**PART B**

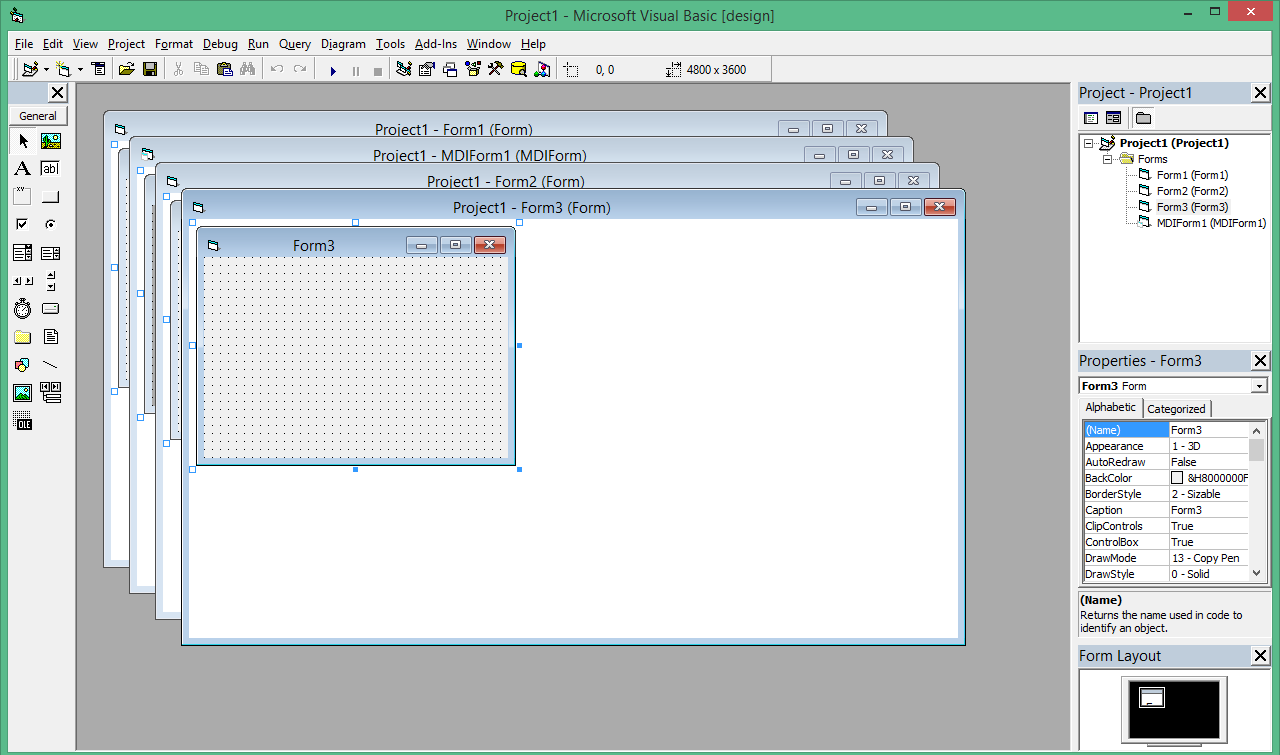
|  |  |
| --- | --- |
| **B1** | **VB Application: Arrange the Windows using MDI and Child Forms with Menu** |
| **B2** | **VB Application: Change the Background Color of the form using Scrollbars** |
| **B3** | **VB Application: Draw shapes using mouse with Pull Down/ Pop Up Menu** |
| **B4** | **VB Application: Load an image using drive list box, directory list box, file list box and image control** |
| **B5** | **VB Application: Calculate age using DTPicker Tool** |
| **B6** | **VB Application: Swap two numbers using CallByValue() and CallByReference()** |
| **B7** | **VB Application: A Simple Calculator** |
| **B8** | **VB Application: Splash Screen with Progress Bar** |
| **B9** | **VB Application: Student database with all options using DAO Control** |
| **B10** | **VB Application: Load files like HTML, MP4,.png etc in form using Web Control** |
| **B11** | **VB Application: Student Database using VB Wizard** |
| **B12** | **VB Application: MS Word using OLE Control** |
| **B13** | **VB Application: Design Layout Using ActiveX Control** |
| **B14** | **VB Application: Find Cube of a Number using ActiveX DLL** |
| **B15** | **VB Application: Animation using Line Control** |

**B1.VB Application: Arrange the Windows using MDI and Child Forms**

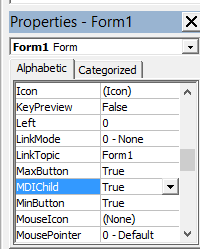
**Aim:** To Create a VB Application to arrange windows using MDI & Child forms

**Procedure:**

1. Project menu> Add MDI form / Project properties window>Project right click > **Add MDI Form**
2. Now **add Form2, Form3 along MDI Form1 & Form1**

****

1. Now Project has 3 forms (Form1, Form2, Form3) and 1 MDI form
2. Go to **Project menu> Properties> Set Start up Object as MDI Form1**
3. **Design the MDI form with Menu Arrange>Horizontal, Vertical, Cascade**, Exit using Menu Editor (Use Tools > Menu Editor for creating menu)
4. **Set the properties MDIChild =True for Form1, Form2 and Form3**

****

**Source Code**

Private Sub Cascade\_Click()

MDIForm1.arrange (0)

End Sub

Private Sub Horizontal\_Click()

MDIForm1.arrange (1)

End Sub

Private Sub MDIForm\_Load()

Form1.Show

Form2.Show

Form3.Show

End Sub

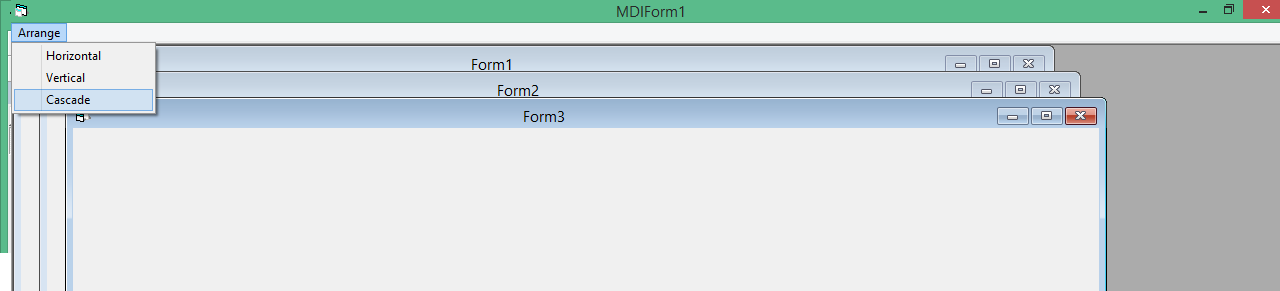
Private Sub Vertical\_Click()

MDIForm1.arrange (2)

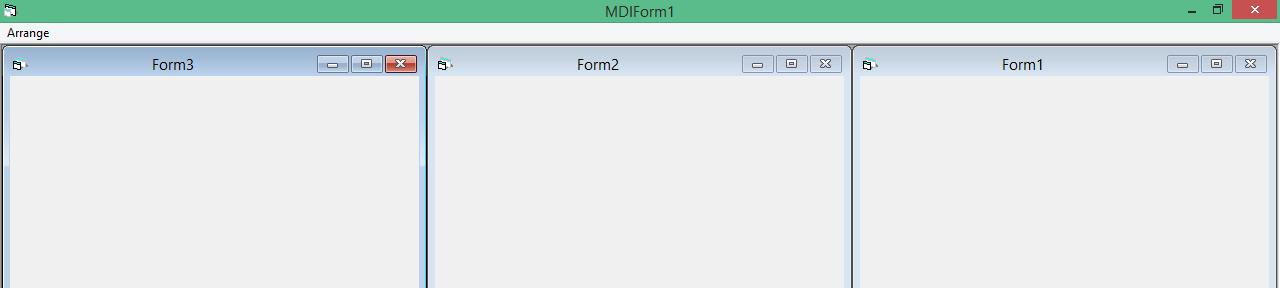
End Sub

**Output**

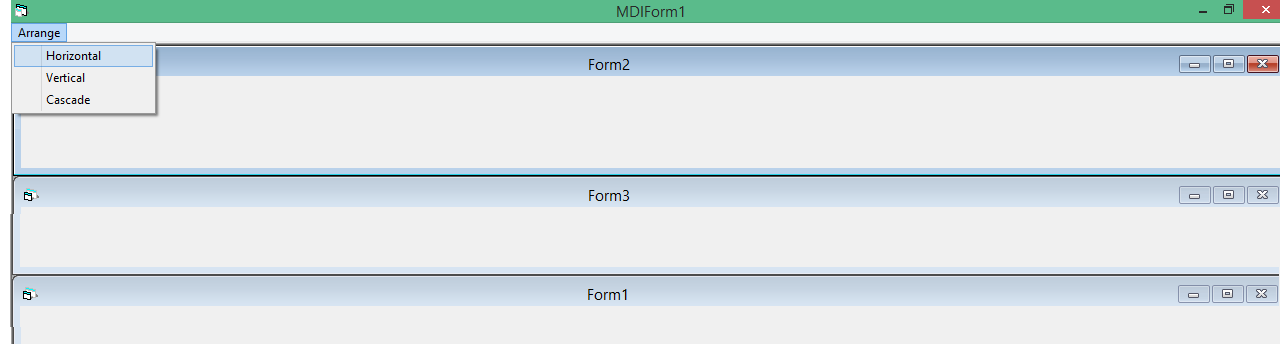
**Cascade Arrangement**

****

**Vertical Arrangement**

****

**Horizontal Arrangement**

****

**B2. VB Application: Change the Background Color of the form using Scroll Bars**

**Aim:** To develop a VB application to change the background color of the form using three scroll bars: Red, Green and Blue

**Procedure:** Design the form with three scroll bars, Text boxes, Clear and Exit button, use RGB function and assign that color to BackColor of the form

**Source Code**

Private Sub Command1\_Click()

'Clear Button

HScroll1.Value = 0

HScroll2.Value = 0

HScroll3.Value = 0

Text1.Text = ""

Text2.Text = ""

Text3.Text = ""

End Sub

Private Sub Command2\_Click()

'Exit Button

End

End Sub

Private Sub HScroll1\_Change()

Form1.BackColor = RGB(HScroll1.Value, HScroll2.Value, HScroll3.Value)

Text1.Text = HScroll1.Value

End Sub

Private Sub HScroll2\_Change()

Form1.BackColor = RGB(HScroll1.Value, HScroll2.Value, HScroll3.Value)

Text2.Text = HScroll2.Value

End Sub

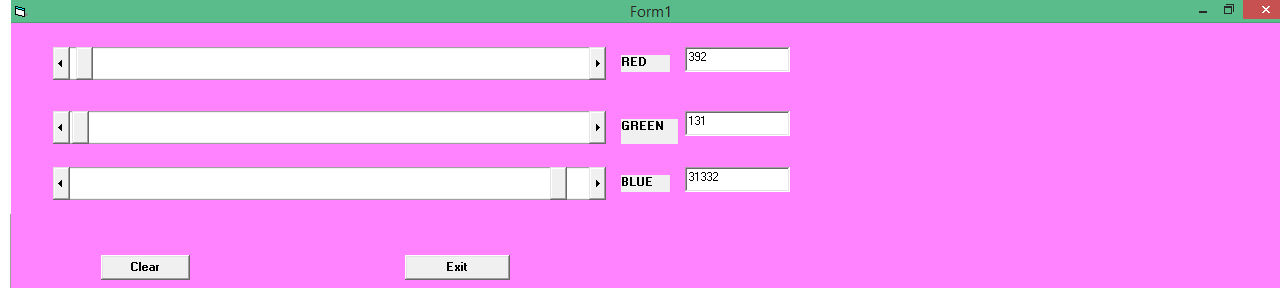
Private Sub HScroll3\_Change()

Form1.BackColor = RGB(HScroll1.Value, HScroll2.Value, HScroll3.Value)

Text3.Text = HScroll3.Value

End Sub

**Output**

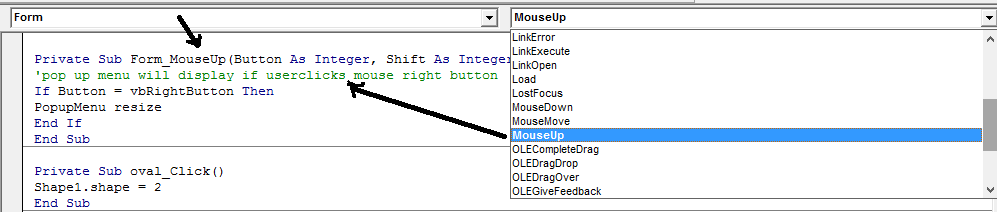
****

**B3. VB Application: Draw shapes using mouse with Pull Down/ Pop Up Menu**

**Aim:** Create a VB Application to display different shapes using pull down and popup menu

**Procedure:** Create a menu Shape (for different shapes) and resize (for zoom/un zoom)

Select object Form with Event MouseUP



If Button is mouse right button call the menu resize. That is if user right click the mouse button the resize popup menu will display near to mouse cursor.

Add the necessary functionalities to different shapes and resize

**Source Code**

Private Sub Form\_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

'pop up menu : resize will display if user clicks mouse right button

If Button = vbRightButton Then

PopupMenu resize

End If

End Sub

Private Sub rectangle\_Click()

‘0 will display rectangle

Shape1.shape = 0

End Sub

Private Sub square\_Click()

‘1 will display square

Shape1.shape = 1

End Sub

Private Sub oval\_Click()

‘ 2 will display oval

Shape1.shape = 2

End Sub

Private Sub circle\_Click()

‘ 3 will display circle

Shape1.shape = 3

End Sub

Private Sub roundedrectangle\_Click()

‘ 4 will display rounded rectangle

Shape1.shape = 4

End Sub

Private Sub roundedsquare\_Click()

‘ 5 will display rounded square

Shape1.shape = 5

End Sub

Private Sub unzoom\_Click()

Shape1.Height = Shape1.Height - 1000

Shape1.Width = Shape1.Width - 1000

End Sub

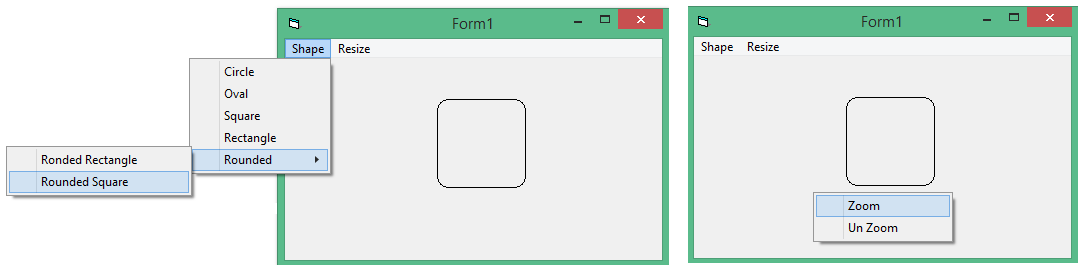
Private Sub zoom\_Click()

Shape1.Height = Shape1.Height + 1000

Shape1.Width = Shape1.Width + 1000

End Sub

**Output**



**B4. VB Application: Load an image using Drive List Box, Directory List Box, File List Box and Image control**

**Aim:** To develop a VB Application to load an image using Drive List Box, Directory List Box, File List Box and Image control

**Procedure:** Design the form with all controls, assign appropriately: drive to directory, directory to file, file to image control

**Source Code**

Private Sub Dir1\_Change()

File1.Path = Dir1.Path

End Sub

Private Sub Drive1\_Change()

Dir1.Path = Drive1.Drive

End Sub

Private Sub File1\_Click()

Dim img As String

If Right(File1.Path, 1) = "\" Then

img = File1.Path + File1.FileName

Else

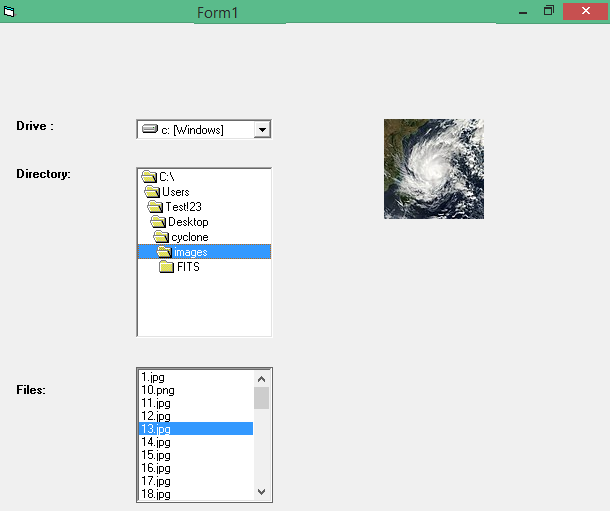
img = File1.Path + "\" + File1.FileName

End If

Image1.Picture = LoadPicture(img)

End Sub

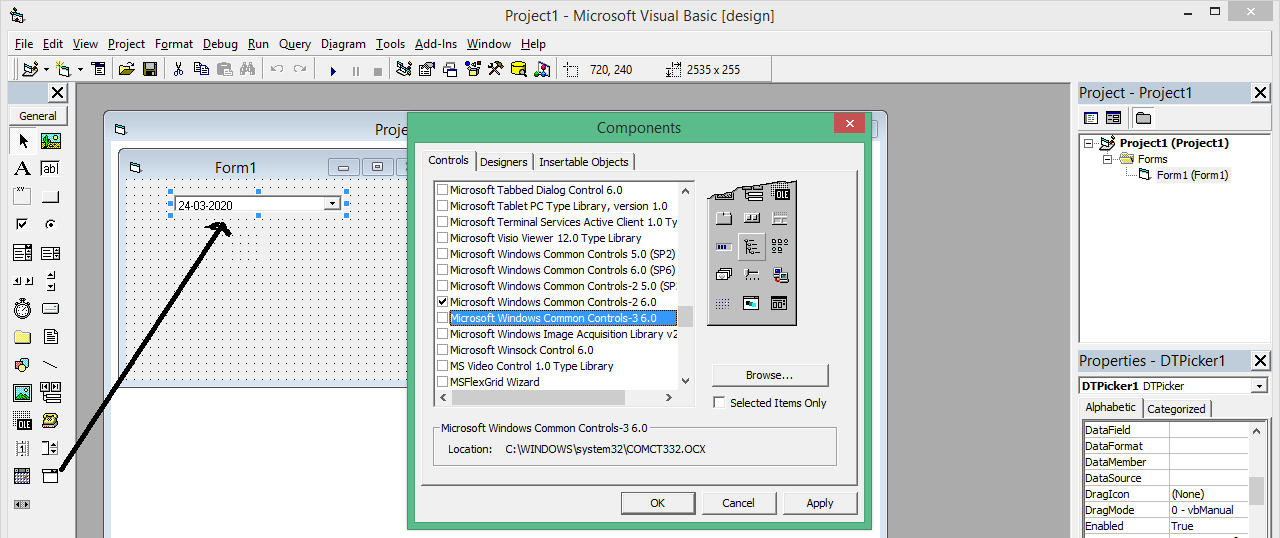
**Output**



**B5. VB Application: Calculate age using DTPicker Tool**

**Aim:** To design a VB Application to find age of a person using DTPicker Tool

**Procedure:** Design the form; Project > Components > Microsoft Windows Common Control 2.6.0 > Apply**;** Add DTPicker Tool to Form1, and add necessary controls



**Source Code**

Private Sub Command1\_Click()

' Calculate Age Button

Dim yy As Integer

Dim frombday As Integer

Dim frombmonth As Integer

Dim frombyear As Integer

Dim diff As Variant

If DTPicker1.Value > Date Then

MsgBox "Please Select Correct DOB"

Else

diff = DateDiff("d", DTPicker1.Value, Date)

' "d" is used for specify no of days between two date

frombyear = diff / 365

yy = diff Mod 365

frombmonth = yy / 31

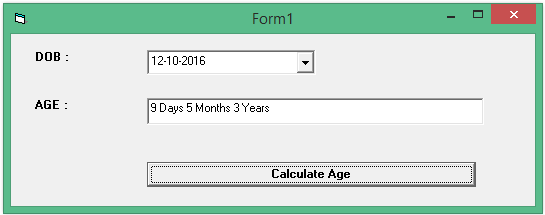
frombday = yy Mod 31

Text1.Text = frombday & " Days " & frombmonth & " Months " & frombyear & " Years "

End If

End Sub

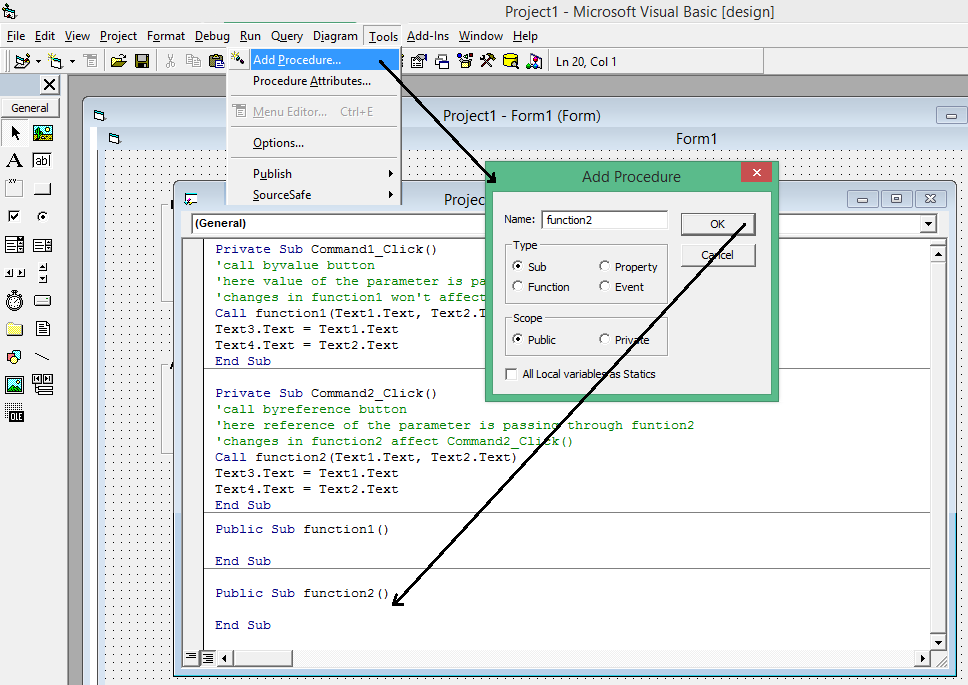
**Output**



**B6. VB application: Swap Two numbers using CallByValue() and CallByReference()**

**Aim:** To design a VB Application to swap two numbers using call by value and reference using subroutine or function

**Procedure:** Click View > Code > Click Tools menu > Add Procedue > add your subroutine/ function. Use **ByVal** key word for pass by value and **ByRef** keyword for pass by reference



**Source code**

Private Sub Command1\_Click()

'Call byvalue button

'Here value of the parameter is passing through funtion1

'Changes in function1 won't affect Command1\_Click()

Dim a As Integer

Dim b As Integer

a = Text1.Text

b = Text2.Text

Call function1(a, b)

Text3.Text = a

Text4.Text = b

End Sub

Private Sub Command2\_Click()

'Call byreference button

'Here reference of the parameter is passing through funtion2

'Changes in function2 affect Command2\_Click()

Dim a As Integer

Dim b As Integer

a = Text1.Text

b = Text2.Text

Call function2(a, b)

Text3.Text = a

Text4.Text = b

End Sub

Public Sub function1(ByVal x As Integer, ByVal y As Integer)

Dim t As Integer

t = x

x = y

y = t

End Sub

Public Sub function2(ByRef x As Integer, ByRef y As Integer)

Dim t As Integer

'Here variable a and x shares same memory address

‘Variable b and y shares same memory address

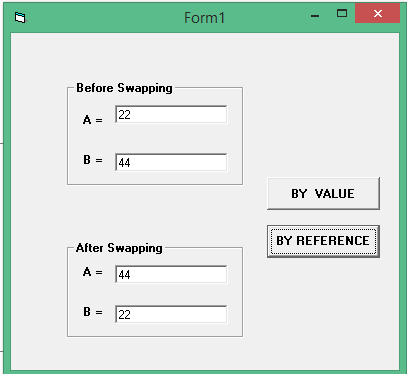
t = x

x = y

y = t

End Sub

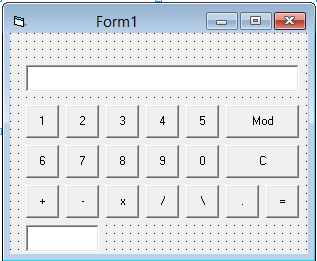
**Output**



**B7. VB Application: A Simple Calculator**

**Aim:** To design a calculator

**Procedure:** use text boxes one is invisible at run time and add all basic functionalities

****

Option Explicit

Public ch As Integer

Private Sub Command1\_Click()

' 1 button

Text1.Text = Text1.Text + "1"

End Sub

Private Sub Command10\_Click()

' 0 button

Text1.Text = Text1.Text + "0"

End Sub

Private Sub Command11\_Click()

' decimal point button

Text1.Text = Text1.Text + "."

End Sub

Private Sub Command12\_Click()

' C button

Text1.Text = ""

End Sub

Private Sub Command13\_Click()

' + button

Text2.Text = Val(Text1.Text)

Text1.Text = ""

ch = 1

End Sub

Private Sub Command14\_Click()

' - button

Text2.Text = Val(Text1.Text)

Text1.Text = ""

ch = 2

End Sub

Private Sub Command15\_Click()

' x button

Text2.Text = Val(Text1.Text)

Text1.Text = ""

ch = 3

End Sub

Private Sub Command16\_Click()

' / button

Text2.Text = Val(Text1.Text)

Text1.Text = ""

ch = 4

End Sub

Private Sub Command17\_Click()

' \ button used for integer division

Text2.Text = Val(Text1.Text)

Text1.Text = ""

ch = 5

End Sub

Private Sub Command18\_Click()

' mod button

Text2.Text = Val(Text1.Text)

Text1.Text = ""

ch = 6

End Sub

Private Sub Command19\_Click()

' = button

Select Case ch

Case 1

Text1.Text = Val(Text2.Text) + Val(Text1.Text)

Case 2

Text1.Text = Val(Text2.Text) - Val(Text1.Text)

Case 3

Text1.Text = Val(Text2.Text) \* Val(Text1.Text)

Case 4

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

Case 5

Text1.Text = Val(Text2.Text) \ Val(Text1.Text)

Case 6

Text1.Text = Val(Text2.Text) Mod Val(Text1.Text)

End Select

End Sub

Private Sub Command2\_Click()

' 2 button

Text1.Text = Text1.Text + "2"

End Sub

Private Sub Command3\_Click()

' 3 button

Text1.Text = Text1.Text + "3"

End Sub

Private Sub Command4\_Click()

' 4 button

Text1.Text = Text1.Text + "4"

End Sub

Private Sub Command5\_Click()

' 5 button

Text1.Text = Text1.Text + "5"

End Sub

Private Sub Command6\_Click()

' 6 button

Text1.Text = Text1.Text + "6"

End Sub

Private Sub Command7\_Click()

' 7 button

Text1.Text = Text1.Text + "7"

End Sub

Private Sub Command8\_Click()

' 8 button

Text1.Text = Text1.Text + "8"

End Sub

Private Sub Command9\_Click()

' 9 button

Text1.Text = Text1.Text + "9"

End Sub

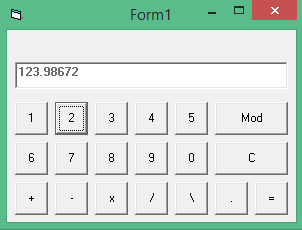
Private Sub Form\_Load()

Text1.Enabled = False

Text2.Visible = False

End Sub

**Output**



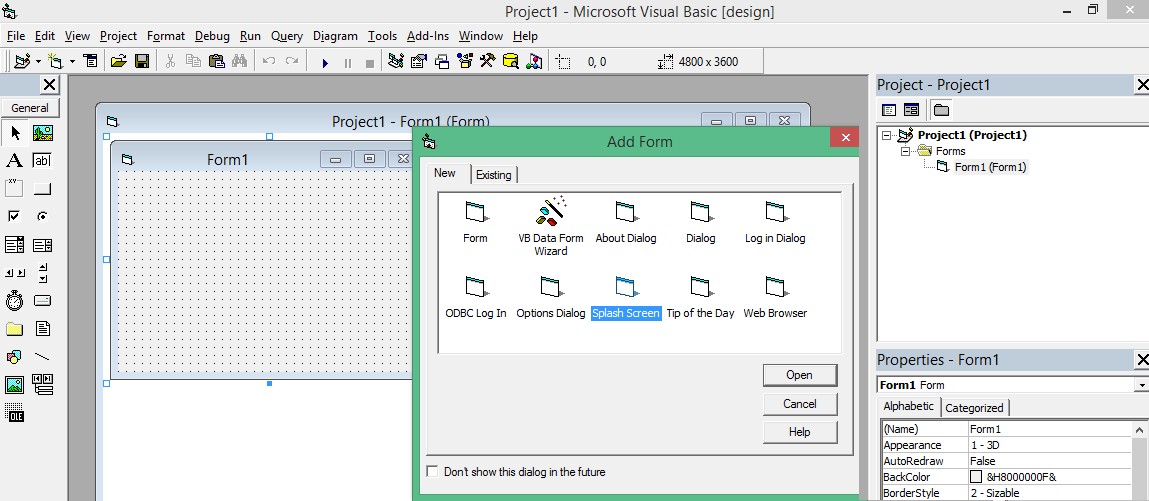
**B8. VB Application: Splash Screen with Progress Bar**

**Aim:** To develop splash screen with progress bar

**Procedure:**

Take the project: with form

Click Project Menu > Click Add Form> Select Splash Screen

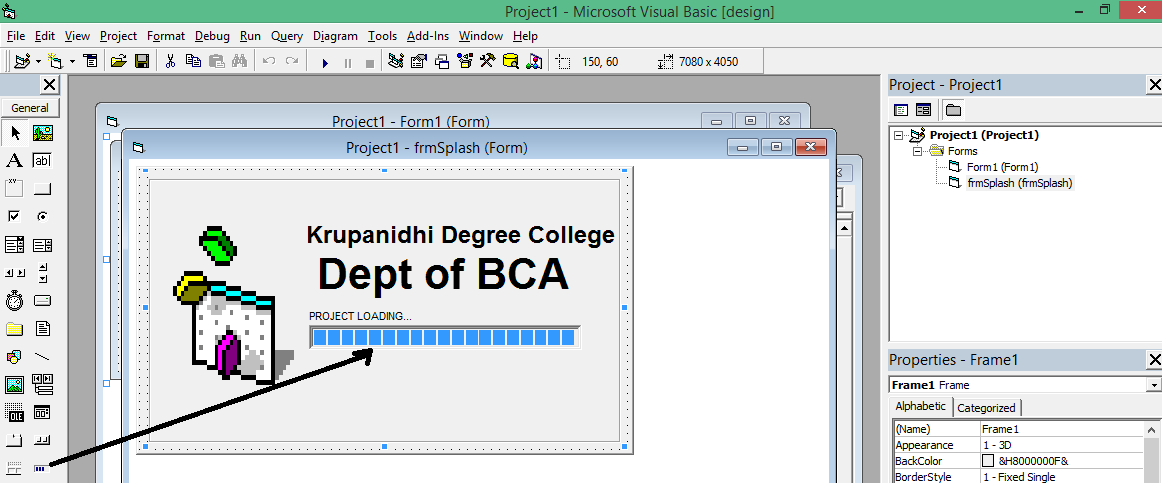


Design the Splash Screen

Click Project Menu > Components>

Select Microsoft Windows Common Dialog Control 6.0, Click OK

Choose Progress Bar from tool box, and Add to Splash Screen Form



Set Progress Bar Max value is 100 in Properties Window

Add Timer, Enable into False, with interval 250

Write the code in Timer() function

**Source Code**

Private Sub Form\_Load()

'This is splash Screen form

Timer1.Enabled = True

End Sub

Private Sub Timer1\_Timer()

ProgressBar1.Visible = True

ProgressBar1.Value = ProgressBar1.Value + 10

'Label1's caption is PROJECT LOADING

'Label2 attaching to label1

Label2.Caption = ProgressBar1.Value & "%"

If ProgressBar1.Value = ProgressBar1.Max Then

Form1.Show

Timer1.Enabled = False

End If

End Sub

Private Sub Command1\_Click()

‘This is a button End in Form1

End

End Sub

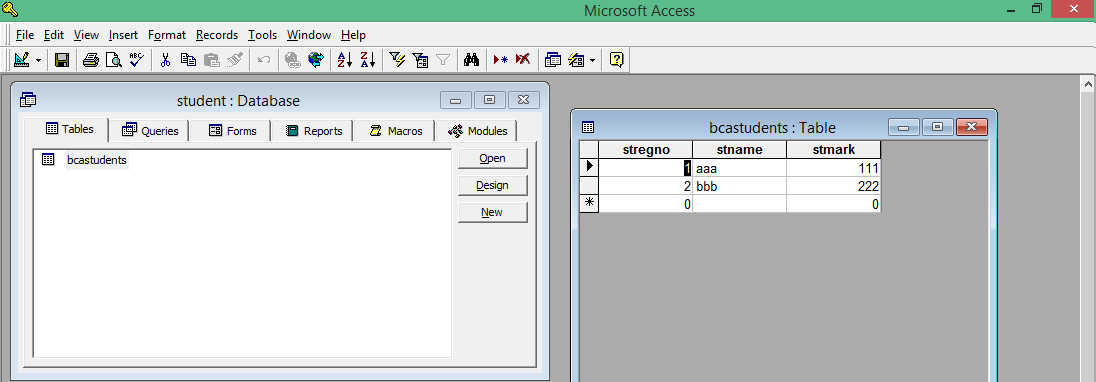
**Output**

****

**B9. VB Application: Student database with all options using DAO Control**

**Aim:** Develop a database application using DAO control

**Procedure:** Open **MS Access 97**, create database as student, table name as bcastudents with fields stregno, stname and stmark.



Design the VB Project with form: insert, delete, update, record moves buttons

Add project > References > Microsoft DAO 3.6 Object Library > OK

**Source Code**

Option Explicit

Dim db As Database

Dim rs As Recordset

Dim rs1 As Recordset

Private Sub Command1\_Click()

'ADD BUTTON

Dim s As String

'rs.AddNew

'rs("stno") = Val(Text1.Text)

'rs("stname") = Text2.Text

'rs("stmark") = Val(Text3.Text)

s = "INSERT INTO bcastudents VALUES (" & Val(Text1.Text) & ",'" & Text2.Text & "'," & Val(Text3.Text) & ");"

MsgBox s

db.Execute (s)

MsgBox "Data Inserted"

Text2.Text = ""

Text1.Text = ""

Text3.Text = ""

End Sub

Private Sub Command2\_Click()

'UPDATE BUTTON

Dim s As String

Dim reg, Name, mark

Dim d

d = InputBox("For Seeing the Record... Enter Regno")

s = "select \* from bcastudents where stregno = " & d

MsgBox s

Set rs1 = db.OpenRecordset(s)

On Error GoTo hh

Text1.Text = rs1.Fields(0)

Text2.Text = rs1.Fields(1)

Text3.Text = rs1.Fields(2)

If MsgBox("You want to Update this Record?", vbOKCancel) = vbOK Then

reg = InputBox("Update Regno Field")

Name = InputBox("Update Name Field")

mark = InputBox("Update Mark Field")

s = "update bcastudents set stregno = " & reg & ", stname= '" & Name & "', stmark=" & mark & " where stregno =" & Val(Text1.Text)

MsgBox s

db.Execute (s)

'rs1.Fields(0) = reg

'rs1.Fields(1) = Name

'rs1.Fields(2) = mark

'rs1.Update

MsgBox ("Record are Updated")

Text1.Text = ""

Text2.Text = ""

Text3.Text = ""

End If

Exit Sub

hh:

MsgBox "Not Found"

End Sub

Private Sub Command3\_Click()

'DELETE BUTTON

Dim s As String

Dim d

d = InputBox("Enter Regno")

s = "select \* from bcastudents where stregno = " & d

MsgBox s

Set rs1 = db.OpenRecordset(s)

On Error GoTo hh

Text1.Text = rs1.Fields(0)

Text2.Text = rs1.Fields(1)

Text3.Text = rs1.Fields(2)

If MsgBox("You want to Delete this Record?", vbOKCancel) = vbOK Then

rs1.Delete

Text1.Text = ""

Text2.Text = ""

Text3.Text = ""

End If

Exit Sub

hh:

MsgBox "Not Found"

End Sub

Private Sub Command4\_Click()

'FIRST BUTTON

rs.MoveFirst

Text1.Text = rs.Fields(0)

Text2.Text = rs.Fields(1)

Text3.Text = rs.Fields(2)

End Sub

Private Sub Command5\_Click()

'PREVIOUS BUTTON

rs.MovePrevious

If rs.BOF Then

Call Command4\_Click

Else

Text1.Text = rs.Fields(0)

Text2.Text = rs.Fields(1)

Text3.Text = rs.Fields(2)

End If

End Sub

Private Sub Command6\_Click()

'NEXT BUTTON

rs.MoveNext

If rs.EOF Then

Call Command7\_Click

Else

Text1.Text = rs.Fields(0)

Text2.Text = rs.Fields(1)

Text3.Text = rs.Fields(2)

End If

End Sub

Private Sub Command7\_Click()

'LAST BUTTON

rs.MoveLast

Text1.Text = rs.Fields(0)

Text2.Text = rs.Fields(1)

Text3.Text = rs.Fields(2)

End Sub

Private Sub Command9\_Click()

'SEARCH BUTTON

Dim s As String

Dim d

d = InputBox("Enter Regno")

s = "select \* from bcastudents where stregno = " & d

MsgBox s

Set rs1 = db.OpenRecordset(s)

On Error GoTo hh

Text1.Text = rs1.Fields(0)

Text2.Text = rs1.Fields(1)

Text3.Text = rs1.Fields(2)

Exit Sub

hh:

MsgBox "Not Found"

End Sub

Private Sub Form\_Load()

'Set db = CreateDatabase(App.Path + "student.mdb", dbLangGeneral, dbEncrypt)

Set db = OpenDatabase("C:\ds9\student.mdb")

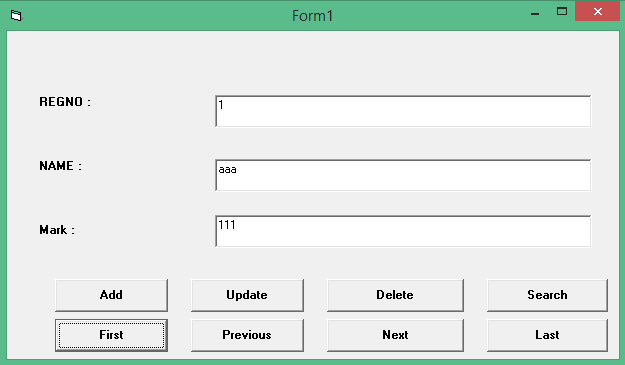
'db.Execute ("create table bcastudent(regno number, name varchar(10), mark number)")

Set rs = db.OpenRecordset("bcastudents")

'MsgBox ("Recordset Created")

End Sub

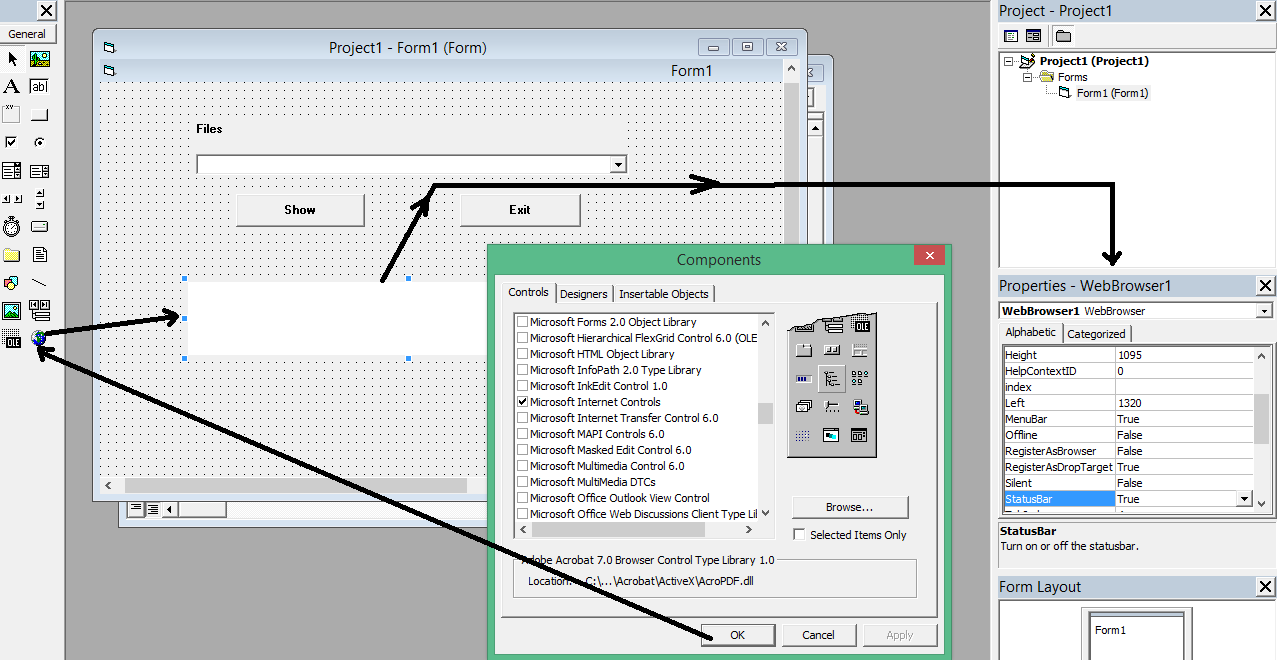
**Output**

****

**B10. VB Application: Load files like HTML, MP4,.png etc in form using Web Control**

**Aim:** design a VB Application to load files like HTML, MP4, .png etc using Web control

**Procedure:** Add Project > Components > Microsoft Internet Controls > OK

****

**Source Code**

Private Sub Command1\_Click()

' Show Button

If Combo1.Text <> "" Then

WebBrowser1.Navigate2 (Combo1.Text)

Else

MsgBox "No file found"

End If

End Sub

Private Sub Command2\_Click()

'Exit Button

End

End Sub

Private Sub Form\_Load()

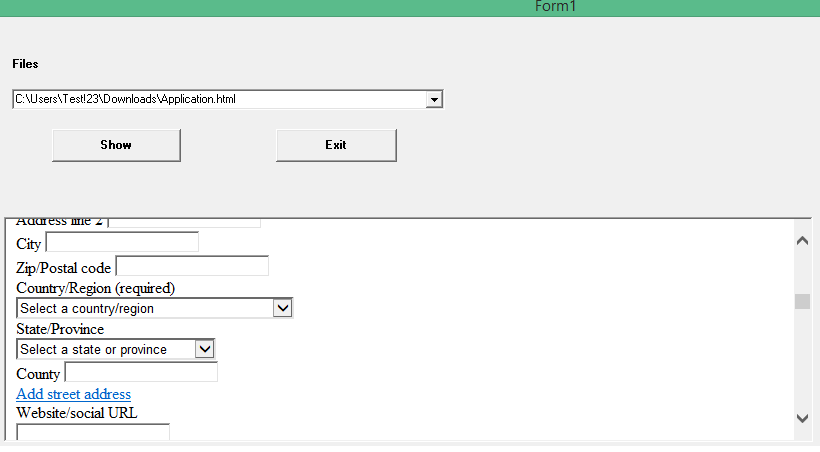
Combo1.AddItem ("C:\Users\Test!23\Desktop\Krupa.html")

Combo1.AddItem ("C:\Users\Test!23\Desktop\Art.html")

Combo1.AddItem ("C:\Users\Test!23\Downloads\Application.html")

End Sub

**Output**



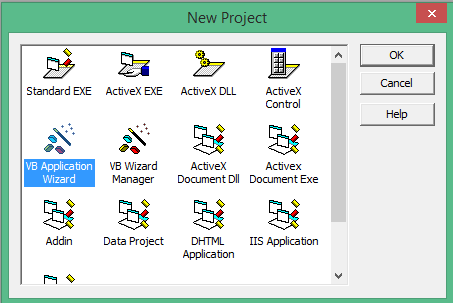
**B11 VB Application: Student Database using VB Wizard**

**Aim:** Create a VB Wizard database application for students

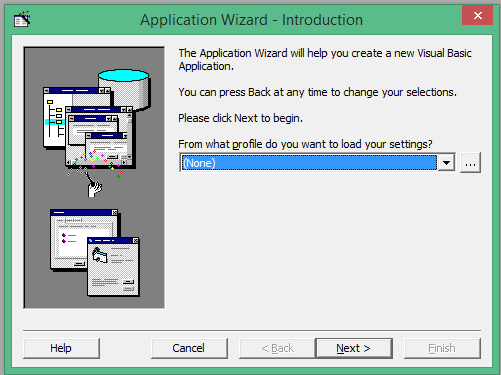
**Procedure:**

Take New Project> Select VB Application Wizard (Instead of Standard EXE)

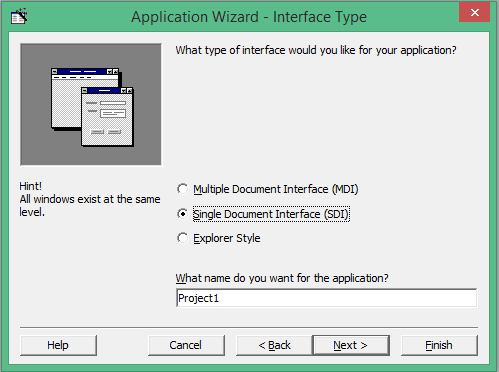
**REMEMBER you made a slight mistake to so when you add a record then the next field gets erased so before adding a new record you first click the add button then you enter the first records field details**

****

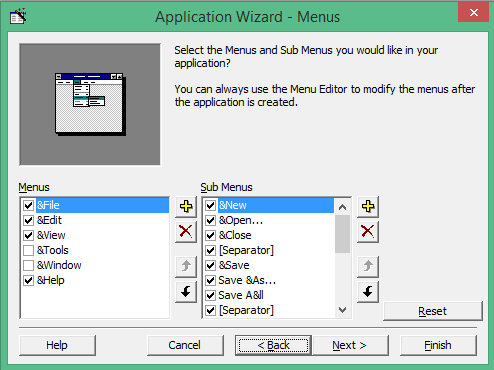
**Click OK**

****

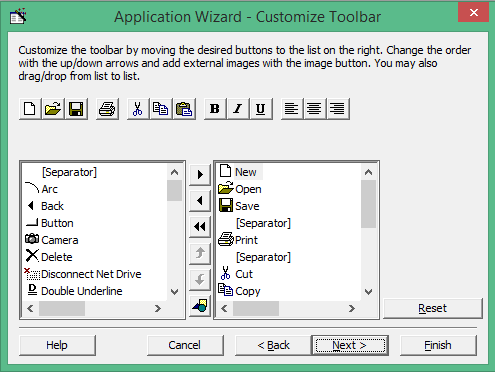
**Click Next**

****

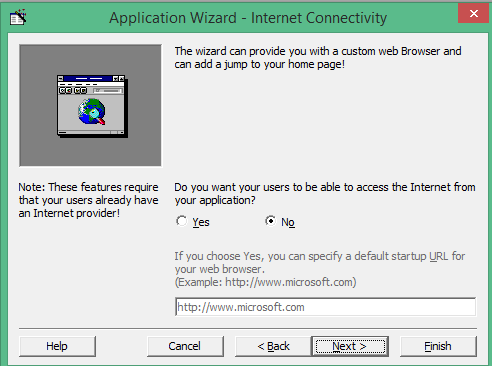
**Select SDI and Click Next**

****

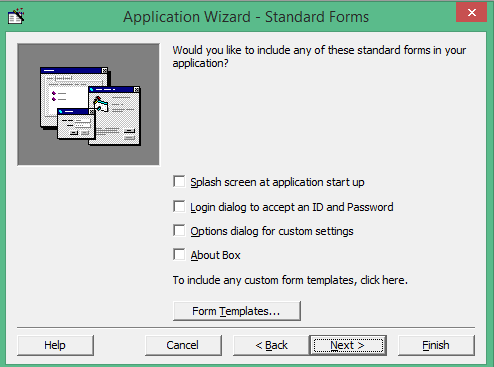
**Click Next**

****

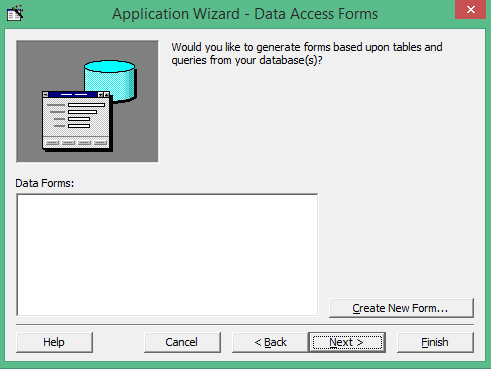
**Click Next**

****

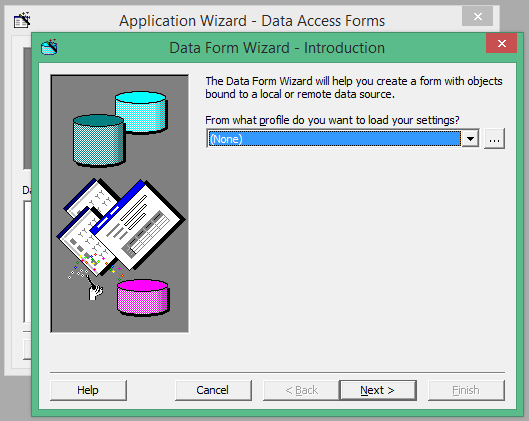
**Select No and Click Next**

****

**Not Choosing and Click Next**

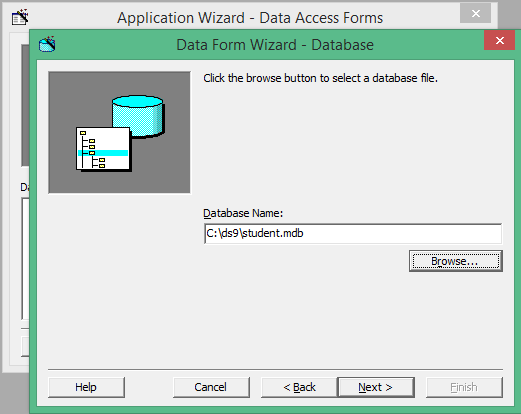
****

**Click Create New Form**

****

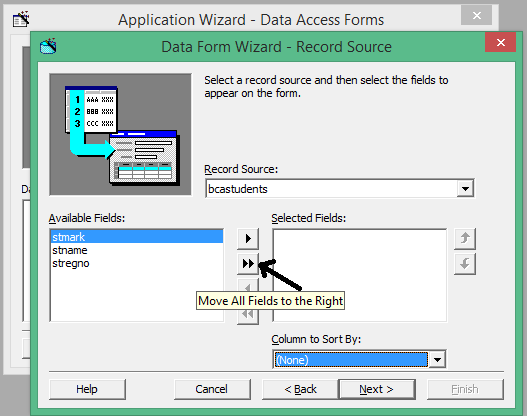
**Click Next and click browse button and**

**Choose Student database student.mdb (MS Access 97)**

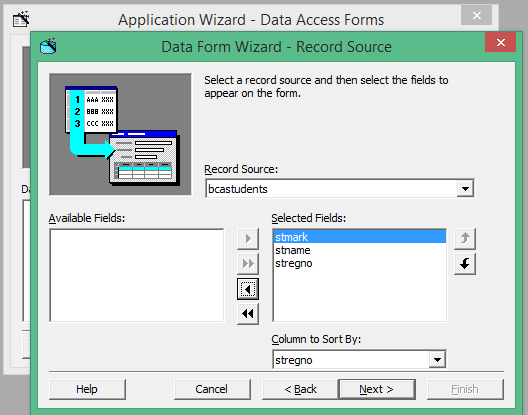
****

**Select the record source bcastudents**

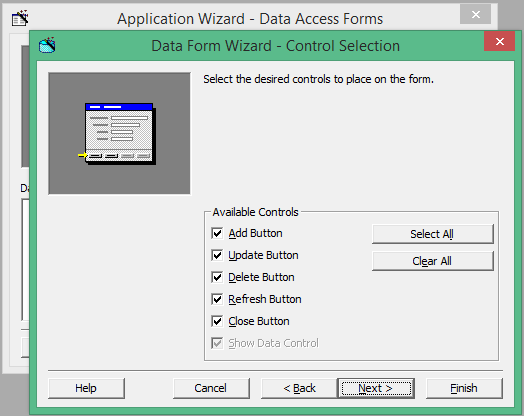
**And Click ‘Move all fields to Right’ Button ()**

****

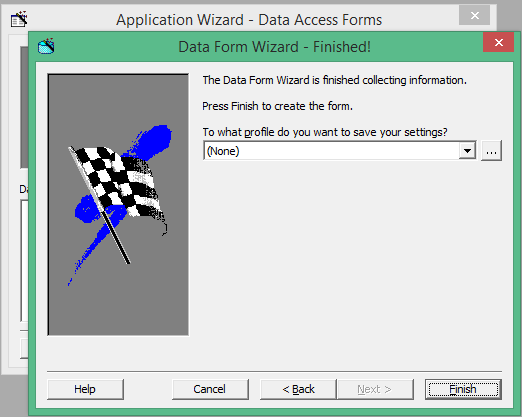
**Select Column Sort by stregno**

****

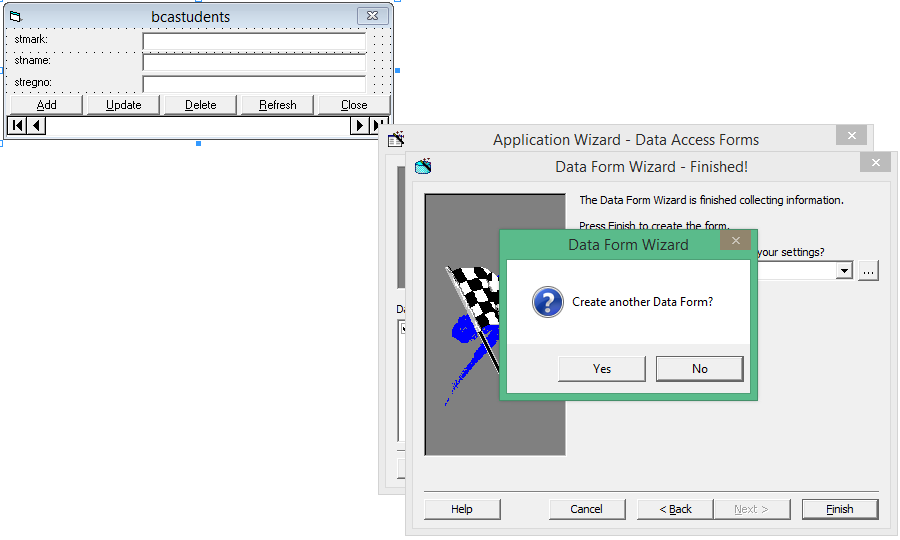
**Click Next**

****

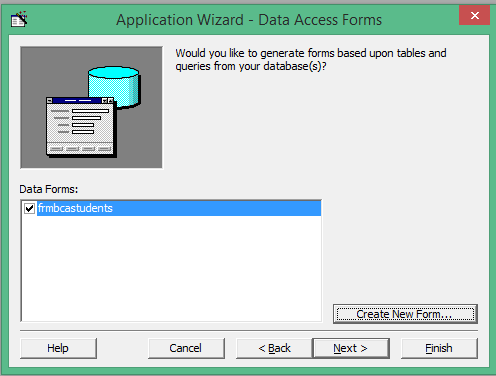
**Click Next**

****

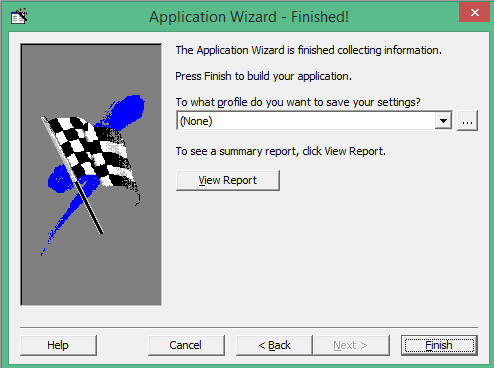
**Click Finish button**

****

**Click No and Continue**

****

**Click Next**

****

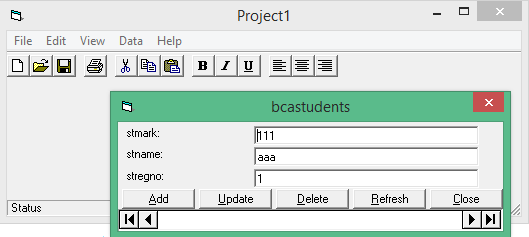
Click Finish Button

Run the application

Error lines make it as Comment lines by putting ‘

Your project will ready with database CRUD operation

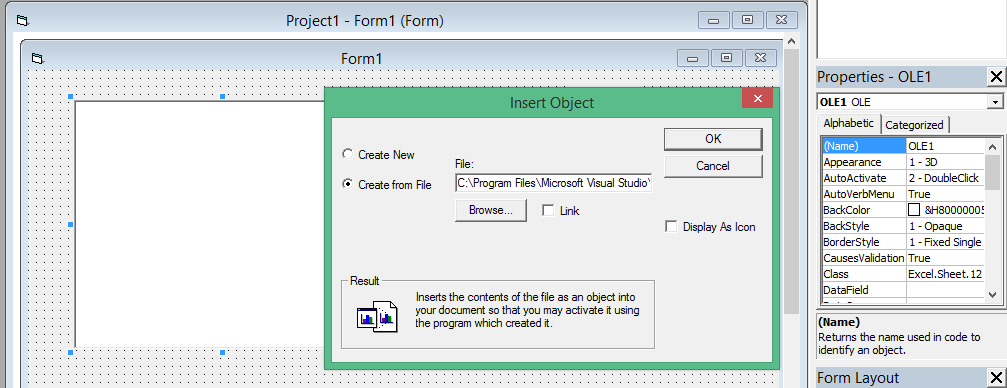
**Output**

****

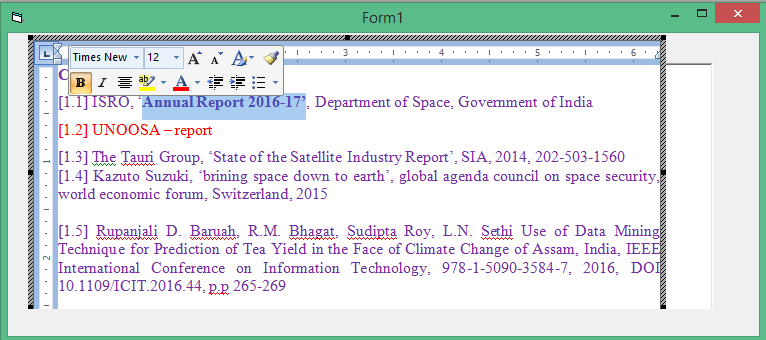
**B12. VB Application: MS Word using OLE Control**

**Aim:** Develop a VB Application using OLE Control for MS Word

**Procedure:** Drag and Drop OLE Tool to Form and follow the procedure

****

1. In Insert Object Dialog box select Create From File radio button.  
   2. Click on Browse button and select a document file .  
   3. Click on Ok  
   4. An object is embedded into OLE Container control and a part of document is displayed.  
   5. Run the project using F5.  
   6. Double click on OLE Container control. This action will invoke MS-Word and run it in OLE Container control. When OLE Server runs in OLE Client, it is called as In-Place Activation.  
   7. Make necessary changes using MS-Word.  
   8. Press ESC key to come out of In-Place activation.



**B13 VB Application: Design Layout Using ActiveX Control**

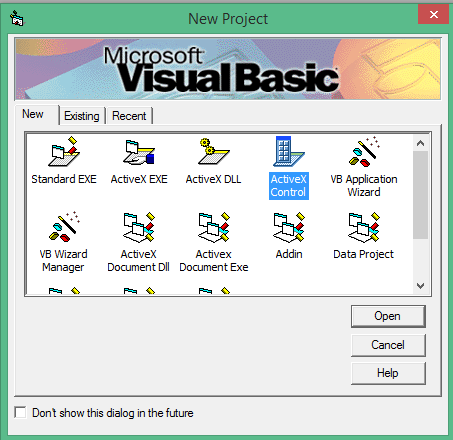
**Aim: To Develop Design using Active X Control**

**Procedure:**

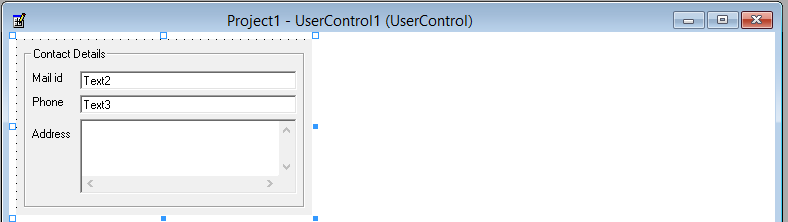
**First open visual basic 6 > run as administrator # remember this was a mistake when**

**you executed**

**Click New Project > select Active X Control**

****

**Design the form NEATLY**

****

Click file menu> **Make contact.ocx** in your folder

Save all forms to folder (user control and project ) and close it

We can import this format to any project

Take new project >Standard EXE

Click project > Components > click Browse button> select contact.ocx>apply> OK

Active X will be available in Tool Box, Drag and drop to your form

**Output**

Active X component added to form

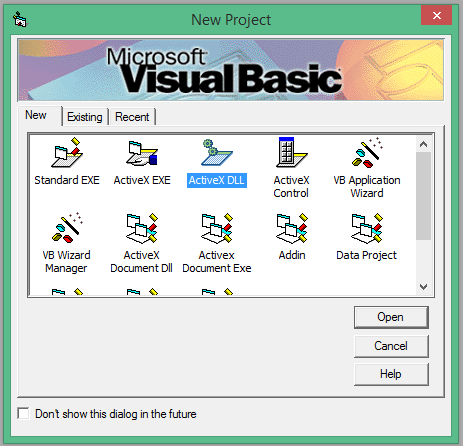
**B14. VB Application: Find Cube of a Number using ActiveX DLL**

**Aim**: To create Active X DLL to find cube of a number

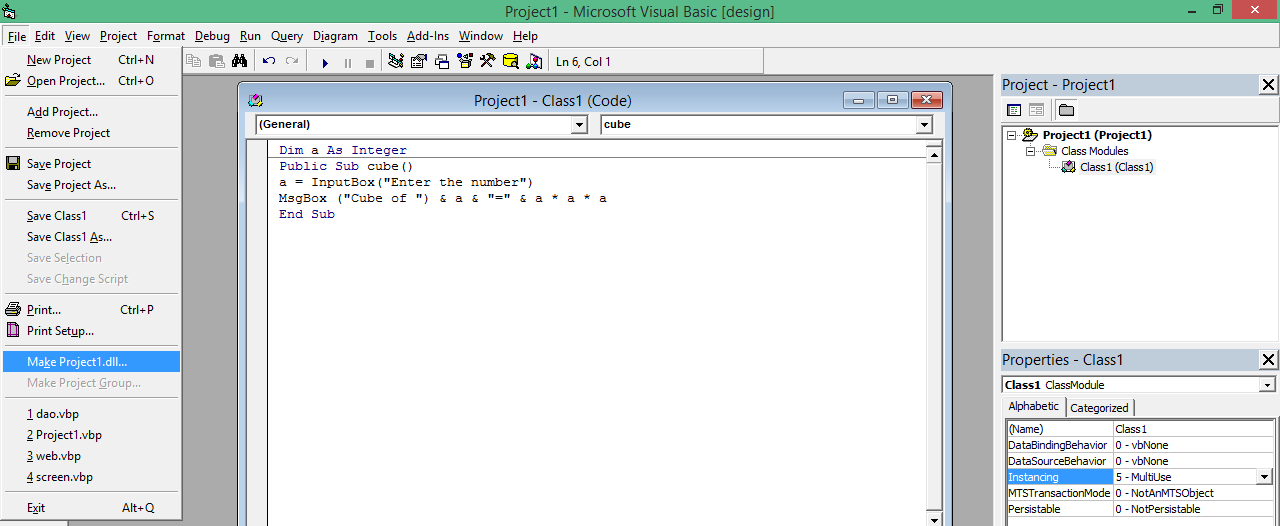
**Procedure:**

DLL are the references used in project to perform particular functionality

Choose Active X DLL (don’t Choose Standard EXE)



**Type cube program and click file menu, click Make Project1.dll.**

****

Dim a As Integer

Public Sub cube()

a = InputBox("Enter the number")

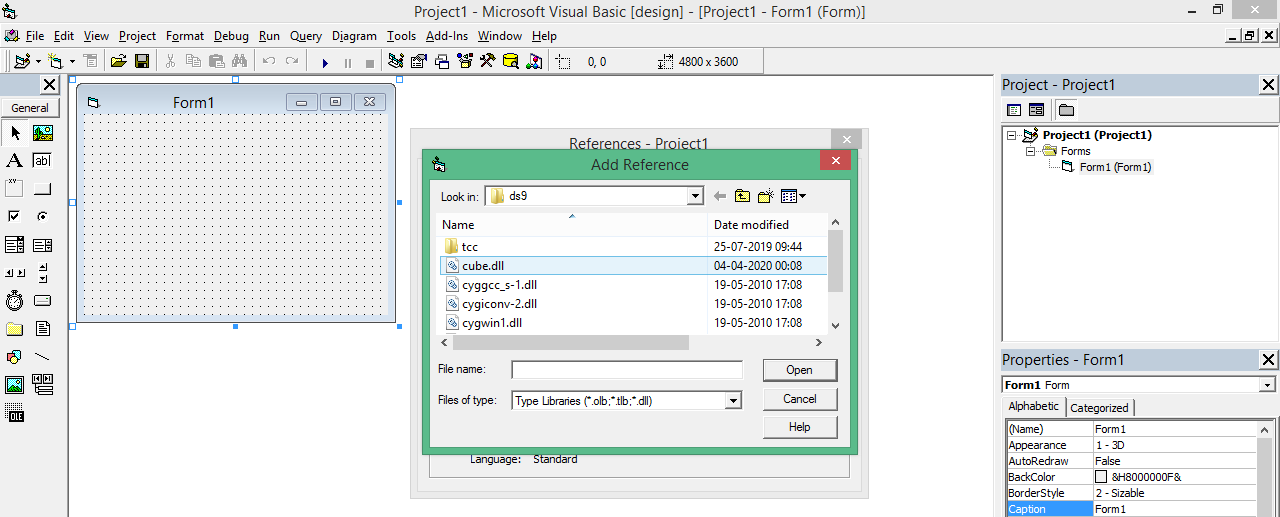
MsgBox ("Cube of ") & a & "=" & a \* a \* a

End Sub

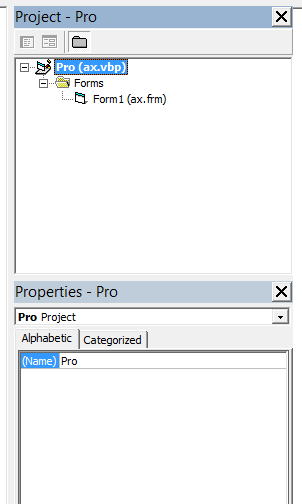
**Save the dll object (name it as cube), save the class file and project**

**Now create New Project and click Standard EXE**

**Click Project > References > click Browse Button > select your dll object (cube.dll)**

****

**It will show name conflict error, change the name of the Project as pro**

****

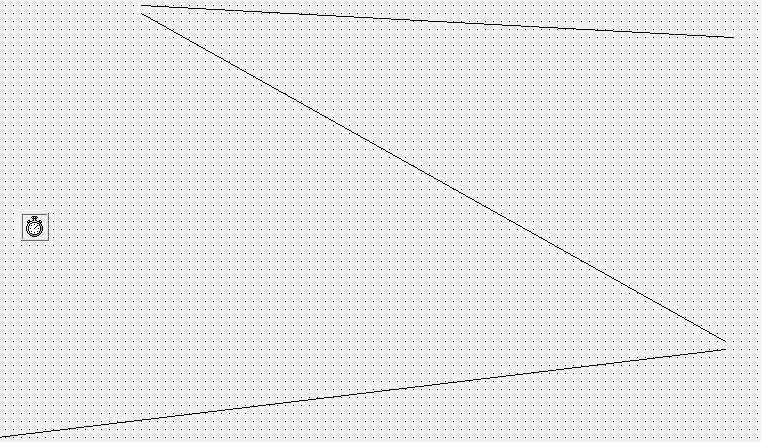
**Click Project > References > click Browse Button > select your dll object (cube.dll)**

**Any object call .cube dll**

**B15 VB Application: Animation Using Line Control**

**Aim:** Animate line control

**Procedure:** Drag and drop timer control, line control to the form



**Source Code**

Dim i As Integer

Private Sub Form\_Load()

Timer1.Enabled = True

End Sub

Private Sub Timer1\_Timer()

If Line1.X2 > 40000 Then

Line1.X2 = 0

Line1.Y2 = 0

End If

Line1.BorderWidth = 17

Line1.BorderColor = vbRed

Line1.X2 = Line1.X2 + 40

Line1.Y2 = Line1.Y2 - 30

Line2.BorderWidth = 17

Line2.BorderColor = vbGreen

Line2.X2 = Line1.X2

Line2.Y2 = Line1.Y2

Line3.BorderWidth = 17

Line3.BorderColor = vbBlue

Line3.X2 = Line2.X2

Line3.Y2 = Line2.Y2

End Sub

**Output**

