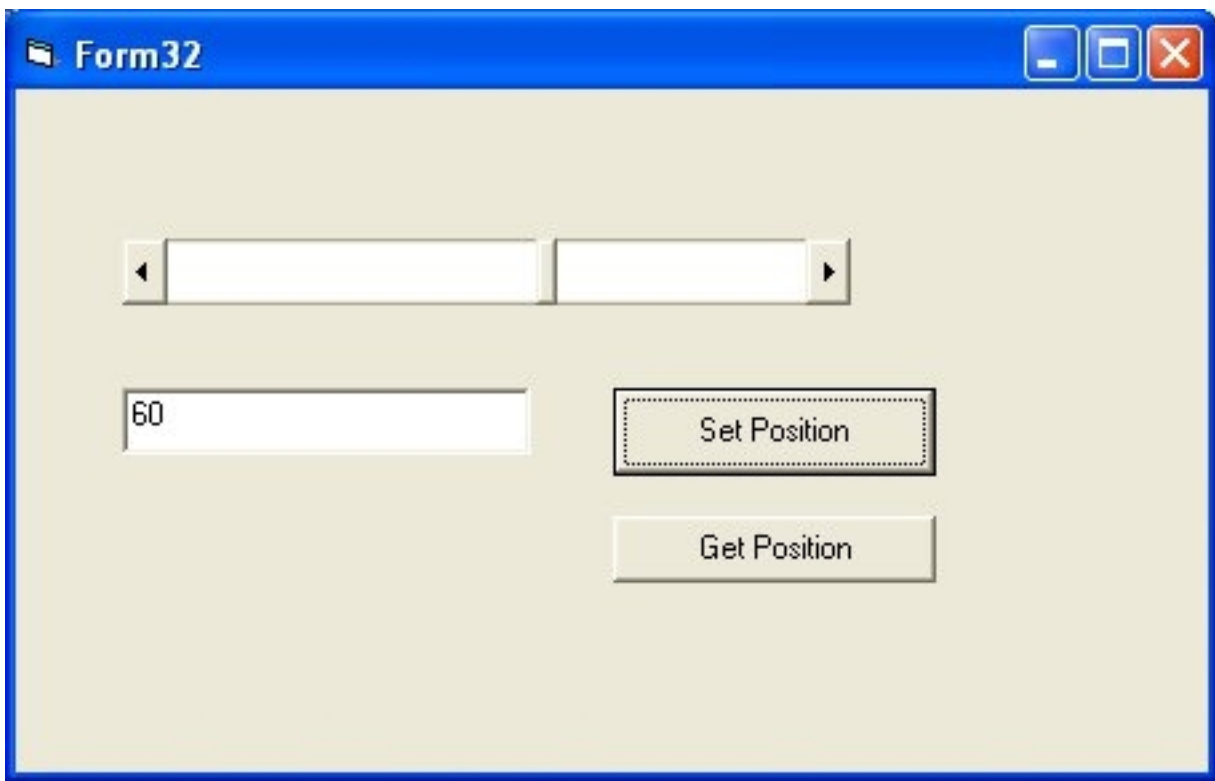
It is used to display a horizontal scroll bar or slider

Main properties are

1. Min
2. Max
3. Value

```
Private Sub Command1_Click()  
HScroll1.Value = Val(Text1.Text)  
End Sub
```

```
Private Sub Command2_Click()  
MsgBox (HScroll1.Value)  
End Sub
```



Vertical Scroll Bar

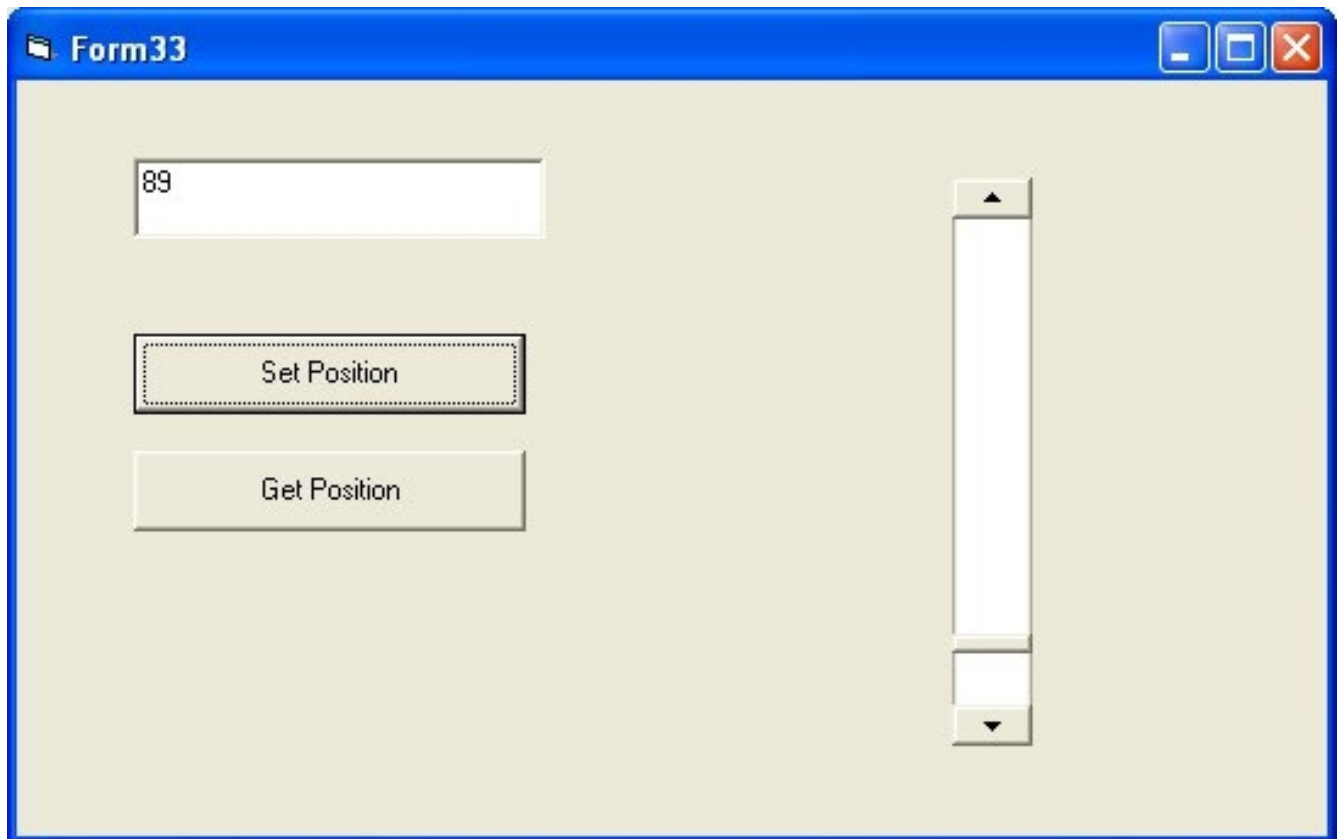
It is used to display a vertical scroll bar or slider

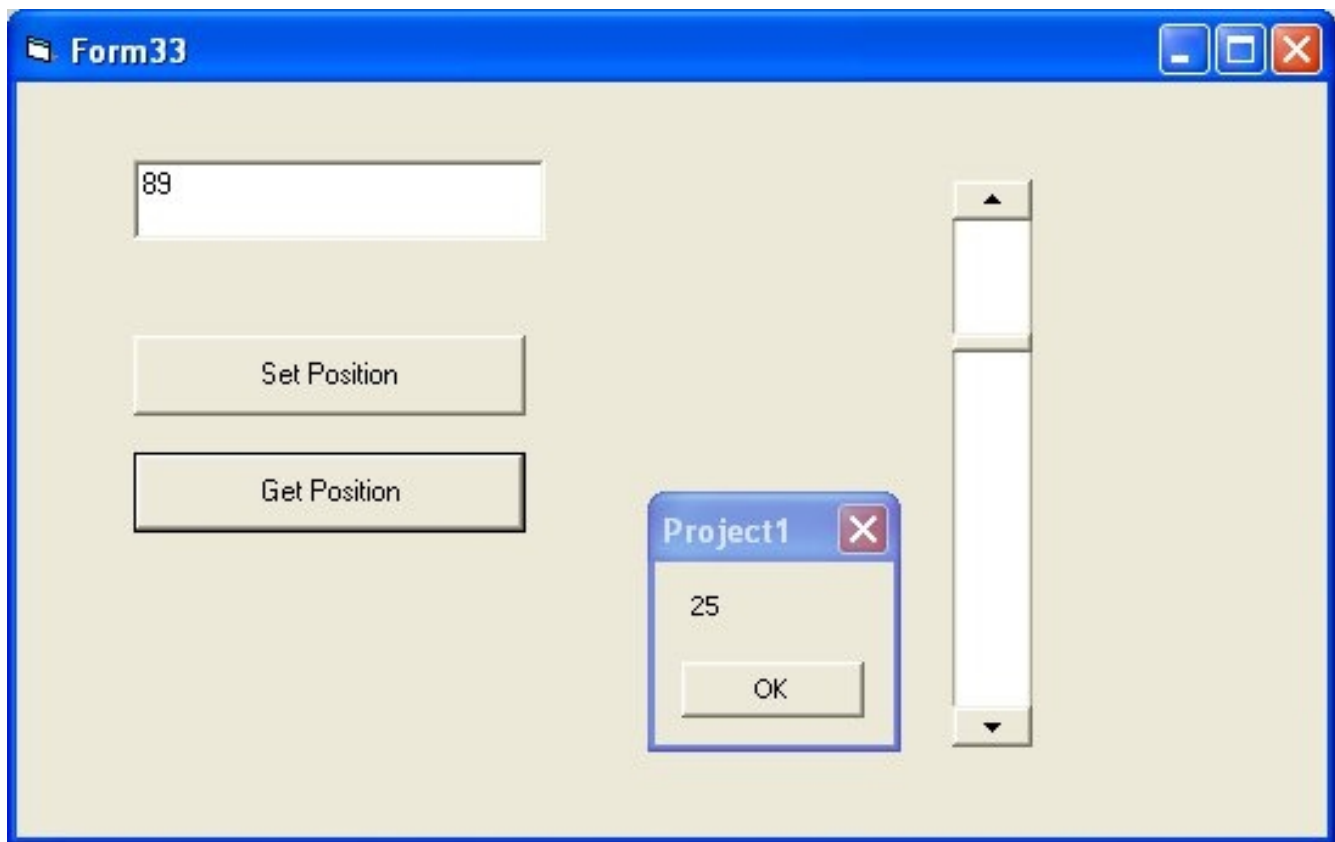
Main properties are

4. Min
5. Max
6. Value

```
Private Sub Command1_Click()  
VScroll1.Value = Val(Text1.Text)  
End Sub
```

```
Private Sub Command2_Click()  
MsgBox (VScroll1.Value)  
End Sub
```





Example of Timer Control

Main property of Timer Control is Interval

Value of Interval is defined in ms so for 5 seconds Value of Interval becomes 5000

Main Event of Timer Control is Timer

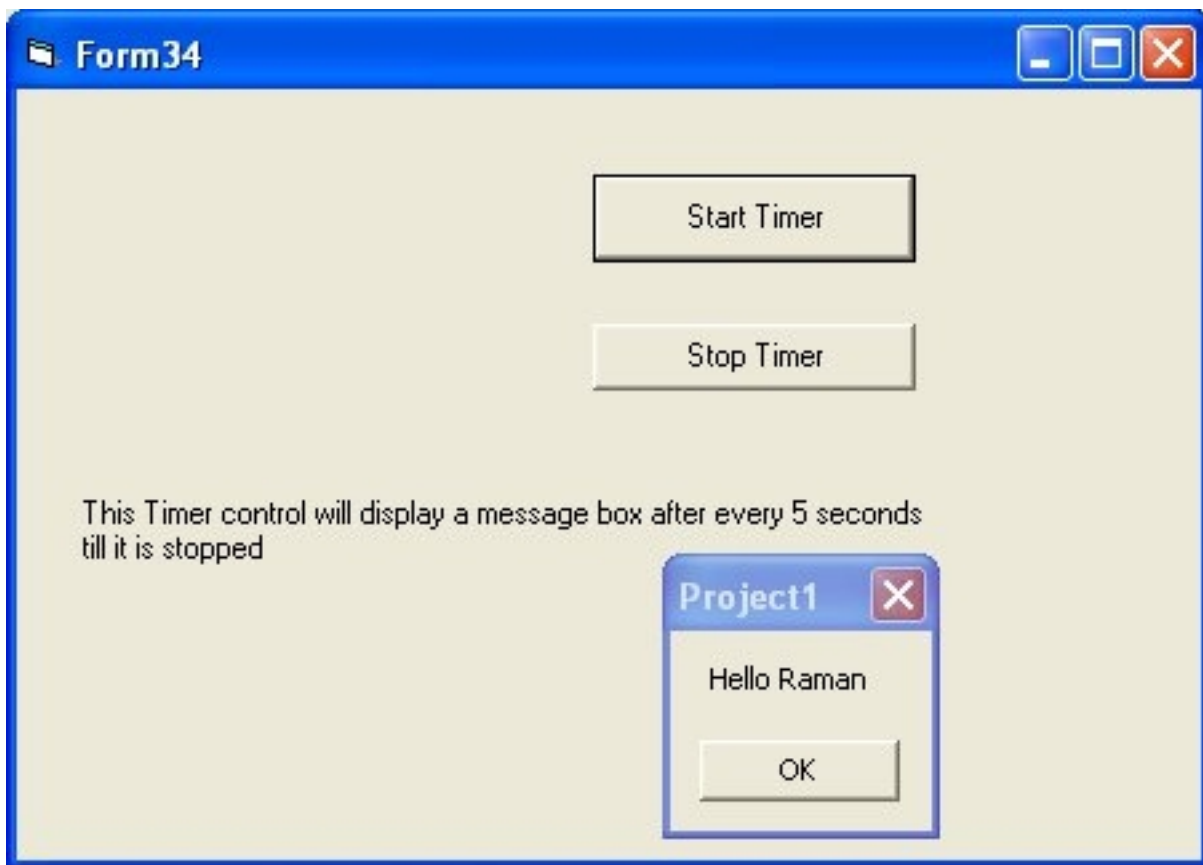
Function associated to Timer Event is Timer1_Timer which is invoked after every milliseconds which are defined in Interval Property

Timer Control is a invisible control

```
Private Sub Command1_Click()  
Timer1.Enabled = True  
End Sub
```

```
Private Sub Command2_Click()  
Timer1.Enabled = False  
End Sub
```

```
Private Sub Timer1_Timer()  
MsgBox ("Hello Raman")  
End Sub
```



Example of Drive List Box Control

Drive List Box displays all the drives (harddisks,cdroms and usb drives) on the computer.

Place it on Form and it will display all drives in computer as a combo box

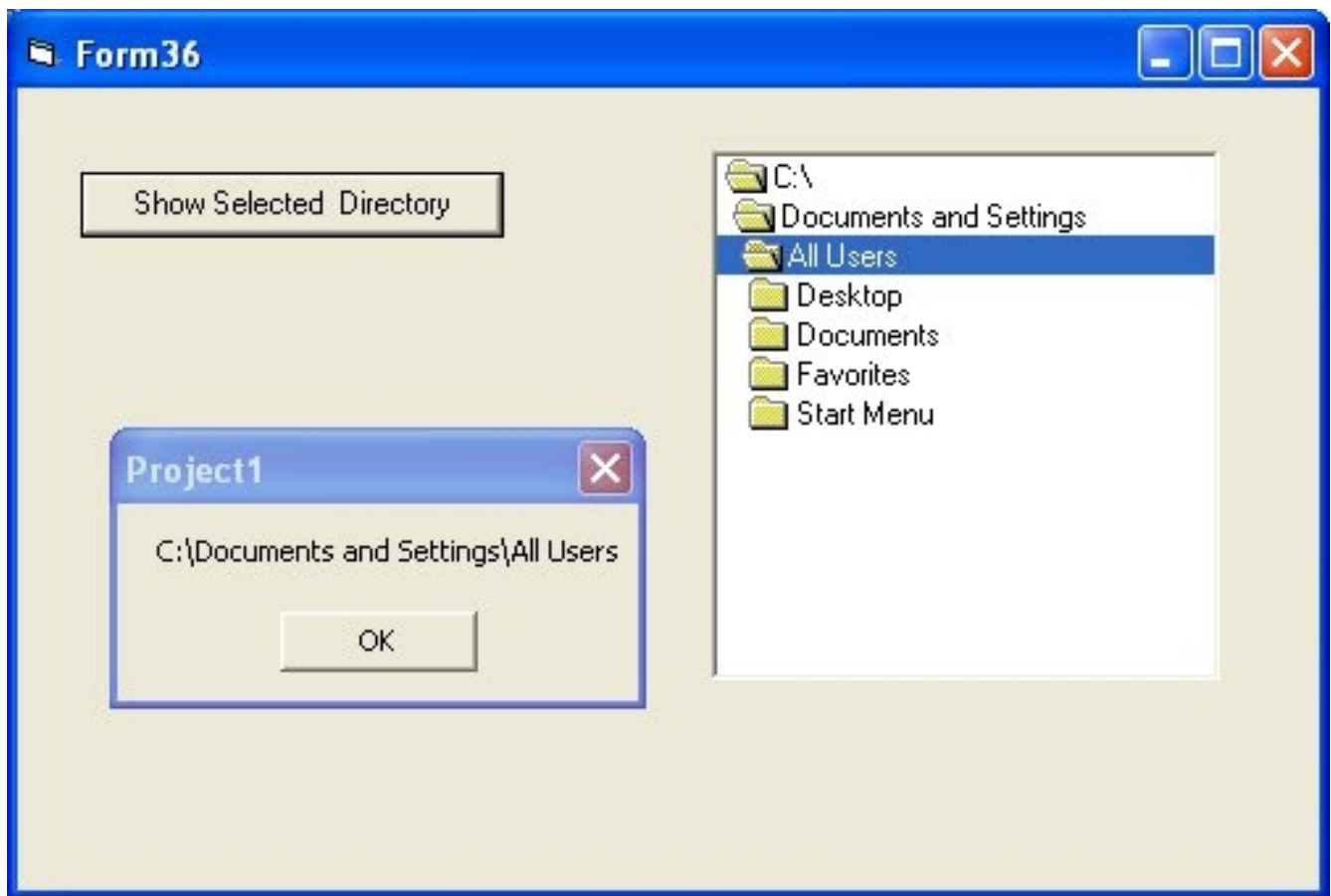
Following code displays current selected drive in Drive List Box Control

```
Private Sub Command1_Click()  
MsgBox (Drive1.Drive)  
End Sub
```

Example of Directory List Box Control

Directory List Box displays all folders in a directory

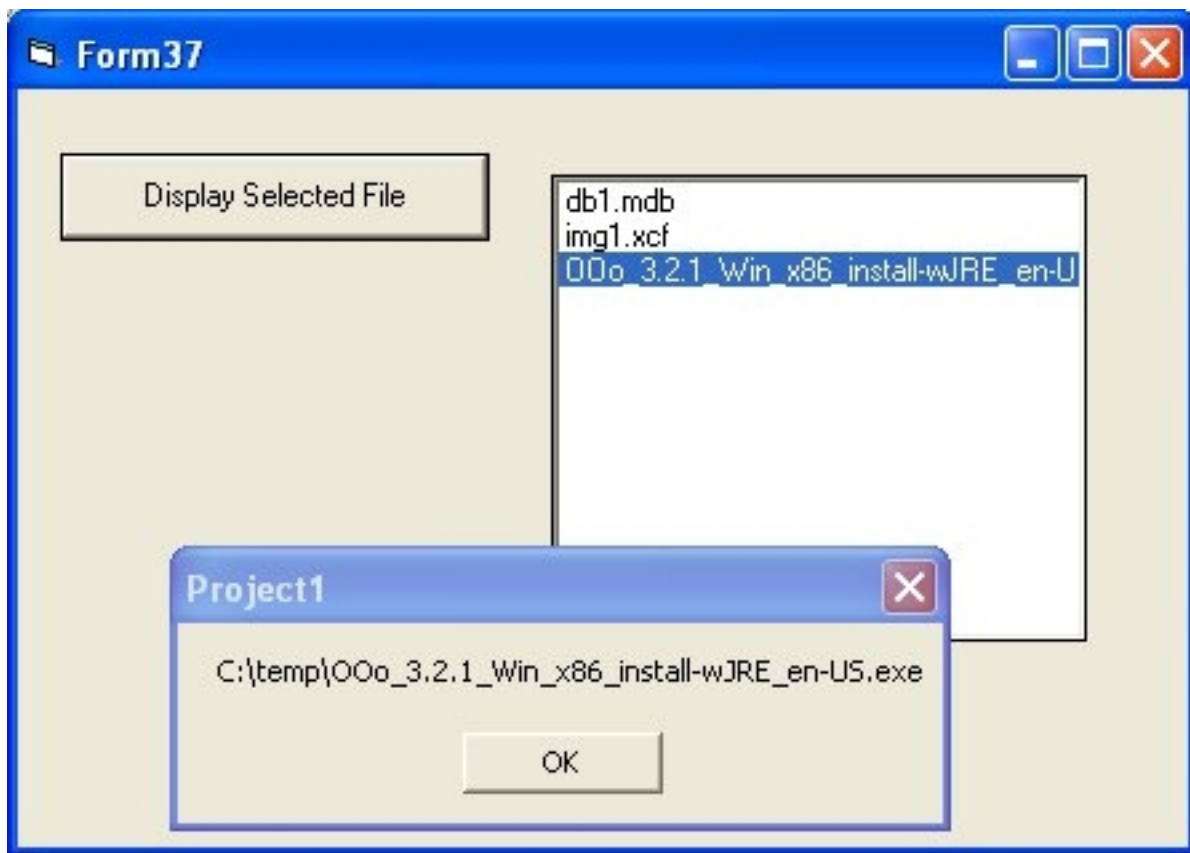
```
Private Sub Command1_Click()  
MsgBox (Dir1.Path)  
End Sub
```



Example of FileListBox Control

File List Box control display lists of all the files in the current working directory

```
Private Sub Command1_Click()  
MsgBox (File1.Path & "\ " & File1.FileName)  
End Sub
```



Example of Shape Control

Shape Control is used to display shapes on form like rectangle, circle, oval, Rounded Rectangle, Rounded Square

Shape Property of Shape Control is used to determine which shape you want to display

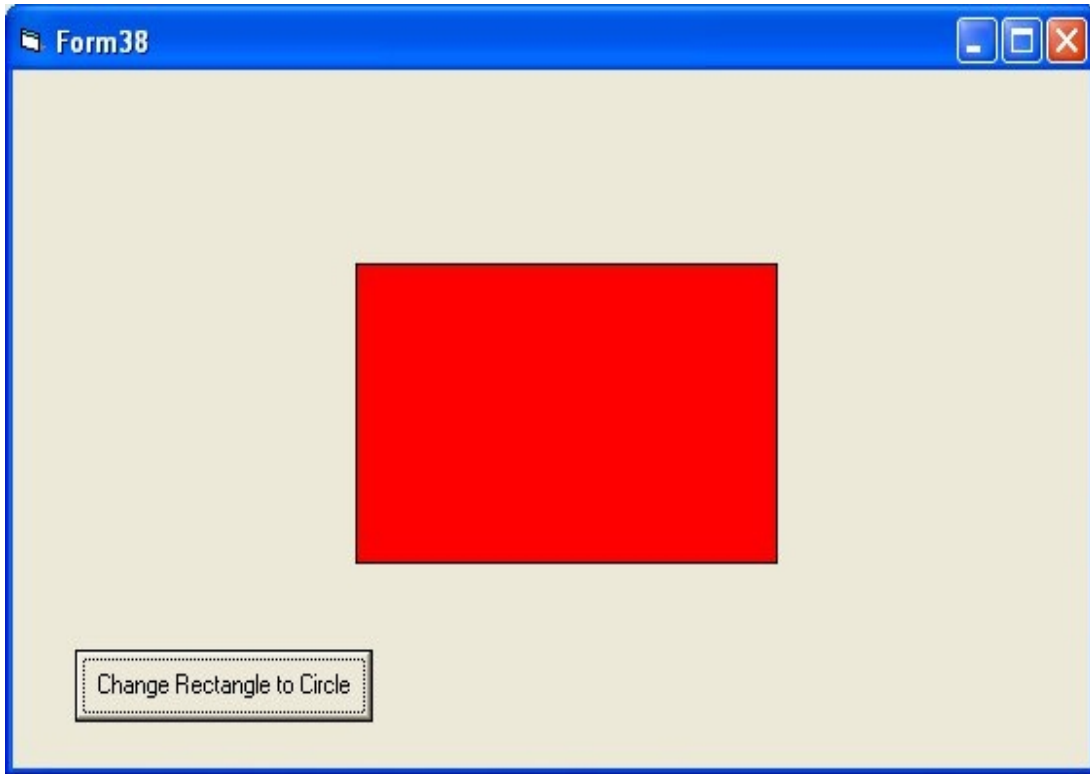
Shape can have these values

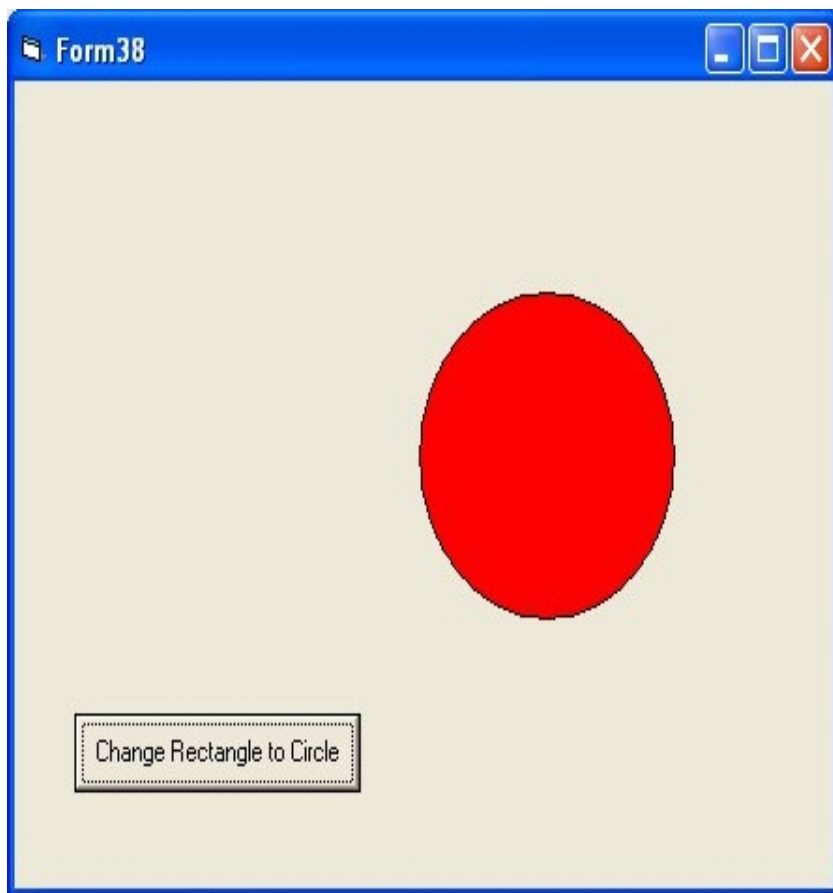
- 0 – Rectangle
- 1 – Square
- 2- Oval
- 3-Circle
- 4-Rounded Rectangle
- 5-Rounded Square

```
Private Sub Command1_Click()  
Shape1.Shape = 3  
Shape1.BackColor = vbRed  
Shape1.Refresh  
End Sub
```

Above code changes shape from rectangle to circle

```
Private Sub Command1_Click()  
Shape1.Shape = 3  
Shape1.BackColor = vbRed  
Shape1.Refresh  
End Sub
```





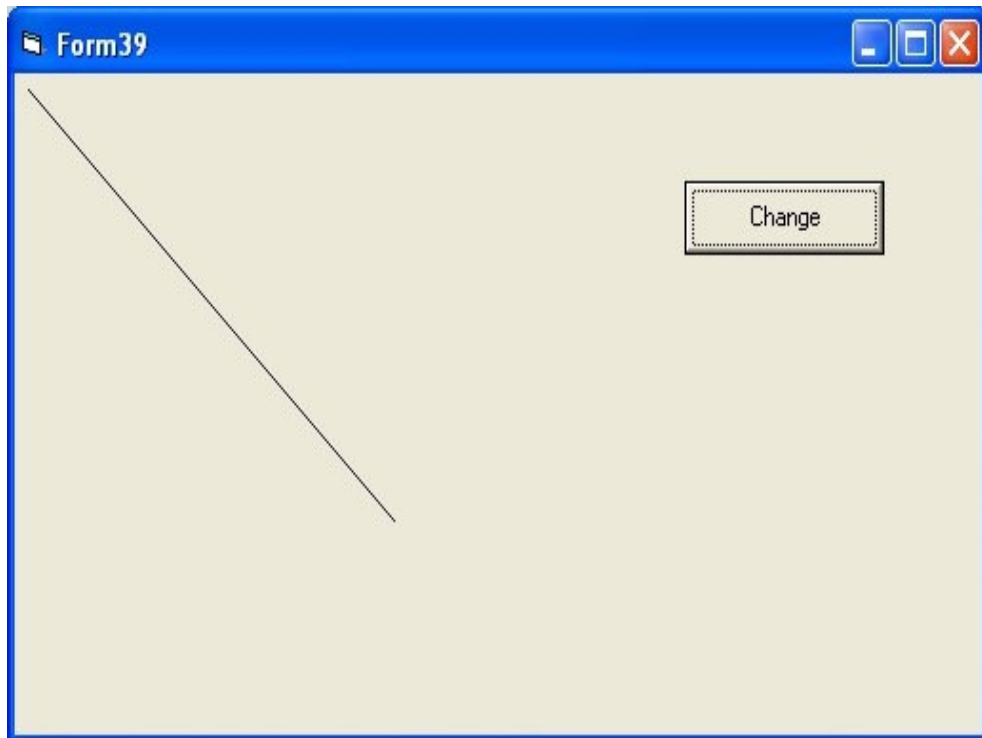
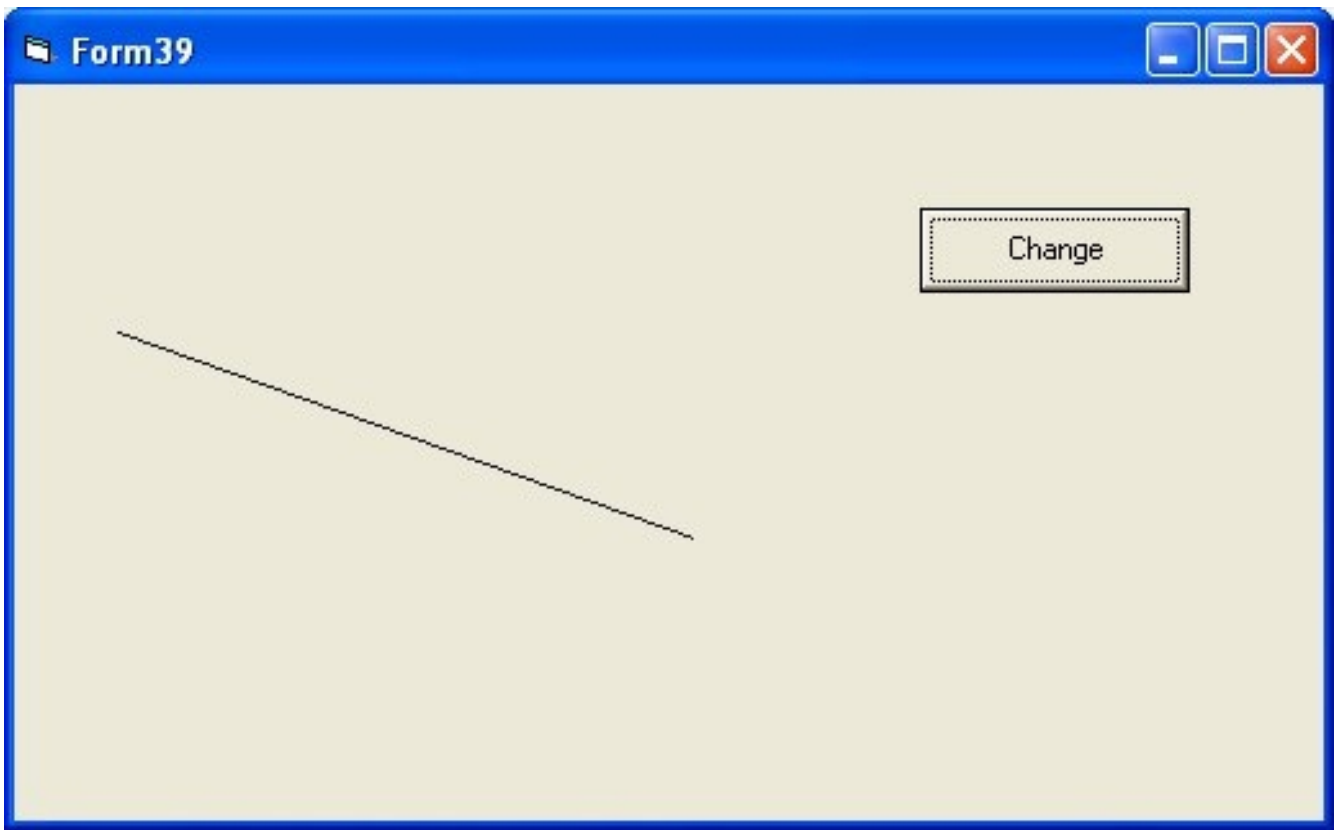
Example of Line Control

Line Control has main properties like

x_1, x_2, y_1, y_2 which determine position of line on form

Following code changes the position of line on the form

```
Private Sub Command1_Click()  
Line1.X1 = 100  
Line1.Y1 = 100  
Line1.X2 = 3000  
Line1.Y2 = 3000  
End Sub
```



Example of Image Control

Image Control is used to display an image in form

Main property is Picture

Picture can be set to an image through LoadPicture Function

In this example

when button1 is clicked image of an apple is shown in Image Box and when button2 is clicked image of a ball is shown in the imagebox

```
Private Sub Command1_Click()
```

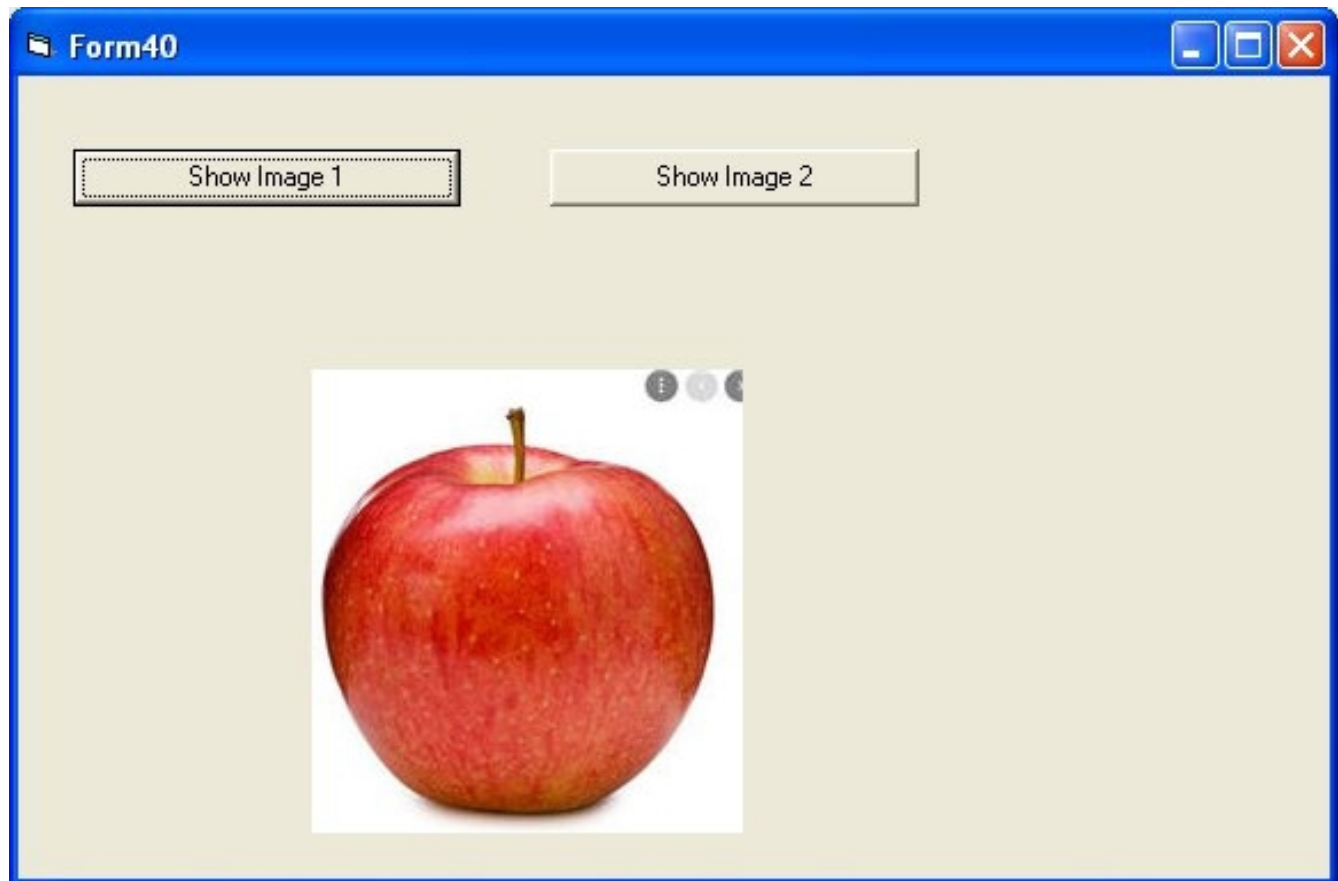
```
Image1.Picture = LoadPicture("c:\temp\aimage.jpg")
```

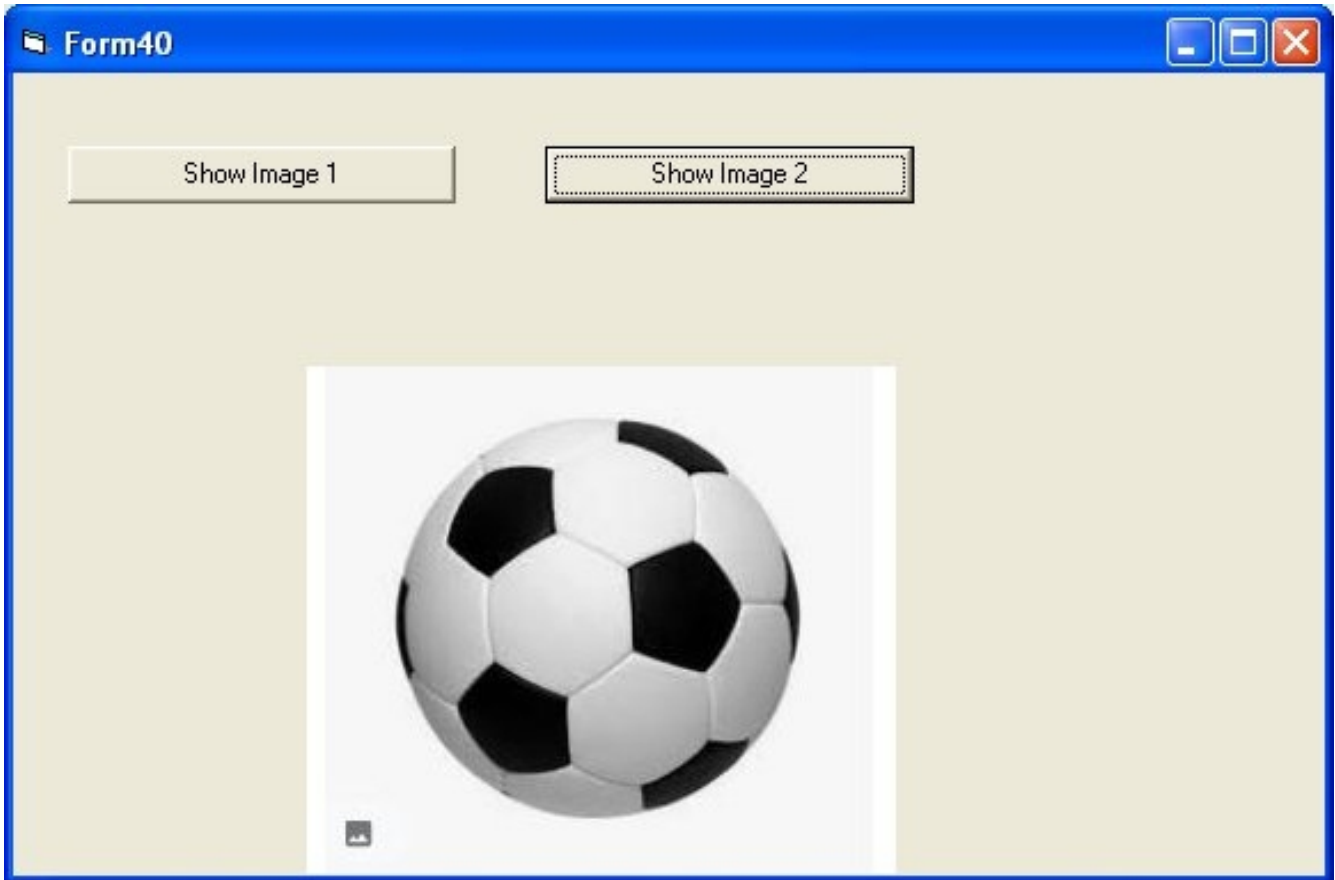
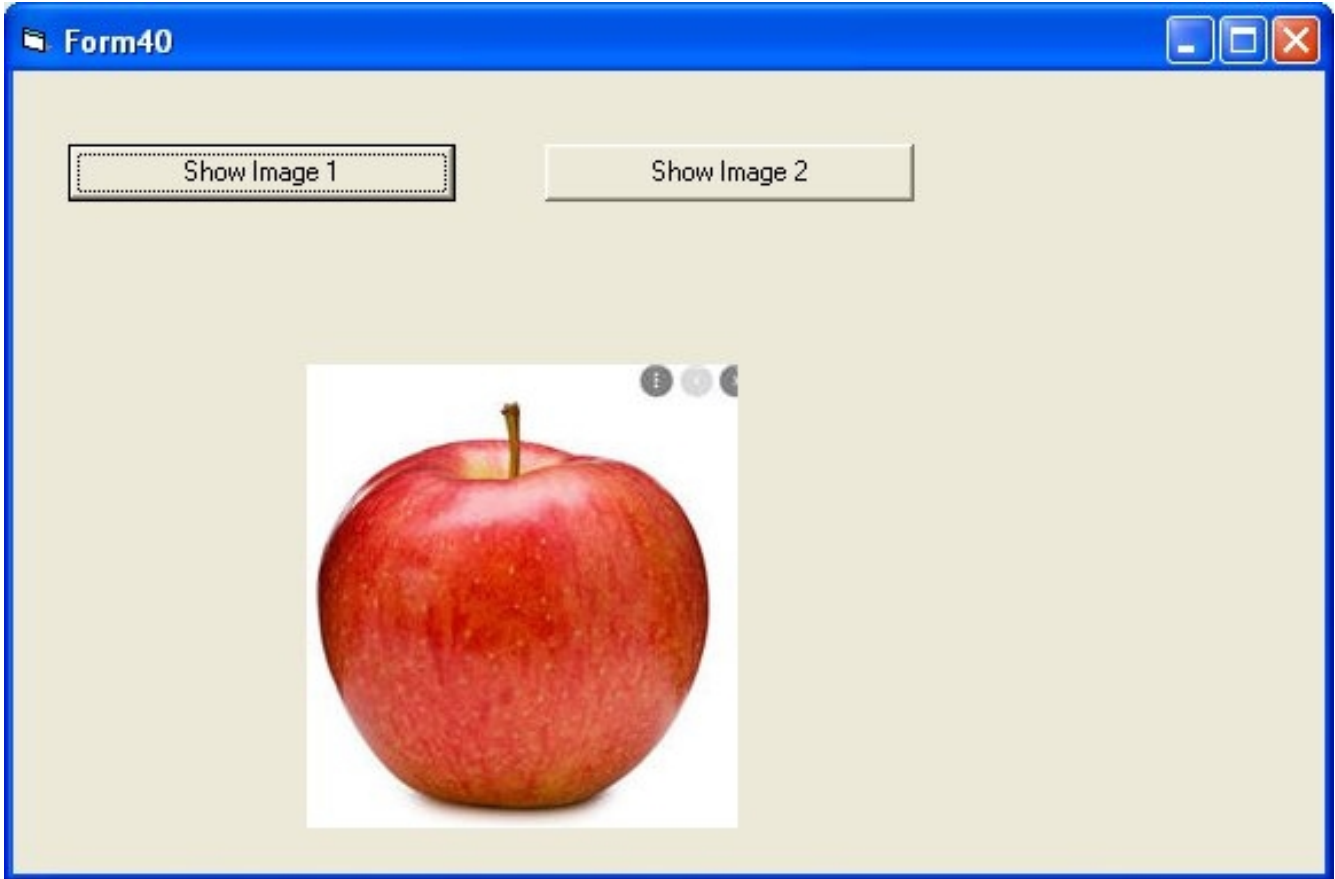
```
End Sub
```

```
Private Sub Command2_Click()
```

```
Image1.Picture = LoadPicture("c:\temp\bimage.jpg")
```

```
End Sub
```

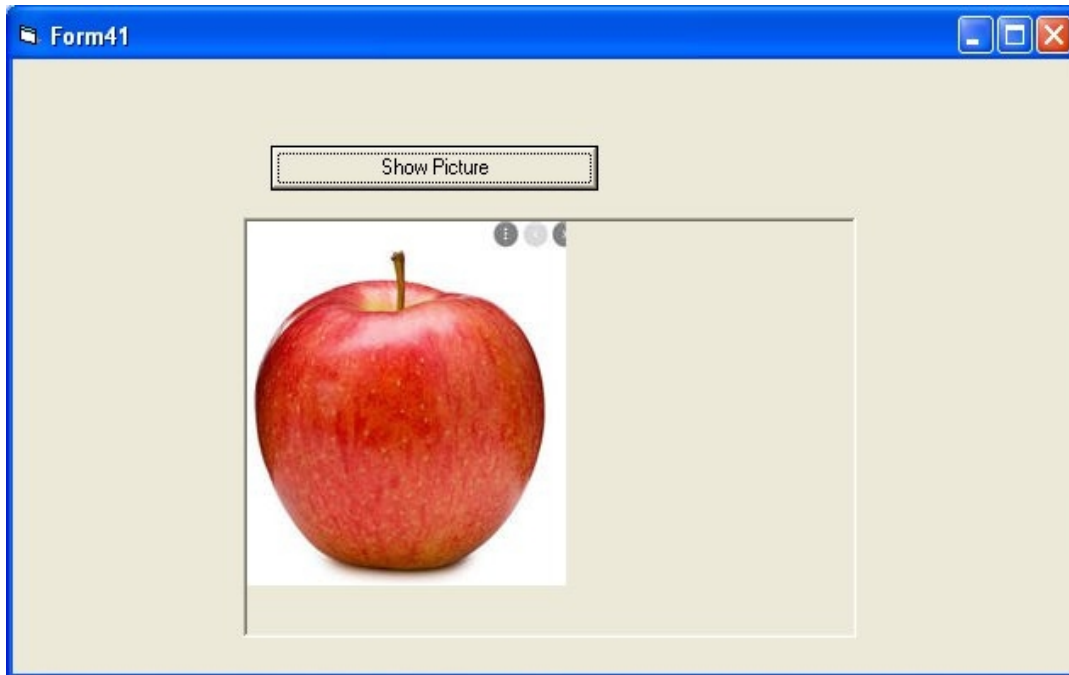




Example of PictureBox Control

PictureBox Control can be used to display a picture in a form

```
Private Sub Command1_Click()  
Picture1.Picture = LoadPicture("c:\temp\aimage.jpg")  
End Sub
```



Connecting to Databases using ADO DB Control

Get Records from a Access Database and display in a ListView

ListView Control is present in Microsoft Windows Common Controls 6.0

```
Private Sub Command1_Click()  
Dim myCon As New ADO DB.Connection  
Dim rs As New ADO DB.Recordset  
Dim lst As ListViewItem  
ListView1.ListItems.Clear  
myCon.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\temp\db1.mdb"  
myCon.Open  
Set rs = New ADO DB.Recordset  
rs.Open "select * from students", myCon, adOpenDynamic, adLockOptimistic  
Dim ListX As ListViewItem  
  
Me.ListView1.ListItems.Clear  
While Not rs.EOF = True  
Set lst = Me.ListView1.ListItems.Add(, , rs.Fields("rollno"))  
lst.SubItems(1) = rs.Fields("name")
```

```
lst.SubItems(2) = rs.Fields("marks")
```

```
rs.MoveNext  
Wend  
Me.ListView1.Refresh
```

```
End Sub
```

```
Private Sub Command1_Click()  
Dim myCon As New ADODB.Connection  
Dim rs As New ADODB.Recordset  
Dim lst As ListItem  
ListView1.ListItems.Clear  
myCon.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\temp\db1.mdb"  
myCon.Open  
Set rs = New ADODB.Recordset  
rs.Open "select * from students", myCon, adOpenDynamic, adLockOptimistic  
Dim ListX As ListItem
```

```
Me.ListView1.ListItems.Clear  
While Not rs.EOF = True  
Set lst = Me.ListView1.ListItems.Add(, rs.Fields("rollno"))  
lst.SubItems(1) = rs.Fields("name")  
lst.SubItems(2) = rs.Fields("marks")
```

```
rs.MoveNext  
Wend  
Me.ListView1.Refresh
```

```
End Sub
```

Add a Record to Database using ADODB

```
Private Sub Command1_Click()  
Dim con As New ADODB.Connection  
Dim cmd As New ADODB.Command  
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\temp\db1.mdb"  
con.Open  
Dim sql As String  
sql = "insert into students values ("  
sql = sql & "" & Text1.Text & ", "  
sql = sql & "" & Text2.Text & ", "  
sql = sql & "" & Text3.Text & ")"  
cmd.ActiveConnection = con  
cmd.CommandText = sql  
cmd.Execute  
MsgBox "Record Inserted to Database"  
End Sub
```

The image shows a Windows application window titled "Form2". Inside the window, there are three text input fields: "Rollno" containing the value "1", "Name" containing "Raman", and "Marks" containing "89". Below these fields is a button labeled "Add Record". Overlaid on the bottom right of the "Form2" window is a smaller dialog box titled "Project1". This dialog box contains the text "Record Inserted to Database" and an "OK" button at the bottom.

Update Record

```

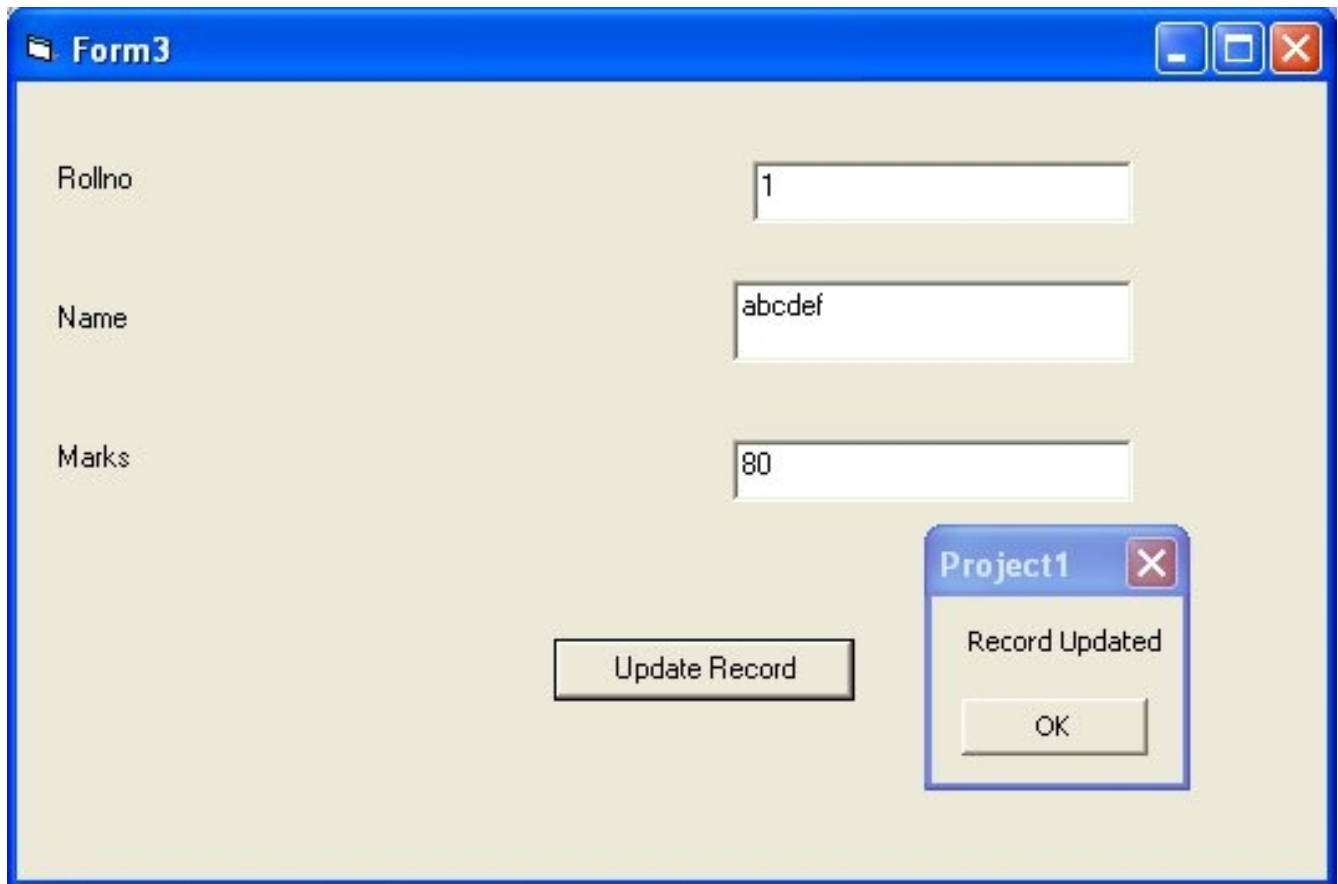
Private Sub Command1_Click()
Dim con As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim cmd As New ADODB.Command
Dim count As Integer
count = 0
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\temp\db1.mdb"
con.Open
Set rs = New ADODB.Recordset
rs.Open "select * from students where rollno=" & Text1.Text, con, adOpenDynamic, adLockOptimistic
While Not rs.EOF = True
count = count + 1
rs.MoveNext
Wend
If count > 0 Then
Dim sql As String
sql = "update students set name=" & Text2.Text & ",marks=" & Text3.Text & " where rollno=" &

```

```

Text1.Text
cmd.ActiveConnection = con
cmd.CommandText = sql
cmd.Execute
Else
MsgBox ("No Such Record Exists")
End If
MsgBox ("Record Updated")
End Sub

```



Delete a Record

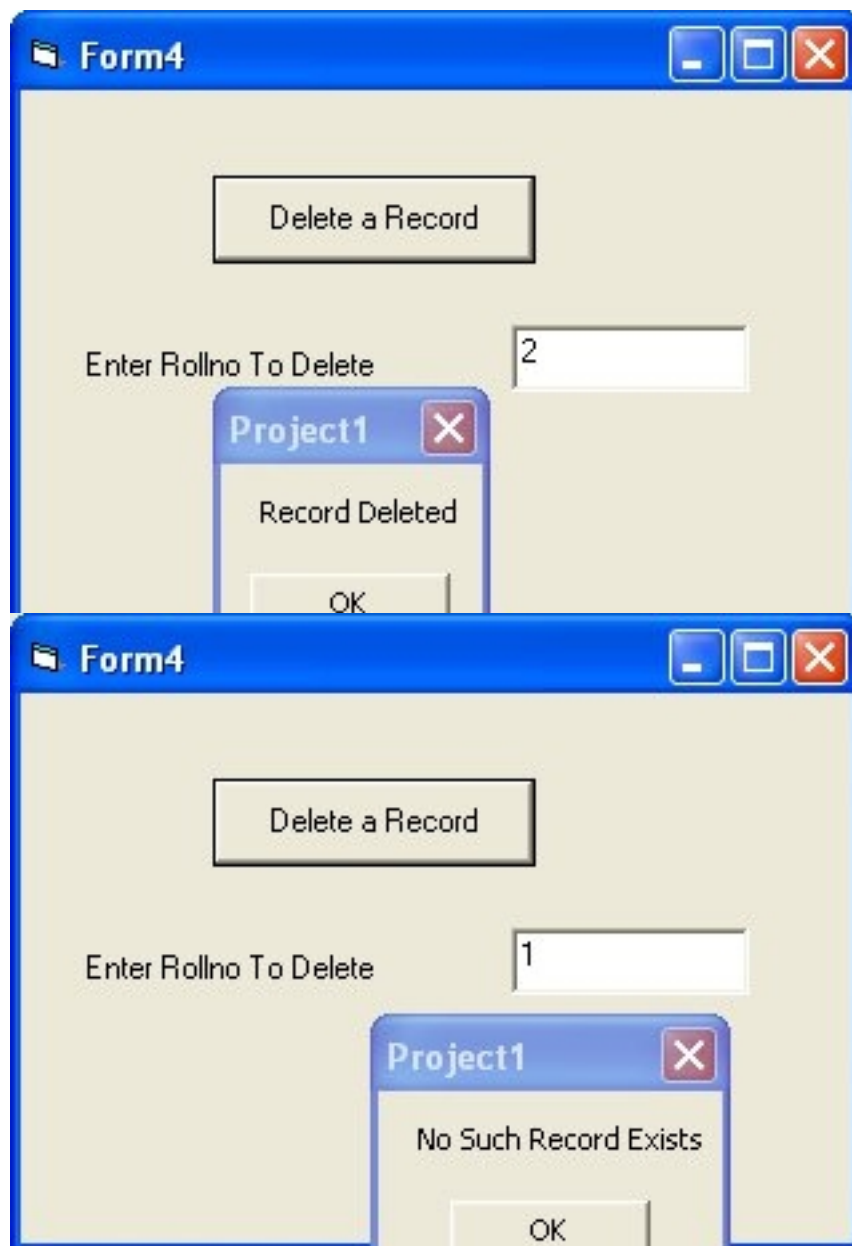
```

Private Sub Command1_Click()
Dim con As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim cmd As New ADODB.Command
Dim count As Integer
count = 0
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\temp\db1.mdb"
con.Open
Set rs = New ADODB.Recordset
rs.Open "select * from students where rollno=" & Text1.Text, con, adOpenDynamic, adLockOptimistic
While Not rs.EOF = True

```



```
count = count + 1
rs.MoveNext
Wend
If count > 0 Then
Dim sql As String
sql = "delete from students where rollno=" & Text1.Text
cmd.ActiveConnection = con
cmd.CommandText = sql
cmd.Execute
MsgBox ("Record Deleted")
Else
MsgBox ("No Such Record Exists")
End If
End Sub
```



Example of TreeView Control

In the example below we read student records from access database and display records in a treeview click

TreeView Control is present in Microsoft Windows Common Controls 6.0

```
Private Sub Command1_Click()
Dim Node1, Node2, Node3 As Node
Set Node1 = TreeView2.Nodes.Add
TreeView2.Nodes(1).Text = "Node 1"
TreeView2.Nodes(1).Key = "Node 1"
Set Node2 = TreeView2.Nodes.Add("Node 1", tvwChild, "Node 2")
TreeView2.Nodes(2).Text = "Node 2"
TreeView2.Nodes(2).Key = "Node 2"
Set Node3 = TreeView2.Nodes.Add("Node 1", tvwChild, "Node 3")
TreeView2.Nodes(3).Text = "Node 3"
TreeView2.Nodes(3).Key = "Node 3"
End Sub

Private Sub Command2_Click()
TreeView1.Nodes.Clear
Dim myCon As New ADODB.Connection
Dim rs As New ADODB.Recordset
myCon.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\temp\db1.mdb"
myCon.Open
rs.Open "select * from students", myCon, adOpenDynamic, adLockOptimistic
Dim rollno As Integer
Dim cname, NameKey, name, rollnokey, markskey As String
Dim marks As Integer

Dim NodeRoot, noderollno, nodename, nodemarks As Node
Dim i As Integer
i = 0
    Set NodeRoot = TreeView1.Nodes.Add(, "RootDB", "Students", 0)
    While Not rs.EOF = True

rollno = rs.Fields("rollno").Value
name = rs.Fields("name").Value
marks = rs.Fields("marks").Value
rollnokey = "R" & Str(i)
NameKey = "N" & name
markskey = "M" & Str(marks)
Set noderollno = TreeView1.Nodes.Add("RootDB", tvwChild, rollnokey, rollno, 0)
Set nodename = TreeView1.Nodes.Add(rollnokey, tvwChild, NameKey, name, 0)
Set nodemarks = TreeView1.Nodes.Add(rollnokey, tvwChild, markskey, marks, 0)
i = i + 1
rs.MoveNext
Wend
```

End Sub

