

CONTENTS

S.NO	DATE	EXPERIMENT	PAGE NO	SIGNATURE
1	16-12-2021	Arithmetic Operations		
2	23-12-2021	Working with Intrinsic Controls		
3	27-12-2021	Working with Multiple Forms		
4	05-01-2022	Working with Dialog Boxes		
5	18-02-2022	Working with Menus		
6	24-02-2022	Working with Data Controls		
7	03-03-2022	Application with error Trapping		
8	10-03-2022	Drag and Drop Applications		
9	17-03-2022	Application using File system objects		
10	24-03-2022	Creating ActiveX controls		

1- ARITHMETIC OPERATION

AIM:

To write a program in visual basic to perform arithmetic operation by using command button and text box.

DESCRIPTION:

1. Open a new visual basic project and save it with .frm and .vbp extension.
2. Design a form by using text boxes and command buttons as needed.
3. Write the coding for the appropriate controls in coding window.
4. Execute the program by pressing F5 or Run button.
5. Give the input and verify the result.

REQUIREMENTS:

1. Text box-3
2. Command button-5
3. Label box-3

Properties to change

TextBox-1

Name: txt_1

Text: “ “

TextBox-2

Name: txt_2

Text: “ “

TextBox-3

Name: txt_result

Text: “ “

Enabled: false

Button-1

Name: cmd_add

Text: &Add

Button-2

Name: cmd_sub

Text: &Sub

Button-3

Name: cmd_mul

Text: &Mul

Button-4

Name: cmd_div

Text: &Div

Button-5

Name: cmd_mod

Text: M&od

CODING:

Add button:

```
Private Sub cmd_add_Click()  
Text3.Text = Val(Text1.Text)+Val(Text2.Text)  
End Sub
```

Subtraction button:

```
Private Sub cmd_sub_Click()  
Text3.Text = Val(Text1.Text) - Val(Text2.Text)  
End Sub
```

Multiplication button:

```
Private Sub cmd_mul_Click()  
Text3.Text = Val(Text1.Text) * Val(Text2.Text)  
End Sub
```

Division button:

```
Private Sub cmd_div_Click()
```

```
Text3.Text = Val(Text1.Text) / Val(Text2.Text)
```

```
End Sub
```

Modulus button:

```
Private Sub cmd_mod_Click()
```

```
Text3.Text = Val(Text1.Text) mod Val(Text2.Text)
```

```
End Sub
```

OUTPUT:

RESULT:

Thus the application have been successfully created and executed.

2- WORKING WITH INTRINSIC CONTROLS

AIM:

To create a student application in visual basic using intrinsic controls

DESCRIPTION:

1. Open a new visual basic project and save it with lab2.frm and lab2.vbp extension
2. Design a form by using label, textboxes, combo box, checkbox, option box and command button.
3. Write the coding for appropriate controls in coding window.
4. Execute the program by pressing F5 or Run button.
5. Give the input and verify the result.

REQUIREMENTS:

1. Textbox- 2
2. Labels-3
3. Checkbox –3
4. Option button-3
5. Combobox -1
6. Command button-2

PROPERTIES TO CHANGE:

TextBox-1

Name: txt_name

Text: “ ”

TextBox-2

Name: txt_class

Text: “ ”

Checkbox-1

Name: Chk_english

Checkbox-2

Name: Chk_tamil

Checkbox-3

Name: Chk_hindi

Optionbutton-1

Name: opt_male

Optionbutton-2

Name: opt_female

Optionbutton-3

Name: opt_trans

Combobox-1

Name: cbo_college

Command button-1

Name: cmd_submit

Command button-2

Name: cmd_clear

CODING:

```
Dim gender As String
```

```
Dim lan_knw As String
```

```
Private Sub cmd_submit_Click()
```

```
Me.Cls
```

```
lan_knw = " "
```

```
Print
```

```
Print
```

```
Print Tab(65);
```

```
Me.FontBold = True
```

```
Print " Name : " & txt_name.Text
```

```
Print
```

```
Print
```

```
Print Tab(65); " Class: " & txt_Class.Text
```

```
If Opt_male.Value = True Then
```

```
gender = "Male"
```

```
ElseIf Opt_female.Value = True Then
    gender = "Female"
Else
    gender = "Trigender"
End If

Print

Print

Print Tab(65); " Gender : " & gender

If Chk_english.Value = 1 Then
    lan_knw = "English"
End If

If Chk_tamil.Value = 1 Then
    lan_knw = lan_knw + " Tamil"
End If

If Chk_hindi.Value = 1 Then
    lan_knw = lan_knw + " Hindi"
End If

Print

Print

Print Tab(65); " Languages Known :" & lan_knw

Print

Print

Print Tab(65); " College :" & Cbo_college.Text

End Sub
```

```
Private Sub Form_Load()  
  
Cbo_college.AddItem "Adhiparasakthi College of Arts and Science", 0  
  
Cbo_college.AddItem "Adhiparasakthi College of Engineering", 1  
  
Cbo_college.AddItem "Adhiparasakthi Agriculture College", 2  
  
Cbo_college.AddItem "Adhiparasakthi College of Education", 3  
  
End Sub
```

OUTPUT

Working with Intrinsic Controls

Name : Dr.P.V.Praveen

Class: II BCA

Gender : Male

Languages Known :English Tamil

College :Adhiparasakthi College of Arts and Science

Name:

Class:

Gender: ☒ Male ☐ Female ☐ Transgender

Languages Known: ☒ Tamil ☒ English ☐ Hindi

College:

RESULT:

Thus, the application have been successfully created and executed.

3- WORKING WITH MULTIPLE FORMS

DATE:27-12-2021

AIM:

To implement various number series application in individual forms.

DESCRIPTION:

1. Open a new visual basic project and save it with .frm and .vbp extension.
2. Design a main form which has a link to display n values, odd numbers and even numbers
3. Write the coding for the appropriate forms in coding window.
4. Execute the program by pressing F5 or Run button.
5. Give the input and verify the result.

REQUIREMENTS:

1. Forms-4
2. Command button-3
3. Option Button-3

PROPERTIES TO CHANGE:

Control	Name	Caption
Form	frm_main	"Working with Multiple Forms"
Form	frm_nvalues	" Display N Values"
Form	frm_odd	"Display Odd Values "
Form	Frm_even	"Display Even Values "
Button	Cmd_ok	"Generate"
Button	Cmd_clear	"Clear"
Button	Cmd_close	"Close"
Option button	Opt_nvalue	"Display N numbers"
Option button	Opt_odd	"Display Odd Numbers"
Option button	Opt_even	"Display Even Numbers"

CODING:

FORM NAME: frm_main

```
Private Sub cmd_clear_Click()  
    opt_nvalue.Value = False  
    opt_odd.Value = False  
    opt_even.Value = False  
End Sub
```

Private Sub cmd_close_Click()

Unload Me

End Sub

Private Sub Cmd_ok_Click()

If opt_nvalue.Value = True Then

frm_nvalues.Show

ElseIf opt_odd.Value = True Then

frm_odd.Show

ElseIf opt_even.Value = True Then

frm_even.Show

End If

End Sub

FORM NAME: frm_nvalues

Private Sub Form_Load()

Dim n As Integer

n = Val(InputBox("Enter the n value"))

For i = 1 To 10

Print

Next

Me.FontSize = 16

Me.FontBold = True

Me.ForeColor = vbBlue

Print Tab(20); "Display N Numbers"

Me.FontSize = 14

Me.ForeColor = vbBlack

For i = 1 To n

Print Tab(25); i

Next

End Sub

FORM NAME: frm_odd

Private Sub Form_Load()

Dim n As Integer

n = Val(InputBox("Enter the n value"))

For i = 1 To 10

Print

Next

Me.FontSize = 16

Me.FontBold = True

Me.ForeColor = vbBlue

Print Tab(20); "Display Odd Numbers"

Me.FontSize = 14

Me.ForeColor = vbBlack

Me.FontBold = True

For i = 0 To n Step 2

Print Tab(25); i

Next

End Sub

FORM NAME: frm_even

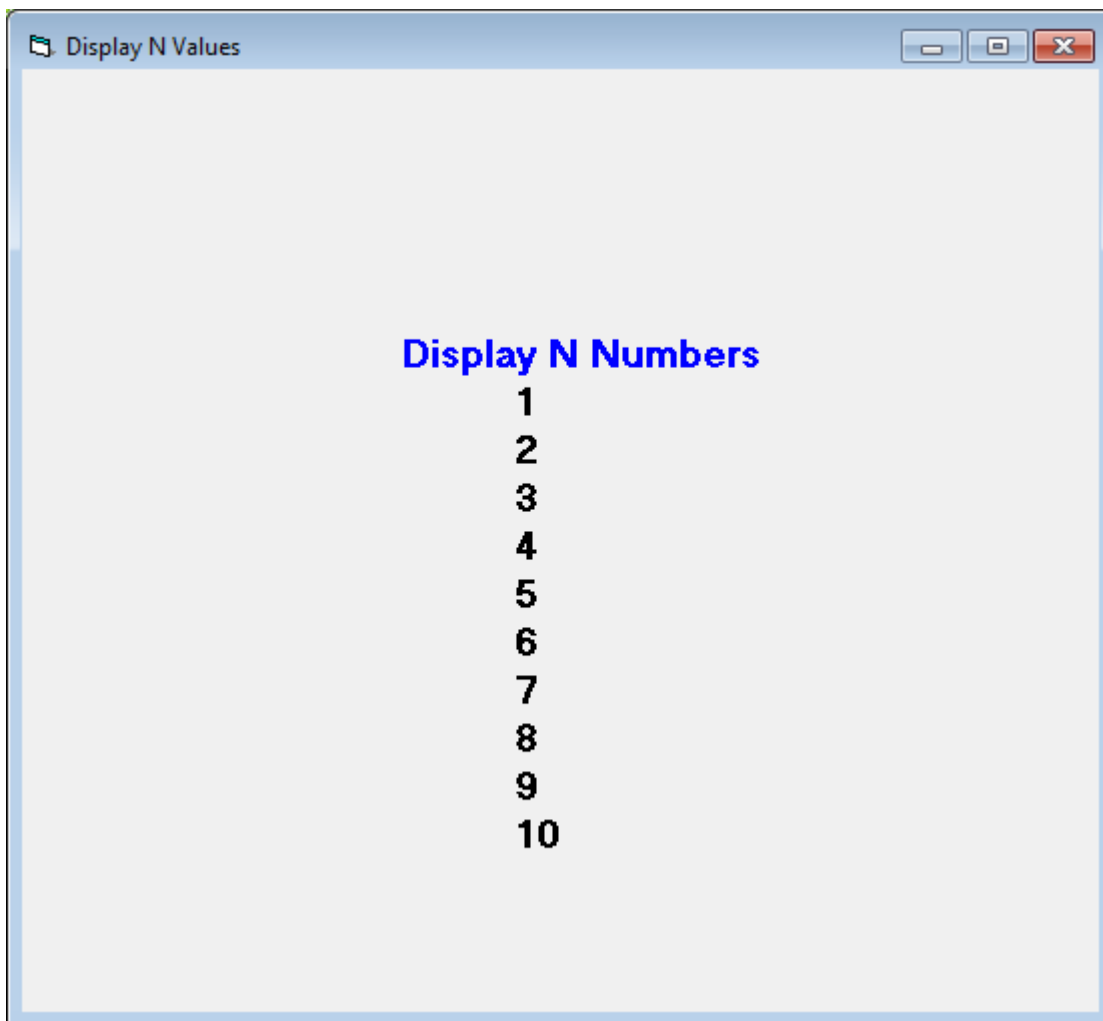
Private Sub Form_Load()

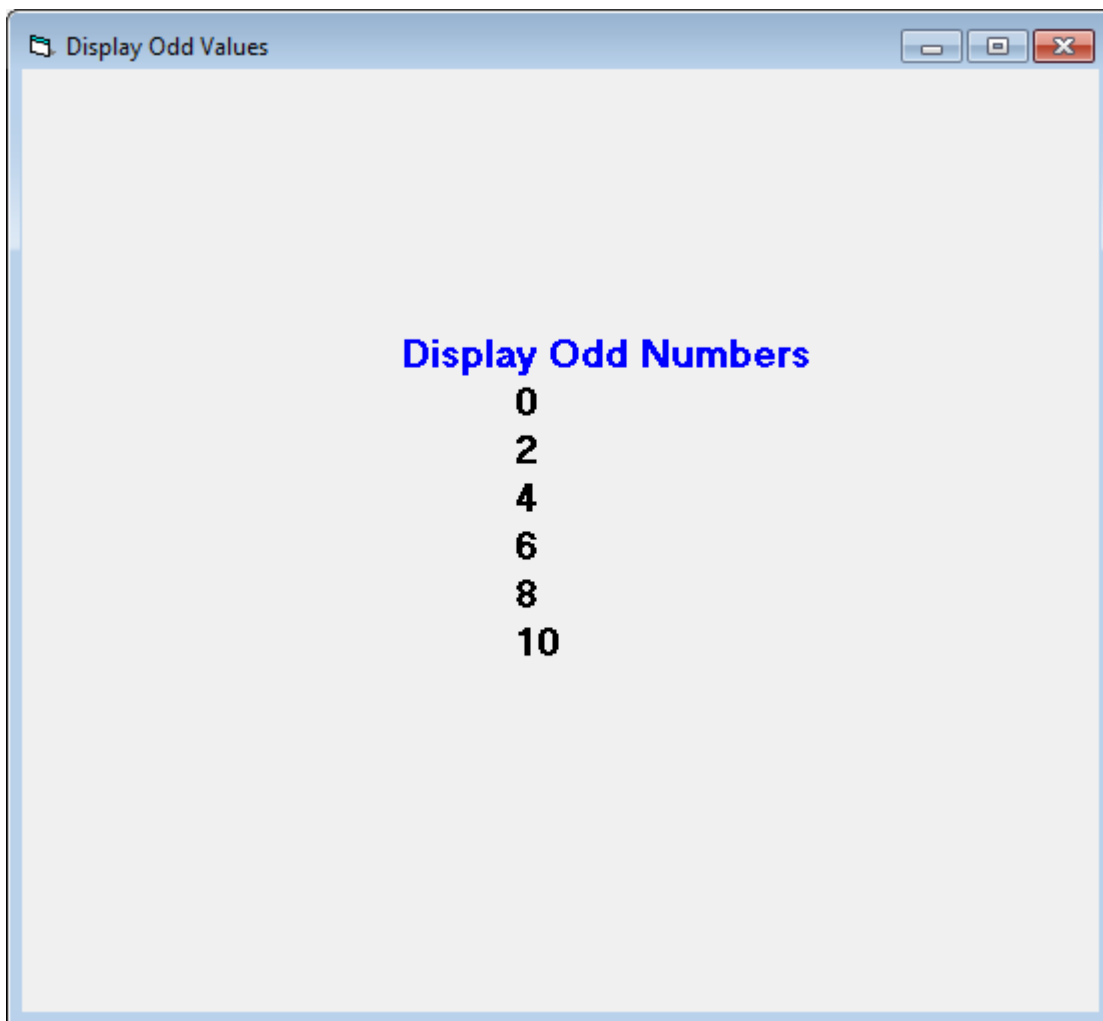
```
Dim n As Integer
n = Val(InputBox("Enter the n value"))
For i = 1 To 10
    Print
Next
Me.FontSize = 16
Me.FontBold = True
Me.ForeColor = vbBlue
Print Tab(20); "Display Even Numbers"
Me.FontSize = 14
Me.ForeColor = vbBlack
Me.FontBold = True
For i = 1 To n Step 2
    Print Tab(25); i
Next
```

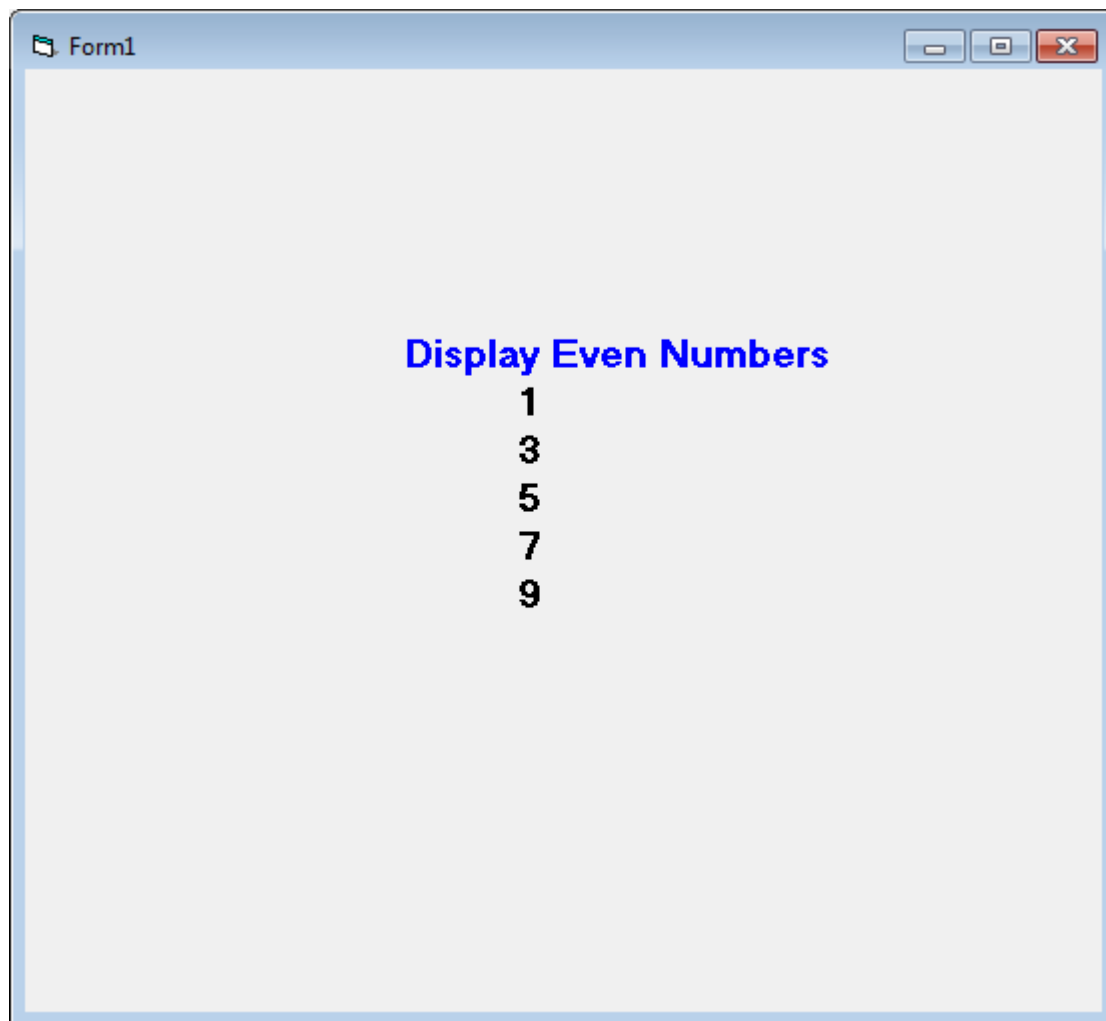
End Sub

OUTPUT:

The screenshot shows a Windows application window titled "Working with Multiple Forms". Inside the window is a form titled "Number Series". The form contains three radio buttons: "Display N Numbers" (which is selected), "Display Odd Numbers", and "Display Even Numbers". Below the form, there are three buttons: "Generate", "Clear", and "Close".







RESULT:

Thus, the application has been successfully executed.

4- WORKING WITH DIALOG BOXES

DATE:05-01-2022

AIM:

To implement the various operations on common dialog boxes.

DESCRIPTION:

1. Open a new visual basic project and save it with frm_dialog.frm and lab4.vbp extension.
2. Insert the common dialog box from add components tab.
3. Insert the Image, Text Box and command buttons in frm_dialog.
4. Write the coding for the appropriate dialog box in coding window.
5. Execute the program by pressing F5 or Run button.
6. Give the input and verify the result.

REQUIREMENTS:

1. Forms-1
2. Command button-4
3. Image-1
4. TextBox-1

PROPERTIES TO CHANGE:

Control	Name	Caption
Form	frm_dialog	"Dialog Box"
Button	cmd_open	"Open"
Button	cmd_close	"Close"
Button	cmd_font	"Font"
Button	cmd_color	"Color"
Image Control	Image1	
TextBox	txt_data	

CODING:

FORM NAME: frm_dialog

Private Sub cmd_open_Click()

```
Dim filename As String
On Error GoTo doNothing
c_dialog.Filter = "JPEG Files (*.JPG) |*.JPG"
Image1.Visible = True
c_dialog.ShowOpen
filename = c_dialog.filename
Image1.Picture = LoadPicture(filename)
```

```

doNothing:
End Sub

Private Sub cmd_close_Click()
    Image1.Picture = Nothing
End Sub

```

```

Private Sub cmd_Color_Click()
    On Error GoTo doNothing
    c_dialog.ShowColor
    txt_data.ForeColor = c_dialog.Color
    doNothing:
End Sub

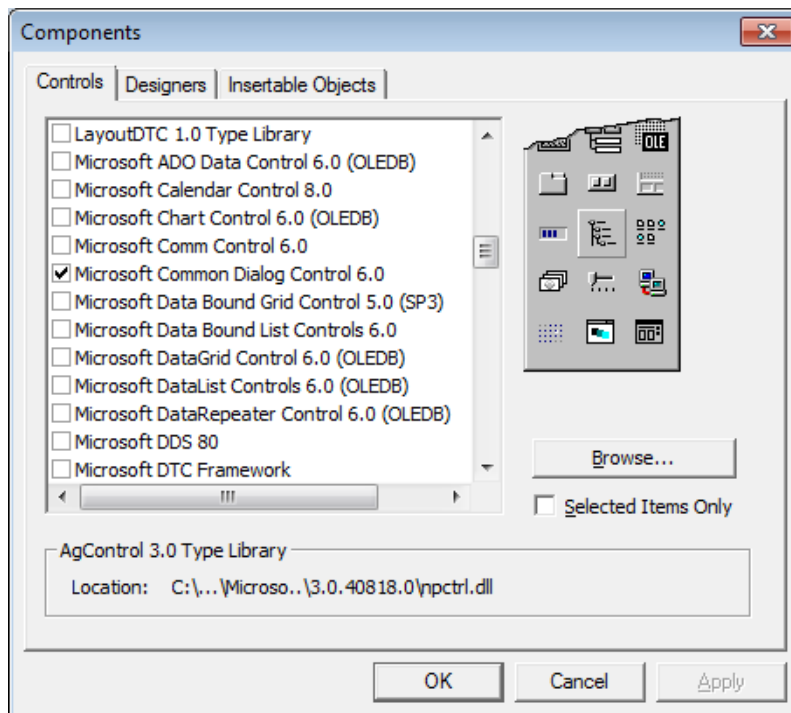
```

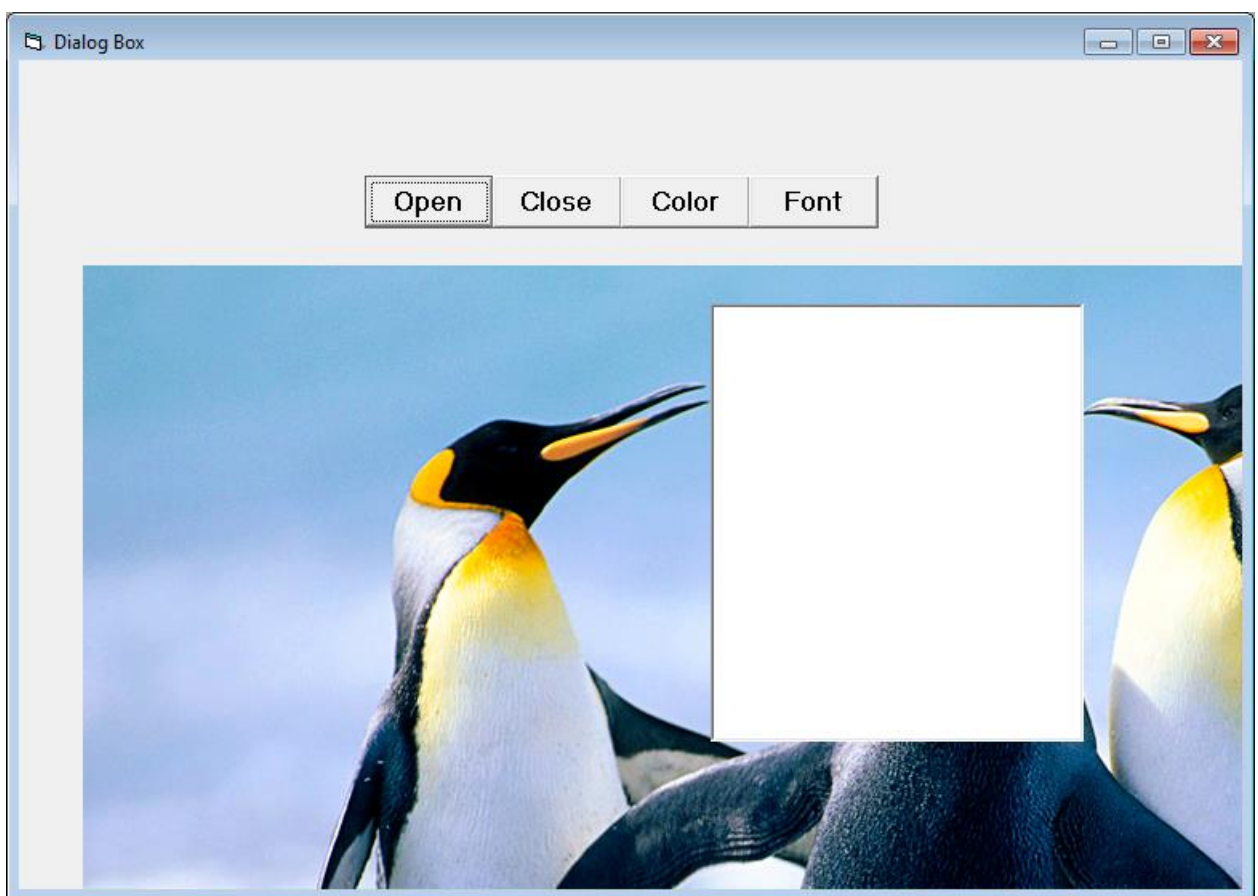
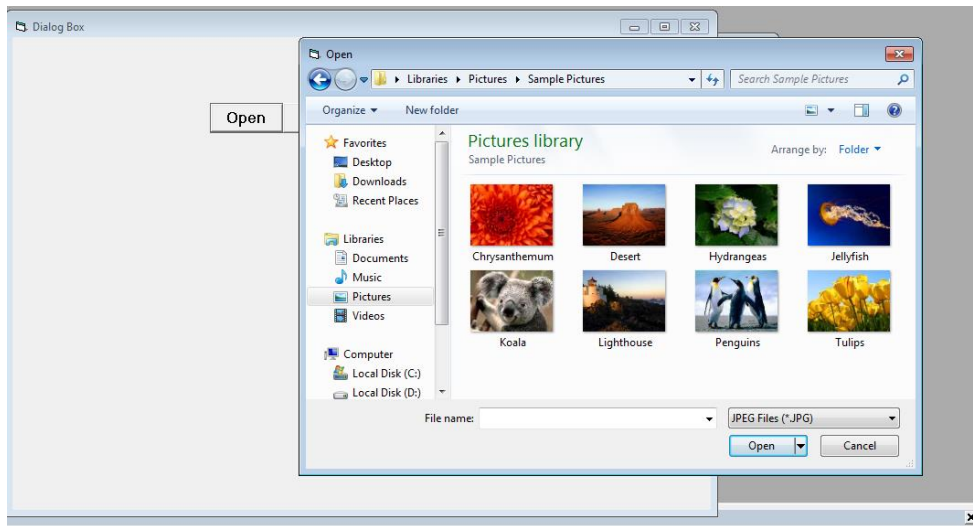
```

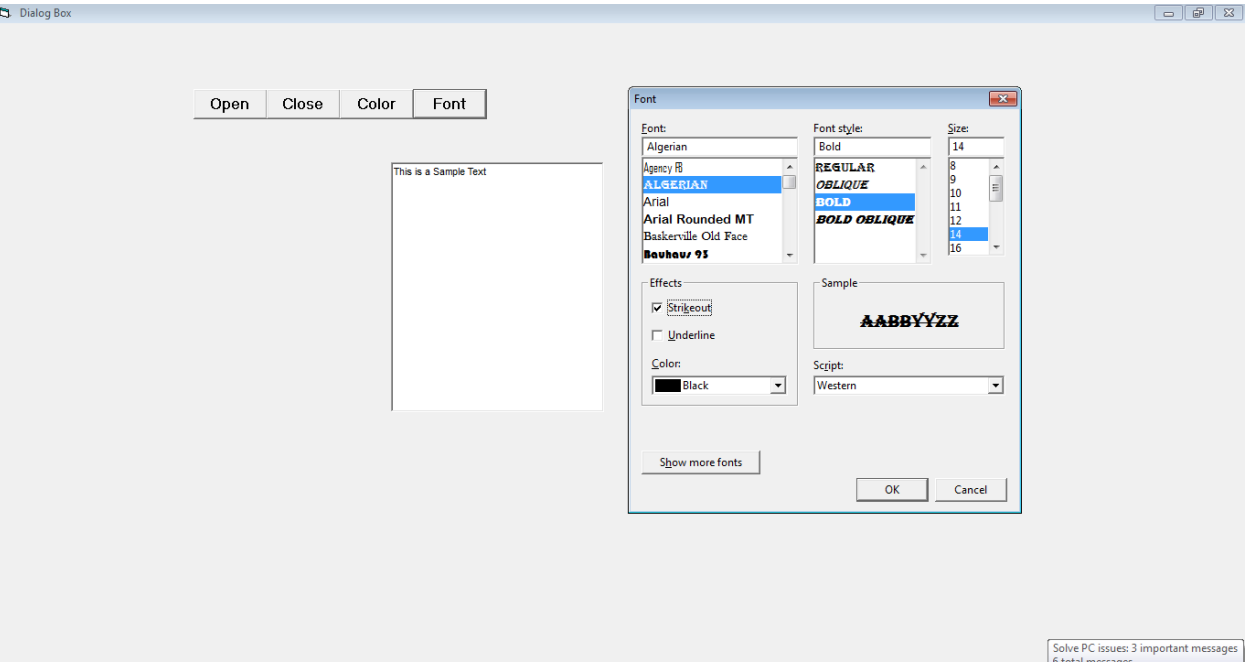
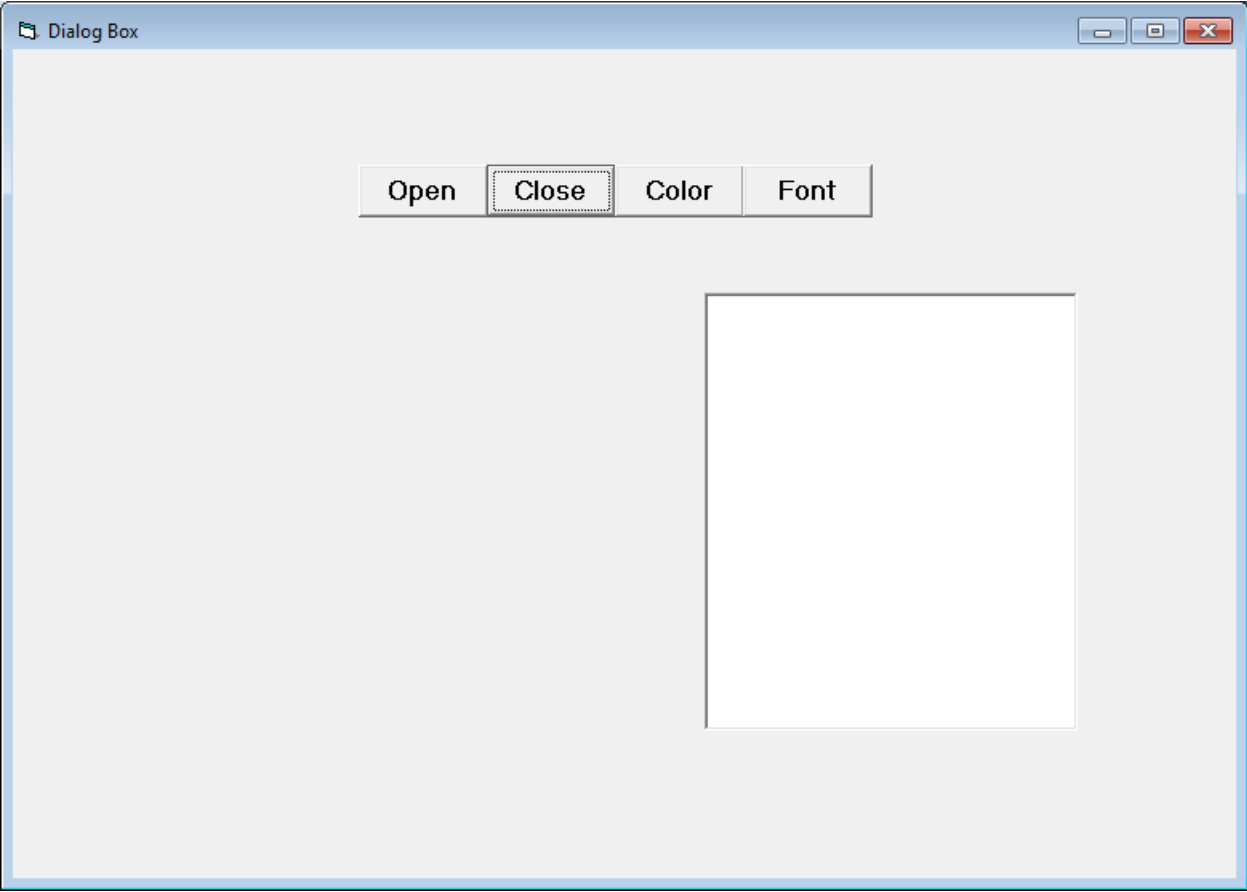
Private Sub cmd_Font_Click()
    On Error GoTo doNothing
    c_dialog.Flags = cdlCFApply Or cdlCFEffects
    c_dialog.ShowFont
    txt_data.Font = c_dialog.FontName
    txt_data.FontSize = c_dialog.FontSize
    txt_data.FontBold = c_dialog.FontBold
    txt_data.FontItalic = c_dialog.FontItalic
    txt_data.FontUnderline = c_dialog.FontUnderline
    txt_data.FontStrikethru = c_dialog.FontStrikethru
    txt_data.ForeColor = c_dialog.Color
    doNothing:
End Sub

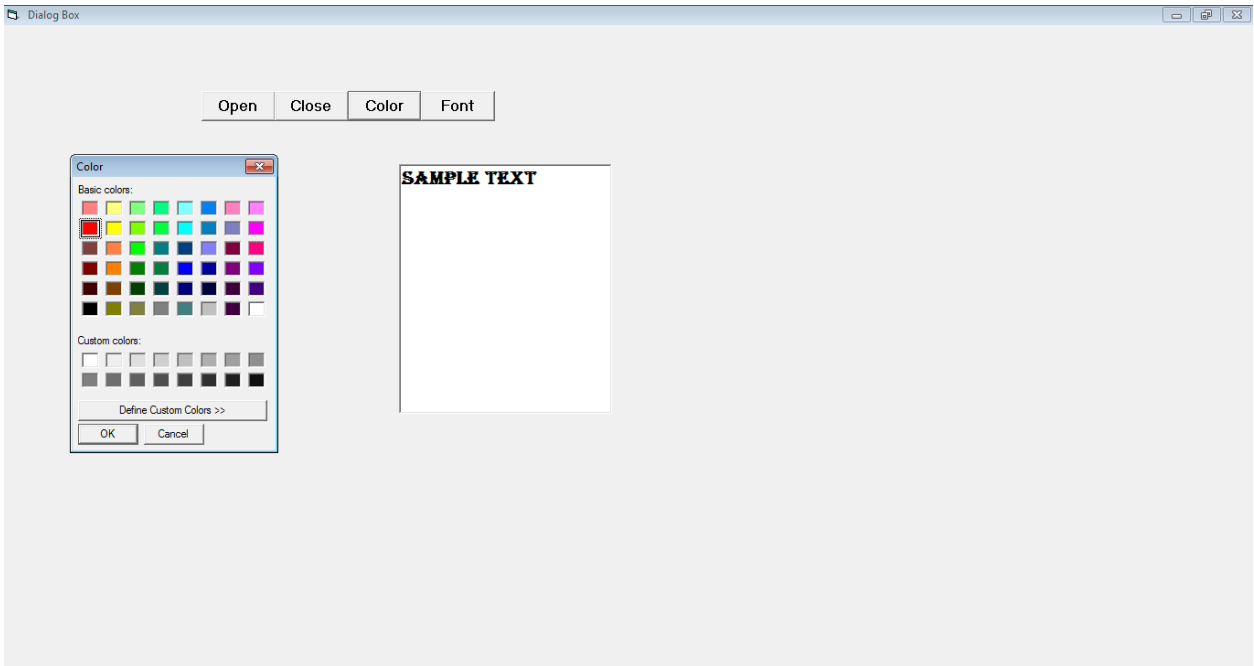
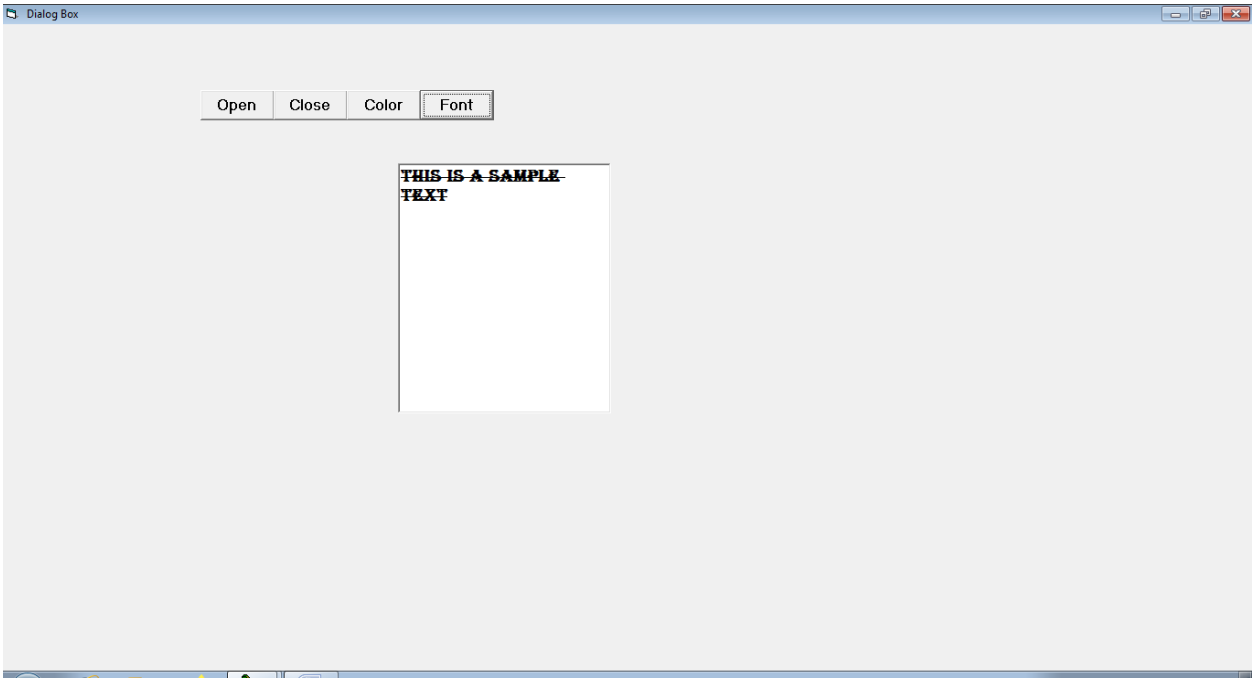
```

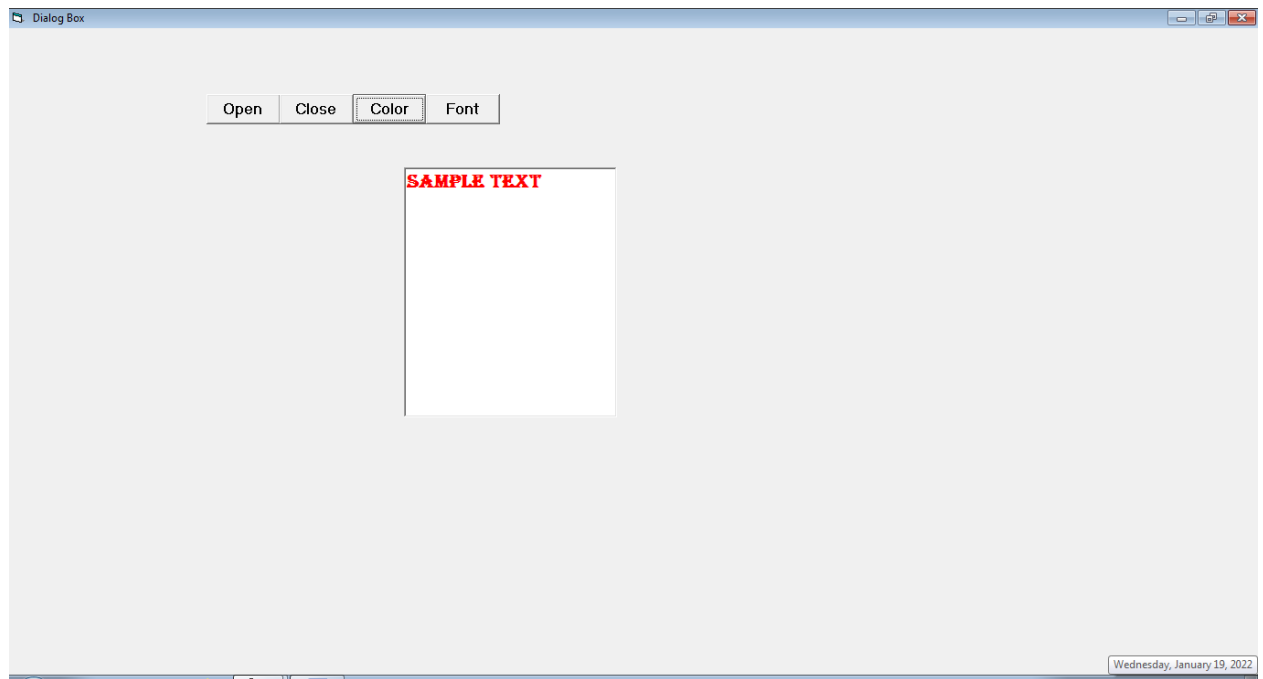
OUTPUT:











RESULT:

Thus, the common dialog boxes are implemented successfully.

5- WORKING WITH MENUS

DATE:18-02-2022

AIM:

To implement the various String operations using Menus.

DESCRIPTION:

1. Open a new visual basic project and save it with frm_menus.frm and lab5.vbp extension.
2. To Open the Menu Editor, Click Menu Editor button on the standard toolbar or Press Ctrl +E
3. Insert Menu Items with suitable Name and Caption
4. Write the coding for the appropriate menu items in coding window.
5. Execute the program by pressing F5 or Run button.
6. Give the input and verify the result.

REQUIREMENTS:

1. Forms-1
2. Menu Items-16

PROPERTIES TO CHANGE:

Control	Name	Caption
Form	frm_menus	“String Operations”
Menu Item	Mnu_chr	Chr
Menu Item	Mnu_asc	Asc
Menu Item	Mnu_lcase	Lcase
Menu Item	Mnu_ucase	Ucase
Menu Item	Mnu_space	Space
Menu Item	Mnu_string	String
Menu Item	Mnu_len	Len
Menu Item	Mnu_mid	Mid
Menu Item	Mnu_left	Left
Menu Item	Mnu_right	Right
Menu Item	Mnu_Instr	Instr
Menu Item	Mnu_Instrrev	Instrrev
Menu Item	Mnu_trim	Trim
Menu Item	Mnu_Ltrim	Ltrim
Menu Item	Mnu_Rtrim	Rtrim
Menu Item	Mnu_Strcomp	Strcomp

CODING:

FORM NAME: frm_menus

Private Sub mnu_asc_Click()

Dim value As **String**

value = InputBox("Enter the character")

Print " ASCII value of the " & value & " is " & Asc(value)

End Sub

Private Sub mnu_chr_Click()

Dim value As **Integer**

value = Val(InputBox("Enter the ASCII Value", "Menu Applications"))

Print "Character value of the " & value & " is " & Chr(value)

End Sub

Private Sub mnu_InStr_Click()

Dim value As **String**

Dim substr As **String**

value = InputBox("Enter the String")

substr = InputBox("Enter the Substr")

Print " The Position of the " & substr & " is " & InStr(1, value, substr)

End Sub

Private Sub mnu_InStrrev_Click()

Dim value As **String**

Dim substr As **String**

value = InputBox("Enter the String")

substr = InputBox("Enter the Substr")

Print " The Position of the " & substr & " is " & InStrRev(value, substr)

End Sub

Private Sub mnu_LCase_Click()

Dim value As **String**

value = InputBox("Enter the String")

Print " Lowercase value of the " & value & " is " & LCase(value)

End Sub

Private Sub mnu_Left_Click()

Dim value As **String**

Dim n As **Integer**

value = InputBox("Enter the String")

n = Val(InputBox("Enter the Position"))

Print " Left value of the " & value & " is " & Left(value, n)

End Sub

Private Sub mnu_Len_Click()

Dim value As **String**

value = InputBox("Enter the String")

Print " Length of the " & value & " is " & Len(value)

End Sub

```
Private Sub mnu_Ltrim_Click()  
    Dim value As String  
    value = InputBox("Enter the String")  
    Print value & " After Trimming is " & LTrim(value) + "."  
End Sub
```

```
Private Sub mnu_Mid_Click()  
    Dim value As String  
    Dim n As Integer  
    value = InputBox("Enter the String")  
    n = Val(InputBox("Enter the Position"))  
    Print " Mid value of the " & value & " is " & Mid(value, n)  
End Sub
```

```
Private Sub mnu_Right_Click()  
    Dim value As String  
    Dim n As Integer  
    value = InputBox("Enter the String")  
    n = Val(InputBox("Enter the Position"))  
    Print " Right value of the " & value & " is " & Right(value, n)  
End Sub
```

```
Private Sub mnu_Rtrim_Click()  
    Dim value As String  
    value = InputBox("Enter the String")  
    Print value & " After Trimming is " & RTrim(value) + "."  
End Sub
```

```
Private Sub mnu_Space_Click()  
    Dim value As String  
    Dim n As Integer  
    n = Val(InputBox("Enter the Space Value"))  
    value = "Visual" + Space(n) + "Programming"  
    Print value  
End Sub
```

```
Private Sub mnu_Strcomp_Click()  
    Dim str1 As String  
    Dim str2 As String  
    Dim value As Integer  
    str1 = InputBox("Enter the String")  
    str2 = InputBox("Enter the String to Compare")  
    value = StrComp(str1, str2)  
    If value = 0 Then  
        Print "Given Strings (" & str1 & ", " & str2 & ") are Equal"  
    ElseIf value = 1 Then  
        Print "Given String (" & str1 & " is greater than " & str2 & ")"  
    Else  
        Print "Given Strings (" & str1 & " is lesser than " & str2 & ")"  
    End If  
End Sub
```

Private Sub mnu_String_Click()

Dim value As **String**

Dim n As **Integer**

n = Val(InputBox("Enter the Value"))

value = "Visual Programming"

Print String(n, value)

End Sub

Private Sub mnu_trim_Click()

Dim value As **String**

value = InputBox("Enter the String")

Print value & " After Trimming is " & Trim(value) + "."

End Sub

Private Sub mnu_UCase_Click()

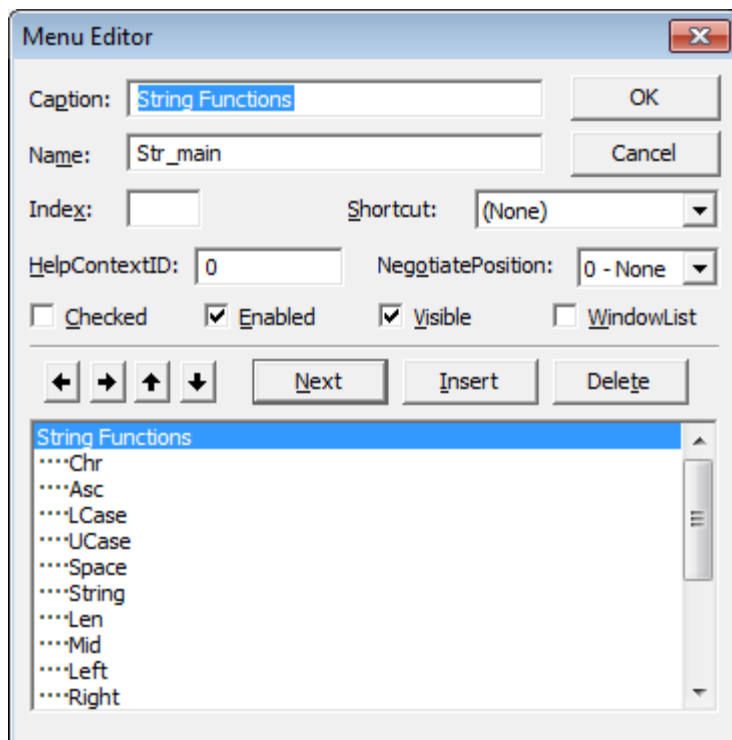
Dim value As **String**

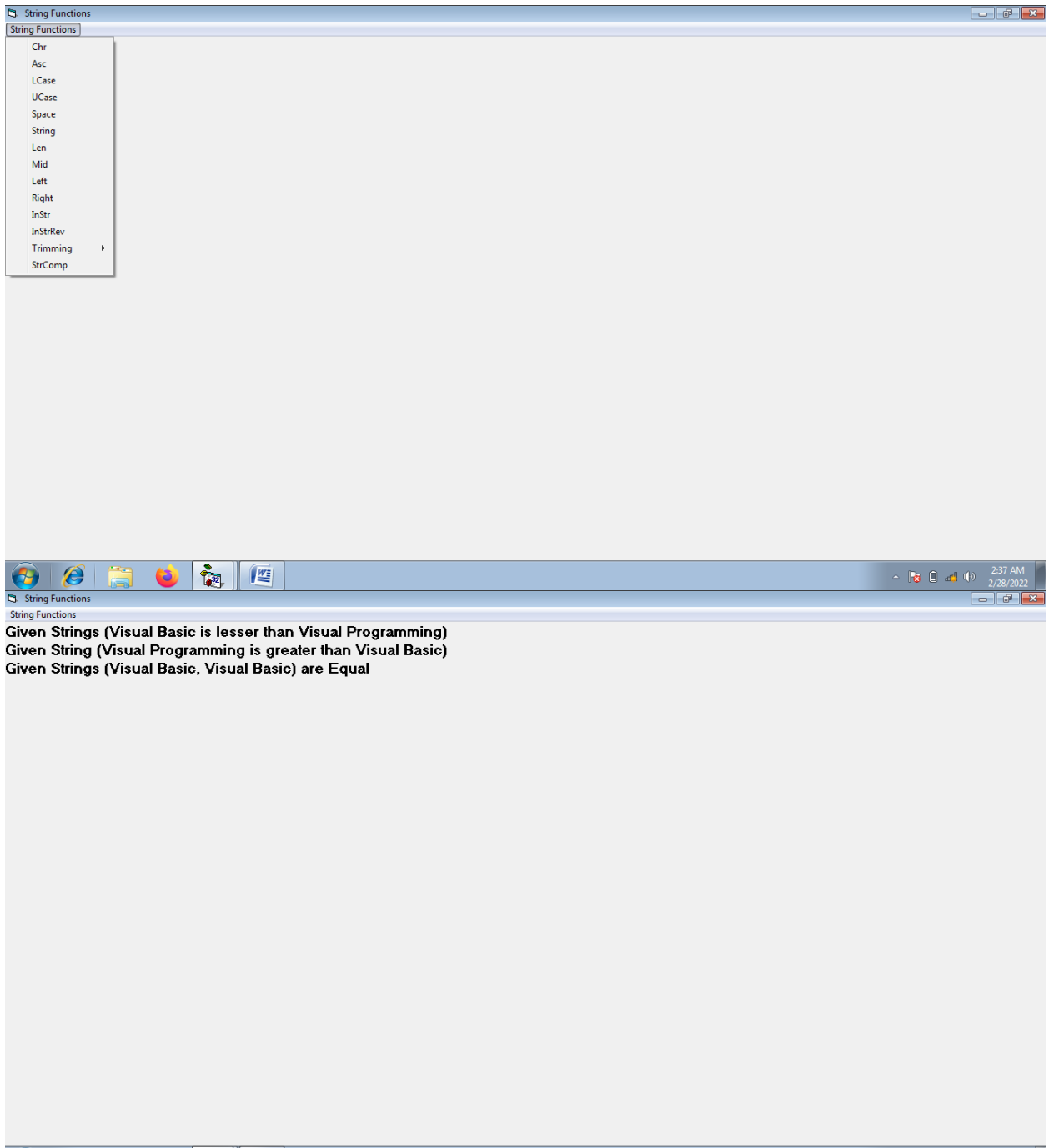
value = InputBox("Enter the String")

Print " Uppercase value of the " & value & " is " & UCase(value)

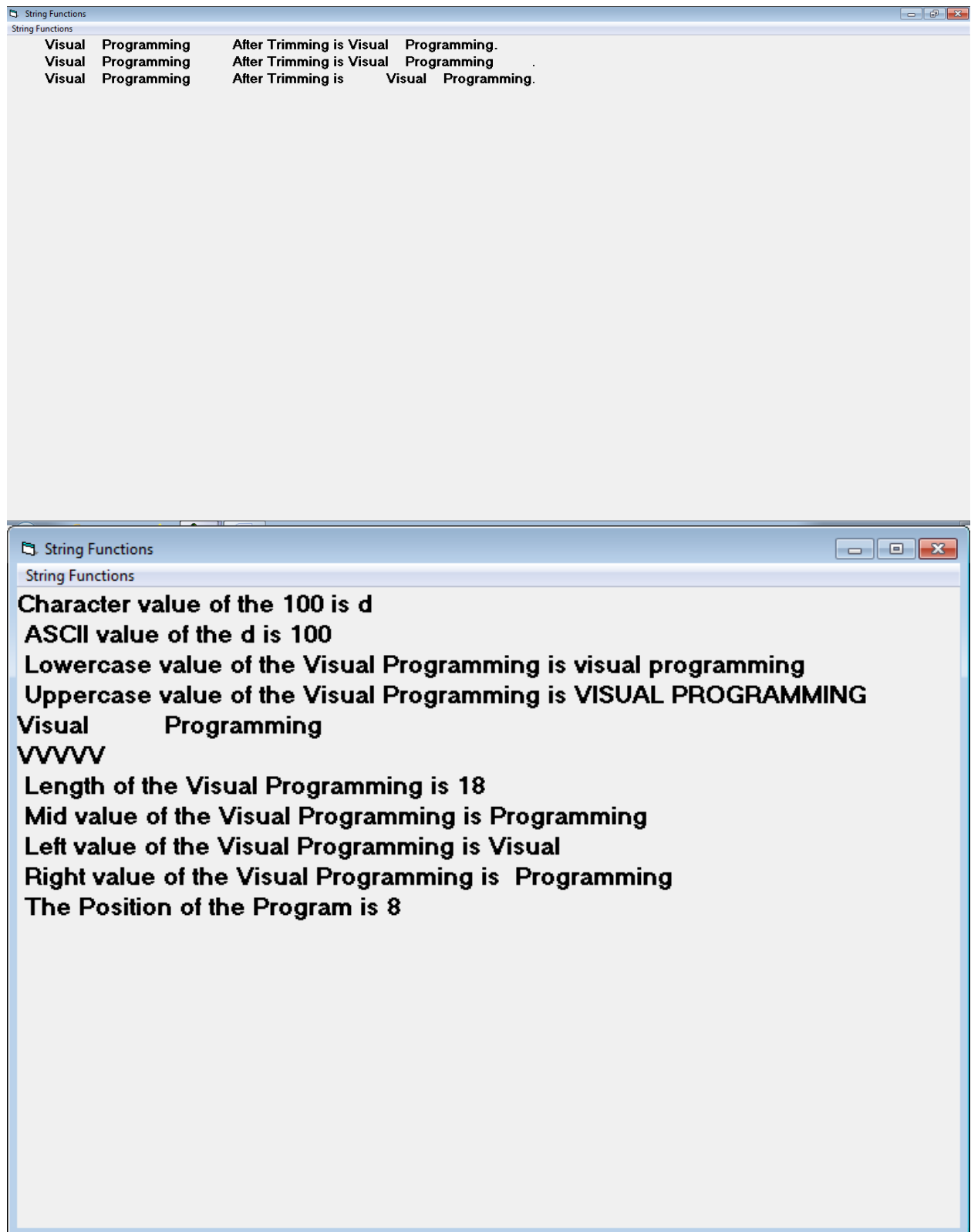
End Sub

OUTPUT:





Given Strings (Visual Basic is lesser than Visual Programming)
Given String (Visual Programming is greater than Visual Basic)
Given Strings (Visual Basic, Visual Basic) are Equal



RESULT:

Thus, the menu operations are implemented successfully.

6- WORKING WITH DATA CONTROLS

DATE:24-02-2022

AIM:

To perform CRUD operations in Access Database using Data Controls.

DESCRIPTION:

1. Open a new visual basic project and save it with frm_stud.frm and lab6.vbp extension.
2. Select Addin and click data manager to create an access database.
3. Create a table with necessary fields with suitable datatypes.
4. Configure the connection string in Data control.
5. Configure the record source to Data control.
6. Write the coding for the insert, update and delete buttons in coding window.
7. Execute the program by pressing F5 or Run button.
8. Give the input and verify the result.

REQUIREMENTS:

1. Forms-1
2. Labels-3
3. TextBox-3
4. Data Control-1

PROPERTIES TO CHANGE:

Control	Name	Caption
Form	frm_Stud	"Student Details"
TextBox	txt_studno	
TextBox	txt_name	
TextBox	txt_course	
DataControl	Data1	

CODING:

FORM NAME: frm_menus

Private Sub cmd_Add_Click()

Data1.Recordset.AddNew

End Sub

Private Sub cmd_Delete_Click()

If MsgBox("Are you confirm to delete the record", **vbYesNo**) = **vbYes** **Then**

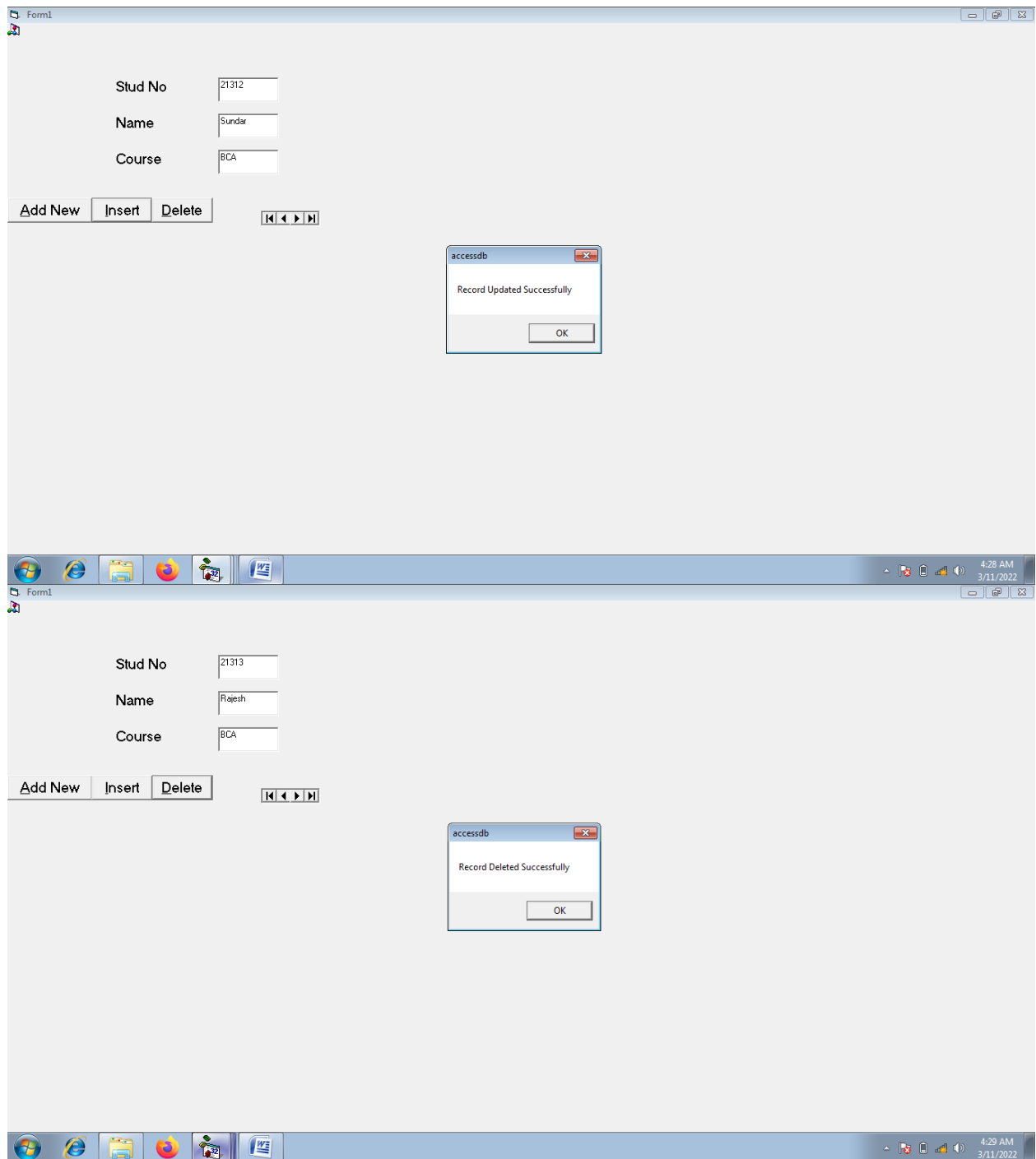
Data1.Recordset.Delete

```
MsgBox ("Record Deleted Successfully")
Data1.Refresh
End If
End Sub
```

```
Private Sub cmd_Insert_Click()
Data1.Recordset(0) = txt_Studno.Text
Data1.Recordset(1) = txt_Name.Text
Data1.Recordset(2) = txt_course.Text
Data1.Recordset.Update
MsgBox ("Record Updated Successfully")
Data1.Refresh
End Sub
```

OUTPUT:

The screenshot shows a Windows application window titled "Form1". Inside the window, there are three text boxes arranged vertically. The first is labeled "Stud No" and contains the text "21312". The second is labeled "Name" and contains the text "Sundar". The third is labeled "Course" and contains the text "BCA". Below these text boxes, there are three buttons: "Add New", "Insert", and "Delete". To the right of these buttons, there is a set of four navigation icons: a double left arrow, a single left arrow, a single right arrow, and a double right arrow.



RESULT:

Thus, the database operations using data controls are implemented successfully.

7- APPLICATION WITH ERROR TRAPPING

DATE:03-03-2022

AIM:

To create an application to implement various error handling operations in Visual basic.

DESCRIPTION:

1. Open a new visual basic project and save it with frm_errors.frm and lab7.vbp extension.
2. Create a button and listbox into the form
3. Accept the n value and get n integer values from inputbox and assign the same in listbox.
4. Write the coding for trapping overflow, checking numeric values in coding window.
5. Execute the program by pressing F5 or Run button.
6. Give the input and verify the result.

REQUIREMENTS:

1. Forms-1
2. Button-1
3. ListBox-1

PROPERTIES TO CHANGE:

Control	Name	Caption
Form	frm_error	"Error Trapping"
Button	cmd_button	Input
ListBox	lst_value	

CODING:

FORM NAME: frm_error

Dim int_array(5) As **Integer**

Dim n As **Integer**

Private Sub cmd_input_Click()

Dim value As **String**

On Error GoTo errorhandler

lst_value.Clear

n = Val(InputBox("Enter the n value", "Error Trapping", 0))

For i = 1 **To** n

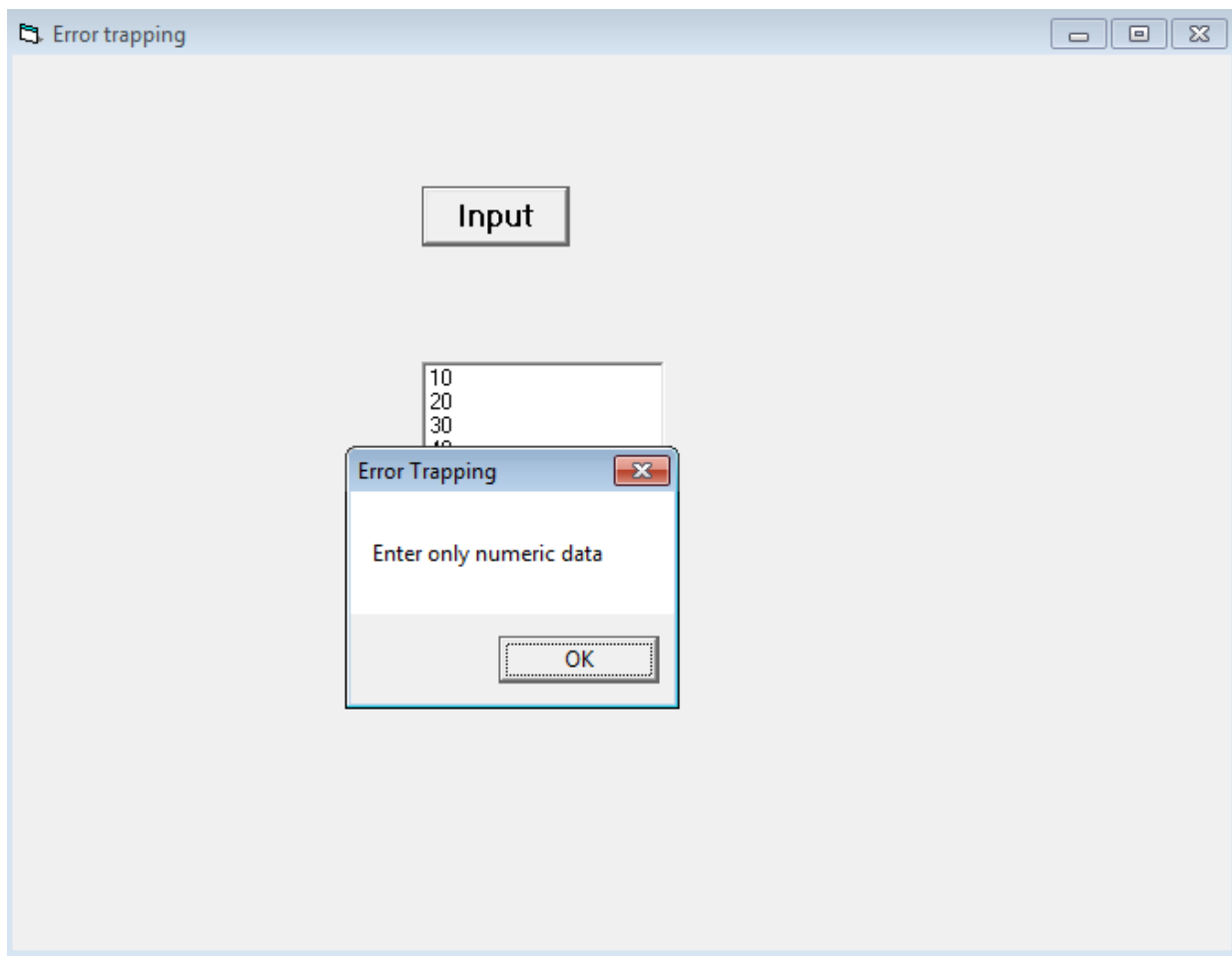
value = InputBox("Enter the value", "Error Trapping")

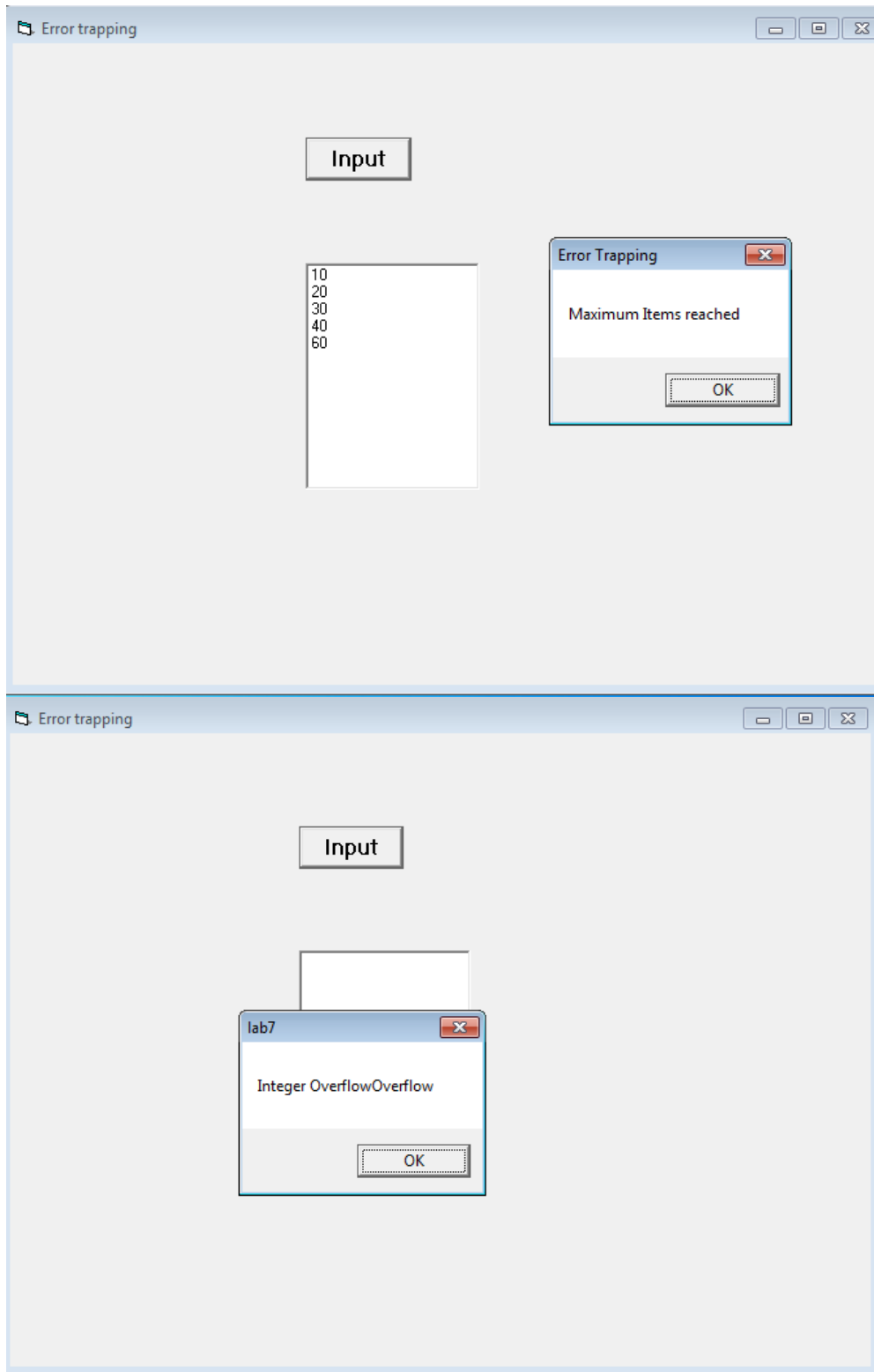
```

int_array(i) = CInt(value)
lst_value.AddItem int_array(i)
Next
errorhandler:
If Err.Number = 13 Then
MsgBox "Enter only numeric data", vbOKOnly, "Error Trapping"
value = InputBox("Enter the value", "Error Trapping")
int_array(i) = CInt(value)
Resume Next
ElseIf Err.Number = 6 Then
MsgBox ("Integer Overflow" & Err.Description)
value = InputBox("Enter the value", "Error Trapping")
int_array(i) = CInt(value)
Resume Next
ElseIf Err.Number = 9 Then
Resume dooverflow
End If
dooverflow:
MsgBox "Maximum Items reached", vbOKOnly, "Error Trapping"
End Sub

```

OUTPUT:





RESULT:

Thus, the various error trapping operations are implemented successfully.

8- DRAG AND DROP APPLICATIONS

DATE:10-03-2022

AIM:

To create an application to perform Drag and Drop events in Visual Basic

DESCRIPTION:

1. Open a new visual basic project and save it with frm_dragdrop.frm and lab8.vbp extension.
2. Insert Two listboxes and command button into Form
3. Accept the values and add it in listbox
4. Write the coding to implement Drag and drop events for both listboxes
5. Execute the program by pressing F5 or Run button.
6. Give the input and verify the result.

REQUIREMENTS:

1. Forms-1
2. Button-1
3. ListBox-2

PROPERTIES TO CHANGE:

Control	Name	Caption
Form	frm_dragdrop	"Drag and Drop Applications"
Button	cmd_button	"Add Items"
ListBox	lst_input	
ListBox	lst_output	

CODING:

FORM NAME: frm_dragdrop

Private Sub cmd_Add_Click()

lst_input.AddItem InputBox("Enter the Value", "Drag & Drop Program")

End Sub

Private Sub lst_input_DragDrop(Source **As Control**, X **As Single**, Y **As Single**)

If Source = lst_output **Then**

lst_input.AddItem lst_output.Text

lst_output.RemoveItem lst_output.ListIndex

End If

End Sub

```
Private Sub lst_input_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
If lst_input.ListCount > 0 Then
```

```
lst_input.Drag
```

```
End If
```

```
End Sub
```

```
Private Sub lst_output_DragDrop(Source As Control, X As Single, Y As Single)
```

```
If Source = lst_input Then
```

```
lst_output.AddItem lst_input.Text
```

```
lst_input.RemoveItem lst_input.ListIndex
```

```
End If
```

```
End Sub
```

```
Private Sub lst_output_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

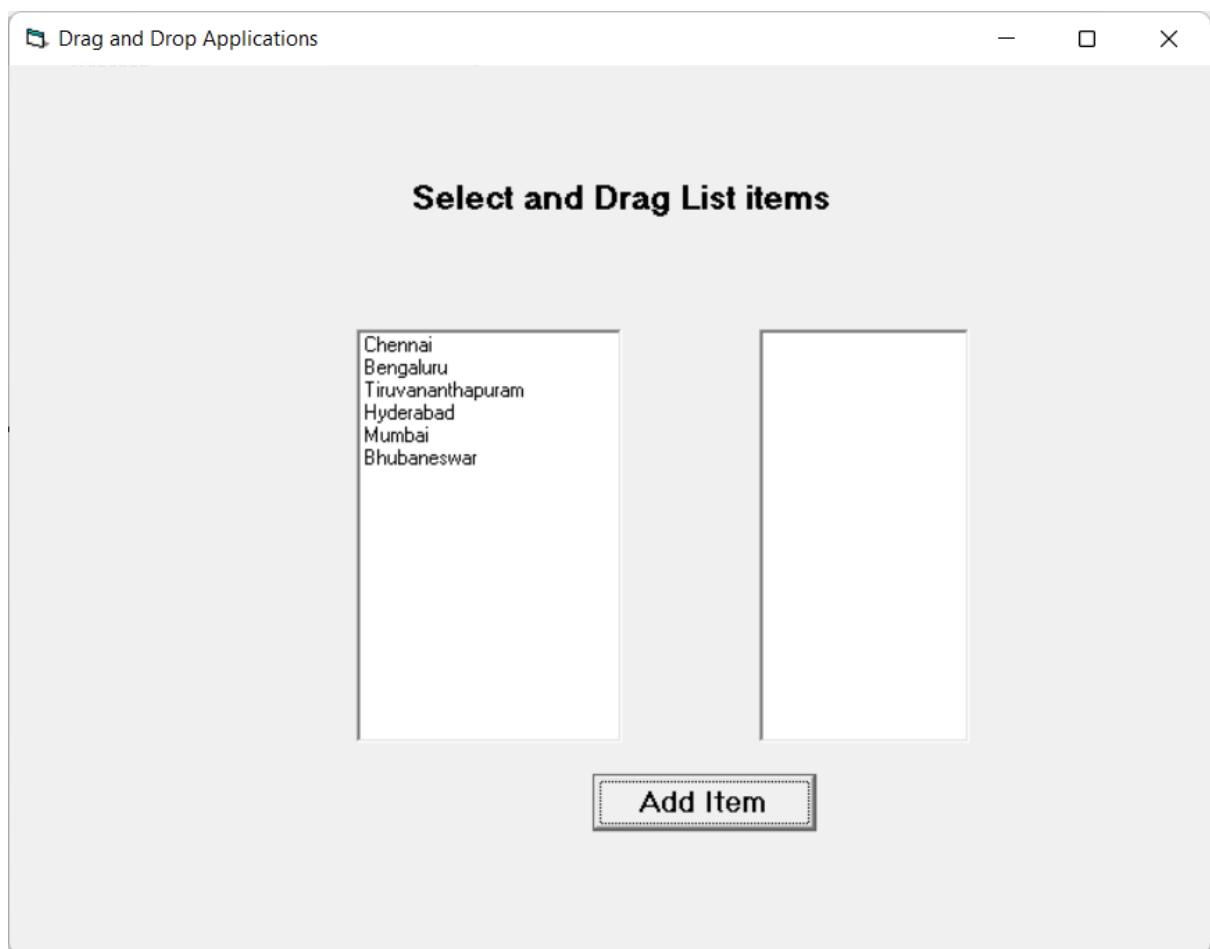
```
If lst_output.ListCount > 0 Then
```

```
lst_output.Drag
```

```
End If
```

```
End Sub
```

OUTPUT:

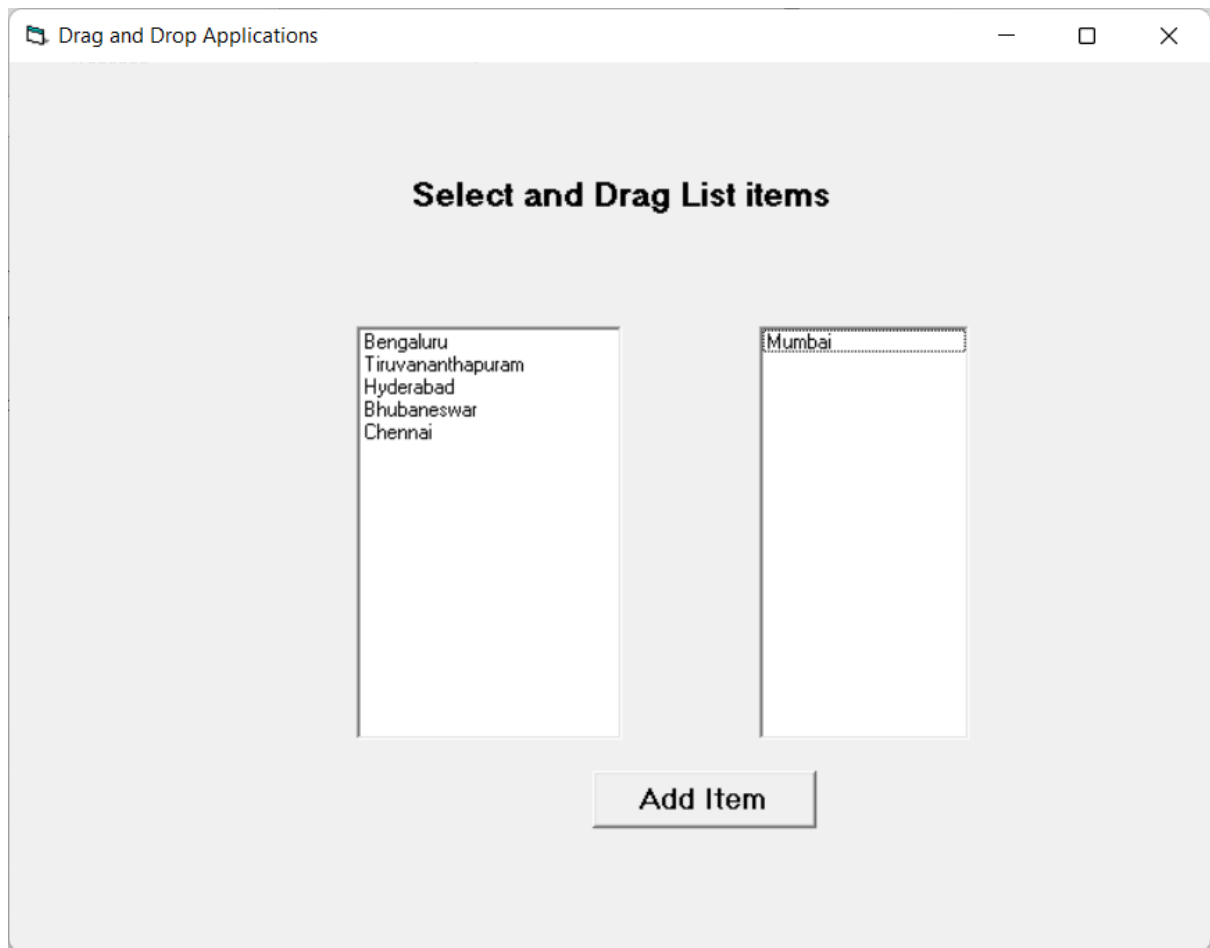


Select and Drag List items

Bengaluru
Tiruvananthapuram
Hyderabad
Bhubaneswar

Chennai
Mumbai

Add Item



RESULT:

Thus, the drag and drop operations are implemented successfully.

9- APPLICATION USING FILE SYSTEM OBJECTS

DATE:17-03-2022

AIM:

To create a notepad application using file system objects in Visual Basic

DESCRIPTION:

1. Open a new visual basic project and save it with frm_notepad.frm and lab9.vbp extension.
2. Insert richtextbox from components tab
3. Insert the appropriate menuitems
4. Write the coding for menuitems
5. Execute the program by pressing F5 or Run button.
6. Give the input and verify the result.

REQUIREMENTS:

1. Forms-1
2. RichTextBox-1
3. MenuItems-16

PROPERTIES TO CHANGE:

Control	Name	Caption
Form	frm_notepad	“Simple Notepad Application”
RichTextBox	Rtbinput	
MenuItem	Mnu_file	File
MenuItem	mnu_New	New
MenuItem	mnu_open	Open
MenuItem	mnu_save	Save
MenuItem	mnu_close	Close
MenuItem	mnu_Print	Print
MenuItem	mnu_Exit	Exit
MenuItem	Mnu_edit	Edit
MenuItem	Cut	Cut
MenuItem	copy	Copy
MenuItem	Paste	Paste
MenuItem	mnu_select	Select All
MenuItem	Mnu_format	Format
MenuItem	mnu_Font	Font
MenuItem	mnu_Color	Color
MenuItem	mnu_bold	Bold
MenuItem	mnu_italic	Italic
MenuItem	mnu_underline	Underline
MenuItem	mnu_strikethough	StrikeThrough

CODING:

```
Private Sub mnu_bold_Click()
```

```
    If rtbinput.SelBold = False Then  
        rtbinput.SelBold = True  
        mnu_bold.Checked = True
```

```
    Else
```

```
        rtbinput.SelBold = False  
        mnu_bold.Checked = False
```

```
    End If
```

```
End Sub
```

```
Private Sub mnu_close_Click()
```

```
Dim result As Variant
```

```
    If Len(rtbinput.Text) > 0 Then  
        result = MsgBox("Do u want to Close the file", vbYesNo, "Sample Notepad  
Application")
```

```
        If result = vbYes Then  
            rtbinput.Text = ""
```

```
        End If
```

```
    End If
```

```
End Sub
```

```
Private Sub mnu_Exit_Click()
```

```
    End
```

```
End Sub
```

```
Private Sub mnu_italic_Click()
```

```
    If rtbinput.SelItalic = False Then  
        rtbinput.SelItalic = True  
        mnu_italic.Checked = True
```

```
    Else
```

```
        rtbinput.SelItalic = False  
        mnu_italic.Checked = False
```

```
    End If
```

```
End Sub
```

```
Private Sub mnu_New_Click()
```

```
Dim result As Variant
```

```
Dim fn As String
```

```
    If Len(rtbinput.Text) > 0 Then  
        result = MsgBox("Do U Want to Save the file", vbYesNoCancel, "Sample Notepad  
Application")
```

```
        If result = vbYes Then
```

```
            CommonDialog1.ShowSave  
            fn = CommonDialog1.FileName
```

```
            Open fn For Append As #1
```

```
            Print #1, rtbinput.Text
```

```
            Close #1
```

```
        Elseif result = vbNo Then
```

```
            rtbinput.Text = ""
```

```

        End If
    End If
End Sub

Private Sub mnu_open_Click()
Dim fn As String
Dim rd As String
    CommonDialog1.ShowOpen
    fn = CommonDialog1.FileName
    rtbinput.Text = ""
    Open fn For Input As #2
    Do While Not EOF(2)
    Line Input #2, rd
    rtbinput.Text = rtbinput.Text + rd + vbCrLf
    Loop
    Close #2
End Sub

Private Sub mnu_Print_Click()
    CommonDialog1.ShowPrinter
    Printer.Print
End Sub

Private Sub mnu_save_Click()
Dim fn As String
    CommonDialog1.ShowSave
    fn = CommonDialog1.FileName
    Open fn For Output As #1
    Print #1, rtbinput.Text
    Close #1
End Sub

Private Sub Cut_Click()
    Clipboard.Clear
    Clipboard.SetText (rtbinput.SelText)
    rtbinput.SelText = ""
End Sub

Private Sub Copy_Click()
    Clipboard.Clear
    Clipboard.SetText (rtbinput.SelText)
End Sub

Private Sub mnu_select_Click()
    rtbinput.SelStart = 0
    rtbinput.SelLength = Len(rtbinput.Text)
End Sub

Private Sub mnu_strikethough_Click()
    If rtbinput.SelStrikeThru = False Then
        rtbinput.SelStrikeThru = True
    End If
End Sub

```

```

mnu_strikethrough.Checked = True
Else
rtbinput.SelStrikeThru = False
mnu_strikethrough.Checked = False
End If

```

End Sub

```

Private Sub mnu_underline_Click()
If rtbinput.SelUnderline = False Then
rtbinput.SelUnderline = True
mnu_underline.Checked = True
Else
rtbinput.SelUnderline = False
mnu_underline.Checked = False
End If

```

End Sub

```

Private Sub Paste_Click()
rtbinput.SelText = Clipboard.GetText
End Sub

```

```

Private Sub mnu_Font_Click()
CommonDialog1.ShowFont
rtbinput.Font = CommonDialog1.FontName
rtbinput.Font.Bold = CommonDialog1.FontBold
rtbinput.Font.Size = CommonDialog1.FontSize
rtbinput.Font.Italic = CommonDialog1.FontItalic
End Sub

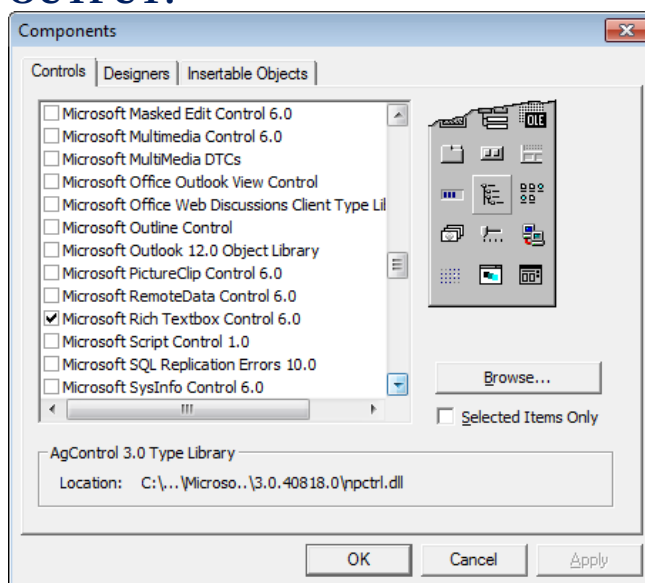
```

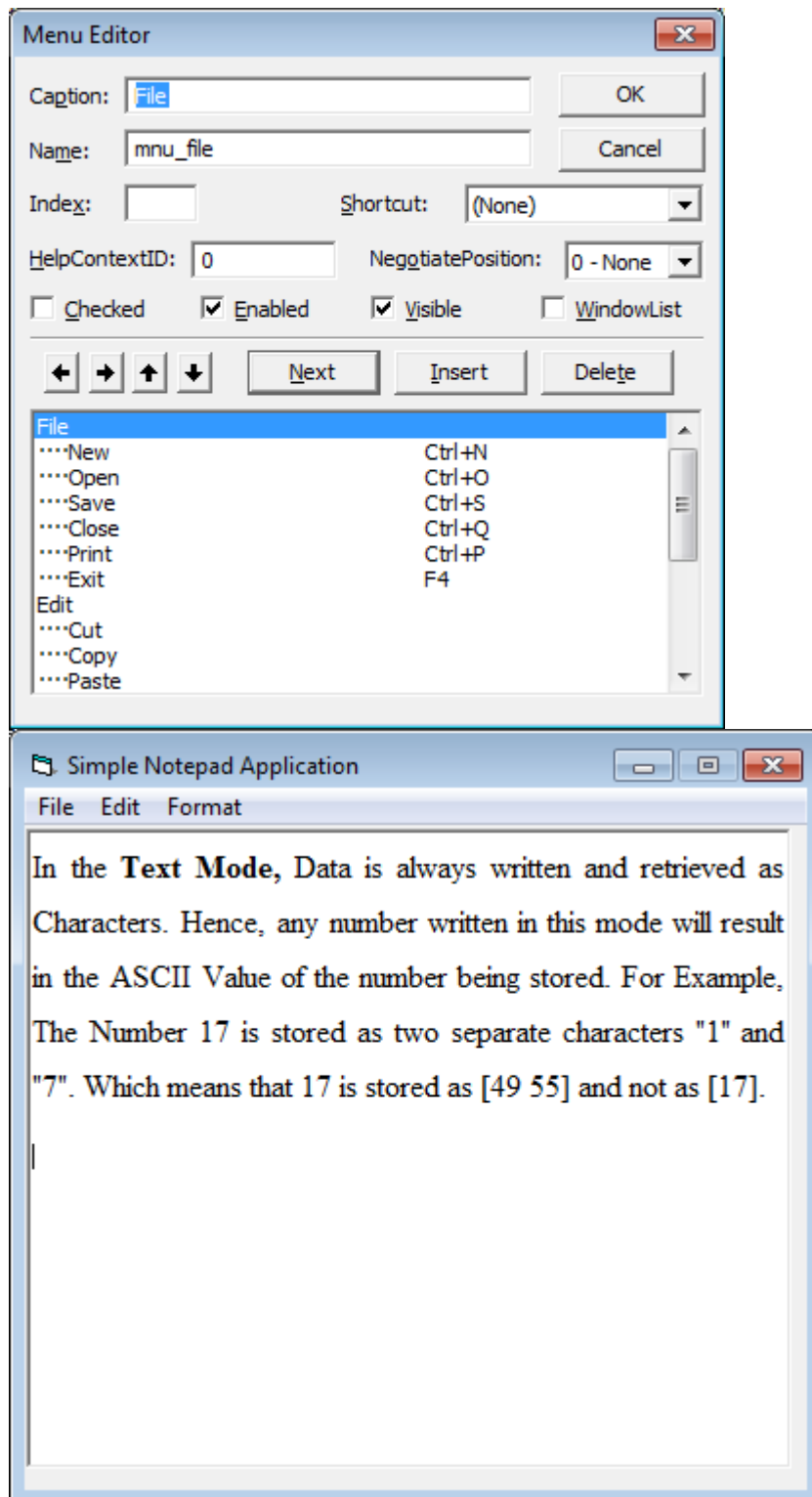
```

Private Sub mnu_Color_Click()
CommonDialog1.ShowColor
rtbinput.SelColor = CommonDialog1.Color
End Sub

```

OUTPUT:





RESULT:

Thus, Files and File system objects are implemented successfully.

10- CREATING ACTIVE X CONTROLS

DATE:24-03-2022

AIM:

To create an Active X control using Visual Basic

DESCRIPTION:

1. Open a new project and select ActiveX control option.
2. Name the user control as lab10
3. Add three labels and three textboxes to the user control.
4. Write appropriate code to assign textboxes.
5. Save the project as lab10.vbp and create lab10.ocx file using makes option in the file menu.
6. Open a Standard.Exe project and include the lab10.ocx in your project by selecting project/components menu.
7. Use the control in your project as needed and execute it.

REQUIREMENTS:

1. Forms-1
2. Label-4
3. TextBox-4
4. UserControl-1

PROPERTIES TO CHANGE:

Control	Name	Caption
Form	frm_main	"Active X Controls"
TextBox	txt_name	
TextBox	txt_regno	
TextBox	txt_course	
TextBox	txt_subject	
UserControl	Lab10	

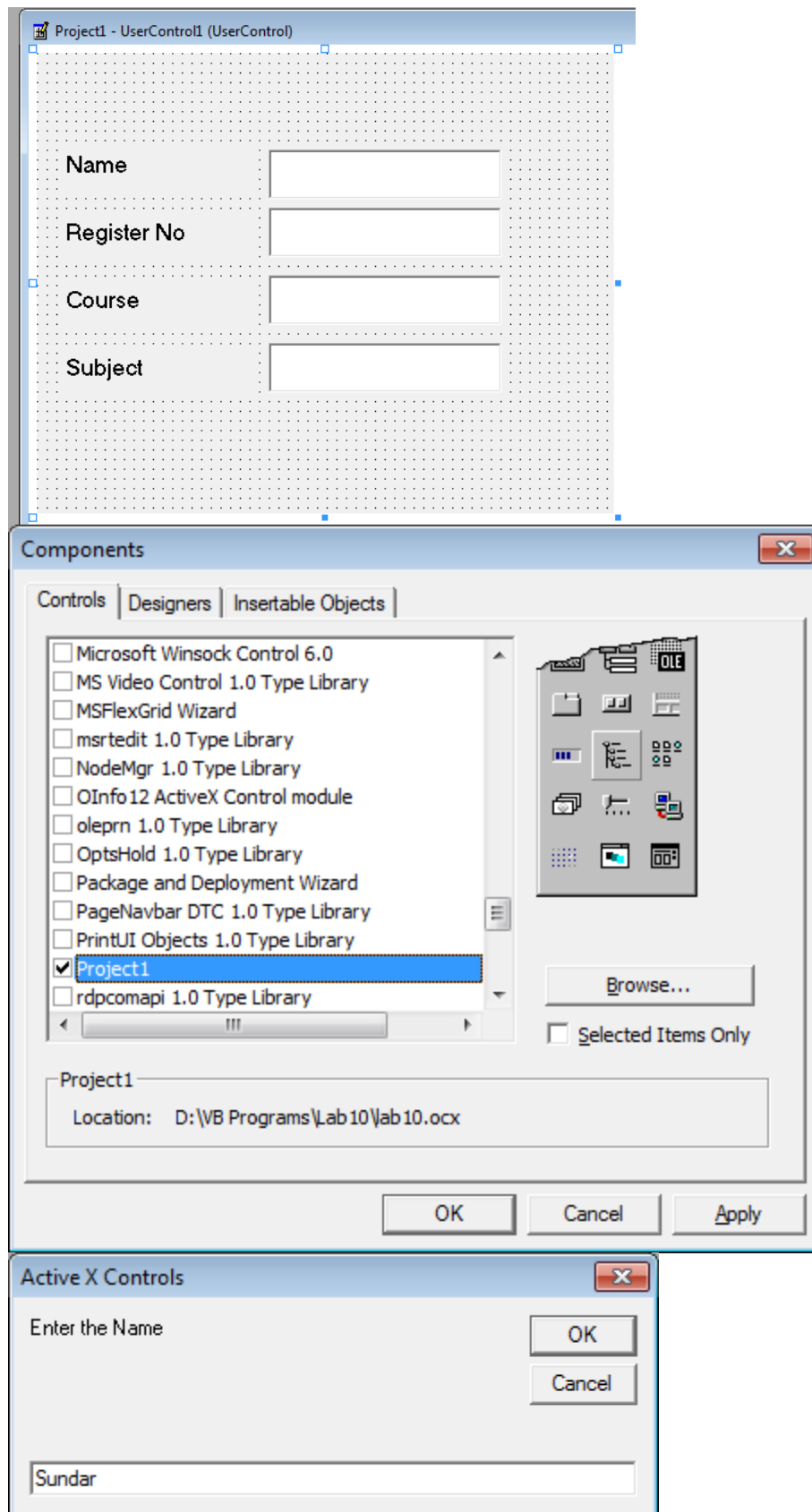
CODING:

Private Sub UserControl_Initialize()

```
txt_name.Text = InputBox("Enter the Name", "Active X Controls")
txt_regno.Text = InputBox("Enter the Reg No", "Active X Controls")
txt_course.Text = InputBox("Enter the Course", "Active X Controls")
txt_subject.Text = InputBox("Enter the Subject", "Active X Controls")
```

End Sub

OUTPUT:



The image displays a sequence of four windows from an Active X control application. The first three are small dialog boxes titled "Active X Controls" with a close button (X) in the top right corner. Each dialog box contains a label, a text input field, and "OK" and "Cancel" buttons.

- Dialog 1:** The label is "Enter the Reg No". The text input field contains the value "21312".
- Dialog 2:** The label is "Enter the Course". The text input field contains the value "BCA".
- Dialog 3:** The label is "Enter the Subject". The text input field contains the value "Visual Programming".

The fourth window is a larger main form titled "Active X Controls" with standard window controls (minimize, maximize, close) in the top right. It contains four labels and corresponding text input fields arranged in a vertical list:

Label	Value
Name	Sundar
Register No	21312
Course	BCA
Subject	Visual Programming

RESULT:

Thus, Active X Control are implemented successfully.