

# RichTextBox

....

-

-:

**Ctrl+T -1**

**Microsoft Rich Textbox -2**

**Control 5.0**

**Browse -3**

**RICHTX32.OCX -4**

**RICHTX32.OCX**

-:

**Name**

**Common Name Rloai**

**CLoai Dialog**

\*\*\*\*\*

\*\*\*\*\*

.....

**SelText -1**

\*\*\*\*\*

\*\*\*\*\*

-:

**RichText. SelText=0**

-:

**CommandButton**

**Private Sub Command1\_Click()**

**RLoai.SelText = 0**

**End Sub**

-:

**Private Sub Command1\_Click()**

**RLoai.SelText = " "**

**End Sub**

\*\*\*\*\*

\*\*\*\*\*

**SelColor -2**

**&H000000FF&RLoai.SelColor =**

-:

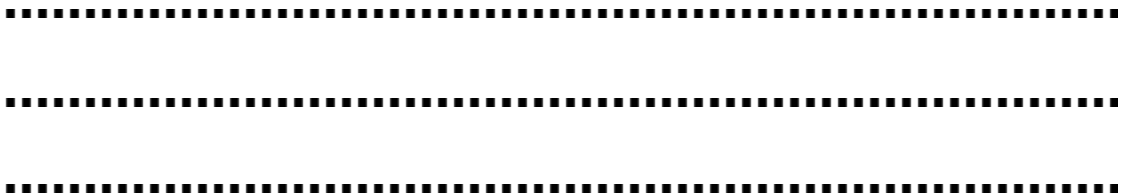
**Ctrl+T -1**

**Microsoft Common Dialog** -2  
**6.0**

**CommandButton**

-:

```
Private Sub Command1_Click()  
    CLoai.ShowColor  
    RLoai.SelColor = CLoai.Color  
End Sub
```



-3

-:

```
Private Sub Command1_Click()  
    CLoai.Flags = cdlCFBoth  
    CLoai.ShowFont  
    With RLoai  
        .SelFontName = CLoai.FontName  
        .SelFontSize = CLoai.FontSize  
        .SelBold = CLoai.FontBold  
        .SelItalic = CLoai.FontItalic  
        .SelStrikeThru = CLoai.FontStrikethru  
        .SelUnderline = CLoai.FontUnderline  
    End With
```

**End Sub**

**SelFontSize**

.....  
.....  
**filename**

**SaveFile**

**-4**

**-:**

**-1**

**Loai.rtf**

**filename**

**-:**

**Private Sub Form\_Load()**

**Dim lo**

**lo = App.Path**

**Dim lm As String**

**lm = lo & "/" & "Loai.rtf"**

**RLoai.filename = lm**

**End Sub**

**RichTextBox**  
.....

```
SaveFile
Dim lo
lo = App.Path
Dim lm As String
lm = lo & "/" & "Loai.rtf"
RLoai.SaveFile lm, 0
Rtf 0
Text 1
```

---

---

```
Private Sub Command1_Click()
Clipboard.SetText RLoai SelText
End Sub
```

---

```
Private Sub Command1_Click()
Clipboard.GetText (vbCFRTF)
End Sub
```

---

```
Private Sub Command11_Click()
```

```
RLoai.SelText =  
Clipboard.GetText(vbCFRTF)  
End Sub
```

**-:**

**Option Explicit**

```
Private Sub Command1_Click()  
    Dim lpobj As OLEObject  
    Dim szfilename As String  
    szfilename = "C:\Loai.bmp"  
    Set lpobj = Rloai.OLEObjects.Add(, ,  
        szfilename)  
    lpobj.DisplayType = rtfDisplayContent  
End Sub
```

```
Private Sub Form_Click()  
    Rloai.SelStart = 3  
    Rloai.SelText = "[          ]"  
End Sub
```

```
Private Sub Form_Load()  
    Me.Rloai.Text = "          "  
End Sub
```

.....

```
Private Sub RichTextBox1_DblClick()  
    With CommonDialog1  
        .Filter = ("bitmap (*.bmp)|*.bmp")  
        .ShowOpen  
    End With  
    RichTextBox1.OLEObjects. _  
Add , , CommonDialog1.filename  
End Sub
```

## Rich Textbox

\*\*\*\*\*

1

RtfArrow

-:

```
Private Sub Command1_Click()  
    Rloai.MousePointer = rtfArrow  
End Sub
```

**RtfArrowHourglass  
RtfArrowQuestion  
RtfCross**

**\*\*\*\*\***

**rtfCustom**

**Rloai.MouseIcon = LoadPicture("C:\Loai.ico")  
Rloai.MousePointer = rtfCustom**

**rtfDefault=0**

**rtfArrowQuestion=14**

**rtfHourglass=11**

**rtfIBeam=3**

**rtfIcon=4**

**rtfNoDrop=12**

**rtfSize=5**

**rtfSizeAll=15**

**rtfSizeEW=9**

**rtfSizeNESW=6**

**rtfSizeNS=7**

**rtfSizeNWSE=8**

**rtfUpArrow=10**

.....



.....

.....

\*\*\*\*\*

## **ScrollBars**

\*\*\*\*\*

**rtfBoth=3**  
**rtfHorizontal=1**  
**rtfNone=0**  
**rtfVertical=2**

\*\*\*\*\*

-:

-:

.....

**ScrollBars= rtfBoth**

.....

.....

.....

**SelAlignment**

.....

**rtfCenter=2**

**rtfLeft=0**

**rtfRight=1**

**3**

**CommadBox**

**Private Sub Command1\_Click()**

**Rloai.SelAlignment = 0**

**End Sub**

**Private Sub Command2\_Click()**

**Rloai.SelAlignment = 1**

**End Sub**

**Private Sub Command3\_Click()**

**Rloai.SelAlignment = 2**

**End Sub**

.....  
**ClipboardConstants**  
.....

.....  
\*\*\*\*\*

**rtfCFBitmap=2**

**BMP**

**rtfCFDIB=8**  
.....

**rtfCFEMetafile=14** **EMF**  
**rtfCFFiles=15**  
**rtfCFMetafile=3** **wmf**  
**rtfCFPalette=9**

**rtfCFRTF=-16639** **RTF**  
.....  
**rtfCFText=1** **TXT**  
.....

**vbCFLink= &HBF00**

.....  
\*\*\*\*\*

**RichTextBox**

**rtfInvalidProcedureCall=5**

**rtfOutOfMemory=7**

**rtfPathFileAccessError=75**  
.....

**rtfInvalidFileFormat=321**  
**Invalid file format**

**rtfInvalidPropertyValue=380**

**rtfInvalidPropertyArrayIndex=381**

**rtfSetNotSupported=383**

**rtfSetNotPermitted=387**

**rtfGetNotSupported=394**

**rtfInvalidObjectUse=425**

**rtfDataObjectLocked=672**

**OEL**

**rtfExpectedAnArgument=673**

**rtfInvalidCharPosition=32000**



**rtfInvalidHdc=32001**

**rtfCannotLoadFile=32002**

**rtfInvalidKeyName=32005**

**rtfInvalidClassName=32006**

**OLE**

**rtfKeyNotFound=32007**

**rtfOLESourceRquired=32008**

**rtfNonUniqueKey=32009**

**rtfInvalidObject=32010**

**rtfProtected=32011**

**rtfOleCreate=32012**

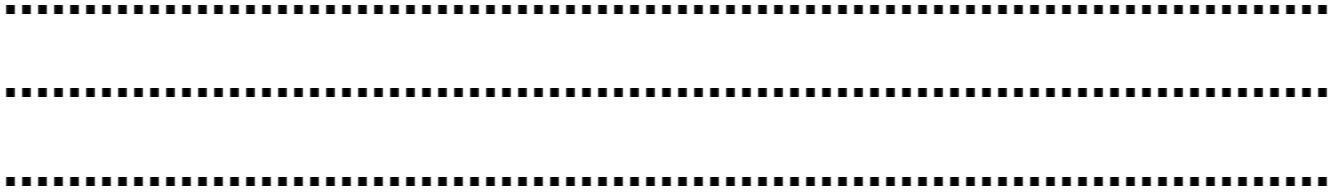
**rtfOleServer=32013**



.....  
.....  
.....  
.....  
**RtfDisplay**  
**RtfDisplayContent=0**  
**RtfDisplayIcon=1**

.....  
.....  
.....  
.....  
**DragOverConstants**

**RtfEnter=0**  
**RtfLeave=1**  
**RtfOver=2**  
.....



**BorderStyleConstants**  
**RtfFixedSingle=1**  
**RtfNoBorder=0**

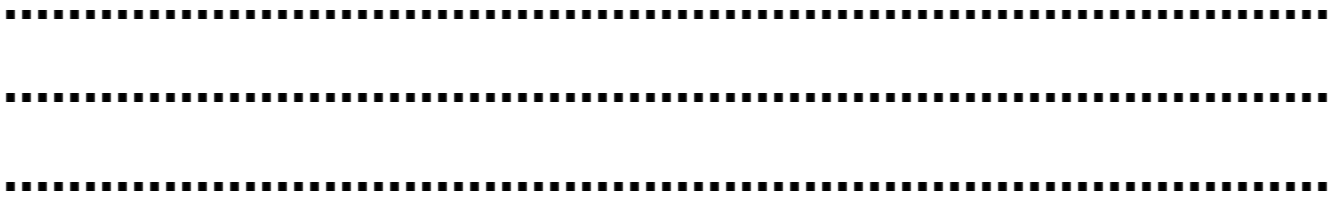
**AppearanceConstants**

A horizontal dotted line is located at the bottom of the page, positioned directly below the text "AppearanceConstants".

**Flat**

**RtfFlat=0**

**RtfThreeD=1**



**FindConstants**

**RtfMatchCase=4**

**RtfNoHighlight=8**

**RtfWholeWord=2**

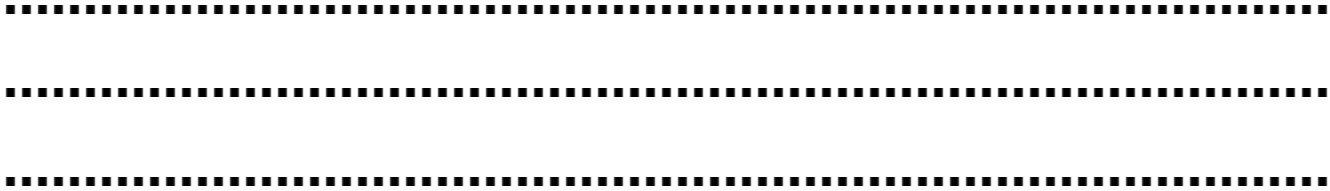
**OLEDragConstants**

**RtfOLEDragAutomatic=1**

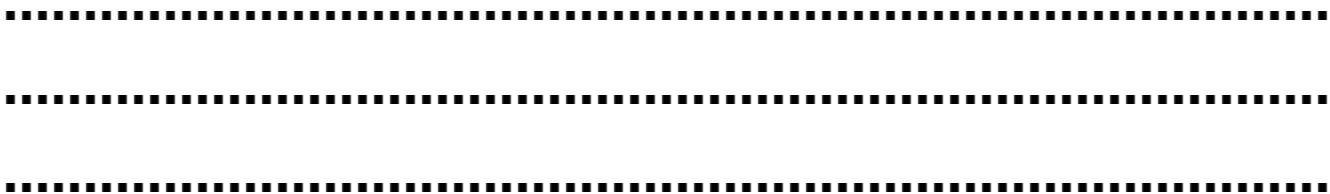




**RtfOLEDragManual=0**

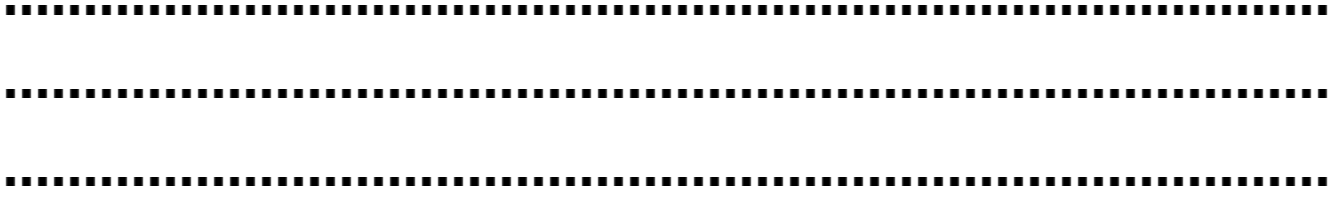


**OLEDropConstants**  
**RtfOLEDropAutomatic=2**  
**RtfOLEDropManual=1**  
**RtfOLEDropNone=0**



**OLEDropEffectConstants**  
**RtfOLEDropEffectCopy=1**  
**RtfOLEDropEffectMove=2**  
**RtfOLEDropEffectNone=0**  
**RtfOLEDropEffectScroll=-2147483648**





**LoadSaveConstants**

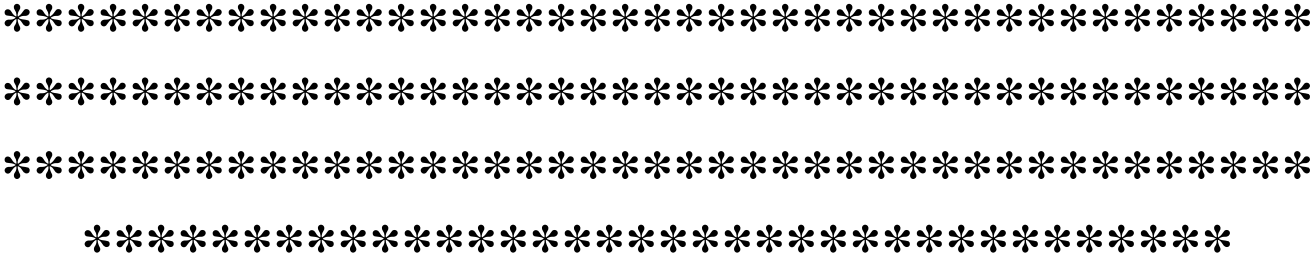
**RtfRTF=0**

**RtfText=1**



```
Dim lo  
lo = App.Path  
Dim lm As String  
lm = lo & "/" & "Loai.rtf"  
RLoai.SaveFile lm, 0
```

**1**



**2**



# DataObject

## Clear

Sub Clear()

.....  
.....  
.....

## GetData

DataObject. GetData

Function GetData(sFormat As Integer)

\*\*\*\*\*

## GetFormat

.....

**Function GetFormat(sFormat As Integer) As  
Boolean**

**SetData**

**Function GetFormat(sFormat As Integer) As  
Boolean**

.....  
.....  
.....

**Files**

.....  
**VbCFFiles**

.....

.....  
**DataObjectFiles**

.....

.....  
**Add**

**Sub Add(bstrFilename As String, [vIndex])**

**vbCFFiles**

---

**Clear  
()Sub Clear**

---

**vbCFFiles**

---

**Remove  
(Sub Remove(vIndex**

---

**VbCFFiles**

---

\*\*\*\*\*

**Count**

**Property Count As Long**

---

## **VbCFFiles**

---

**Item**

**Property Item(Index As Long) As  
String**

---

## **VbCFFiles**

---

**OLEObject**

---

**Class**

---

**Property Class As String**

---

---

---

## **DisplayType**

---

### **Property DisplayType As DisplayTypeConstants**

---

---

---

## **Index**

---

### **Property Index As Long**

---

---

---

## **Key**

---

### **Property Key As String**

---

## **ObjectVerbFlags**

---

### **Property ObjectVerbFlags As Variant**

.....  
.....  
**ObjectVerbs**  
.....

**Property ObjectVerbs As Variant**

.....  
.....  
**ObjectVerbsCount**  
.....

**Property ObjectVerbsCount As Long**

.....  
.....  
**DoVerb**  
.....

**([Sub DoVerb([verb**  
.....  
.....  
.....



## **FetchVerbs**

---

**Function FetchVerbs() As Long**

---

### **Class OLEObjects**

#### **Add**

**Function Add([Index], [Key], [source],  
[objclass]) As OLEObject**

#### **Clear**

**()Sub Clear**

**(Sub Remove(Item**

**2**

**RichTextBox**

---

# Appearance

## Property Appearance As AppearanceConstants

\*\*\*\*\*

## AutoVerbMenu

### Property AutoVerbMenu As Boolean

\*\*\*\*\*

## BackColor

### Property BackColor As OLE\_COLOR

.....

.....

.....

## BorderStyle

### Property BorderStyle As BorderStyleConstants

.....

.....

.....

.....

**BulletIndent**

**Ture**

**Property BulletIndent As Single**

.....

.....

.....

**Container**

**Property Container As Object**

.....

.....

**DataBindings**

**Property DataBindings As  
DataBindings**

.....

.....

**DataChanged**

**Property DataChanged As Boolean**

.....  
.....  
**DataField**

.....  
**Property DataField As String**  
.....

.....  
**DisableNoScroll**  
**Property DisableNoScroll As Boolean**  
.....

.....  
**DragIcon**  
**Property DragIcon As IpictureDisp**  
.....

.....  
**DragMode**  
**Property DragMode As Integer**  
.....

.....  
.....  
**Enabled**  
**Property Enabled As Boolean**  
.....

.....  
.....  
**FileName**  
**Property FileName As String**  
.....

.....  
.....  
**Font**  
**Property Font As IfontDisp**  
.....

.....  
**Height**  
.....

**Property Height As Single**

---

---

**HelpContextID**

**Property HelpContextID As Long**

---

---

**HideSelection**

**Property HideSelection As Boolean**

---

---

**Hwnd**

**Property Hwnd As OLE\_HANDLE**

---

---

**Index**

---

.....  
**Property Index As Integer**  
.....

.....  
**Left**  
.....

.....  
**Property Left As Single**  
.....

.....  
**Locked**  
.....

.....  
**Property Locked As Boolean**  
.....

.....  
**MaxLength**  
.....

.....  
**Property MaxLength As Long**  
.....

.....  
**MouseIcon**  
.....

.....  
**Property MouseIcon As IpictureDisp**  
.....

**MousePointer**

.....  
**Property MousePointer As  
MousePointerConstants**  
.....

**MultiLine**

.....  
**Property MultiLine As Boolean**  
.....

**Name**

**Property Name As String**  
.....  
.....  
.....



# **Object**

## **Property Object As Object**

---

---

---

---

## **OLEDragMode**

### **Property OLEDragMode As OLEDragConstants**

---

---

## **OLEDropMode**

### **Property OLEDropMode As OLEDropConstants**

---

---

## **OLEObjects**

---

## **Property OLEObjects As OLEObjects**

---

---

---

## **Parent**

## **Property Parent As Form**

---

---

---

## **RightMargin**

## **Property RightMargin As Single**

---

---

---

## **ScrollBars**

## **Property ScrollBars As ScrollBarsConstants**

---

---

---

## **SelAlignment**

---

.....  
**Property SelAlignment As Variant**  
.....

**SelBold**

.....  
**Property SelBold As Variant**  
.....

**SelBullet**

.....  
**Property SelBullet As Variant**  
.....

**SelCharOffset**

.....  
**Property SelCharOffset As Variant**  
.....

**SelColor**

.....  
**Property SelColor As Variant**  
.....

.....  
.....  
**SetFontName**

.....  
**Property SetFontName As Variant**  
.....

**SetFontSize**

.....  
**Property SetFontSize As Variant**  
.....

**SetHangingIndent**

.....  
**Property SetHangingIndent As Variant**  
.....

**SetIndent**

.....  
**Property SetIndent As Variant**  
.....

**SelItalic**

.....  
**Property SelItalic As Variant**  
.....

**SelLength**

.....  
**Property SelLength As Long**  
.....

**SelProtected**

.....  
**Property SelProtected As Variant**  
.....

**SelRightIndent**

.....  
**Property SelRightIndent As Variant**  
.....  
.....

**SelRTF**

.....  
**Property SelRTF As String**  
.....

**SelStart**

.....  
**Property SelStart As Long**  
.....

**SelStrikeThru**

.....  
**Property SelStrikeThru As Variant**  
.....

**SelTabCount**

.....  
**Property SelTabCount As Variant**  
.....

**SelTabs**

**TAB**  
.....

**Property SelTabs(sElement As Integer)**

---

**SelText**

**Property SelText As String**

---

**SelUnderline**

**Property SelUnderline As Variant**

---

**TabIndex**

**Property TabIndex As Integer**

---

**TabStop**

**TAB**

**Property TabStop As Boolean**

.....  
.....  
**Tag**

.....  
**Property Tag As String**  
.....

.....  
**Text**

.....  
**Property Text As String**  
.....

.....  
**TextRTF**

.....  
**RTF**  
.....  
**Property TextRTF As String**  
.....

.....  
**ToolTipText**  
.....



.....  
**Property ToolTipText As String**  
.....

.....  
**Top**  
.....

.....  
**Property Top As Single**  
.....

.....  
**Visible**  
.....  
.....  
.....

.....  
**WhatsThisHelpID**  
.....  
.....  
.....

.....  
**Width**  
.....

.....  
**Property Width As Single**  
.....

**Change**

.....  
**()Event Change**  
.....

**Click**

**()Event Click**

\*\*\*\*\*

**DbClick**

.....  
**()Event DbClick**  
.....

## **DragDrop**

.....  
**Event DragDrop(source As Control, x  
(As Single, y As Single**  
.....

## **DragOver**

.....  
**Event DragOver(source As Control, x  
(As Single, y As Single, State As Integer**  
.....

## **GotFocus**

.....  
**()Event GotFocus**  
.....

## **KeyDown**

.....

.....  
**Event KeyDown(KeyCode As Integer,  
(Shift As Integer**  
.....

.....  
**KeyPress**  
.....

.....  
**(Event KeyPress(KeyAscii As Integer**  
.....

.....  
**KeyUp**  
.....

.....  
**Event KeyUp(KeyCode As Integer,  
(Shift As Integer**  
.....

.....  
**LostFocus**  
.....

## **()Event LostFocus**

---

---

## **MouseDown**

---

**Event MouseDown(Button As Integer,  
Shift As Integer, x As Single, y As  
Single**

---

---

## **MouseMove**

---

**Event MouseMove(Button As Integer,  
Shift As Integer, x As Single, y As  
Single**

---

---

## **MouseUp**

---

**Event MouseUp(Button As Integer,  
Shift As Integer, x As Single, y As  
(Single**

---

---

**OLECompleteDrag**

**Event OLECompleteDrag(Effect As  
(Long**

---

---

**OLEDragDrop**

**Event OLEDragDrop(Data As  
DataObject, Effect As Long, Button As  
Integer, Shift As Integer, x As Single, y  
(As Single**

---

---

**OLEDragOver**

---

**Event OLEDragOver(Data As  
DataObject, Effect As Long, Button As  
Integer, Shift As Integer, x As Single, y  
(As Single, State As Integer**

---

---

**OLEGiveFeedback**

**Event OLEGiveFeedback(Effect As  
(Long, DefaultCursors As Boolean**

---

---

**OLESetData**

**Event OLESetData(Data As  
(DataObject, DataFormat As Integer**

---

---

**OLEStartDrag**

---

**Event OLEStartDrag(Data As  
(DataObject, AllowedEffects As Long**

---

---

**SelChange**

---

**()Event SelChange**

---

---

**Drag**

---

**([Sub Drag([Action**

---

---

**Find**

---

**Function Find(bstrString As String,  
[vStart], [vEnd], [vOptions]) As Long**

---

---



## **GetLineFromChar**

.....  
**Function GetLineFromChar(lChar As  
Long) As Long**  
.....

.....  
**LoadFile**  
.....

**rtf TXT**

.....  
**Sub LoadFile(bstrFilename As String,  
([[vFileType**  
.....

.....  
**Move**  
.....

.....  
**Sub Move(Left As Single, [Top],  
([[Width], [Height**  
.....

.....  
**OLEDrag**  
.....

.....  
**()Sub OLEDrag**  
.....

.....  
.....  
**Refresh**

.....  
.....  
**()Sub Refresh**

.....  
**SaveFile**

**RTF**

.....  
**TXT**

.....  
**SelPrint**

.....  
**(Sub SelPrint(lHDC As Long**

.....  
**SetFocus**

.....  
**ShowWhatsThis**

**()Sub ShowWhatsThis**

**UpTo**

**Sub UpTo(bstrCharacterSet As String,  
([vForward], [vNegate**

**Zorder**

**([Sub ZOrder([Position**

**1**

**() Private Sub Form\_Load  
Timer1.Interval = 500  
End Sub**

**() Private Sub Timer1\_Timer**

```
(BackColor = QBColor(Rnd * 15  
(ForeColor = QBColor(Rnd * 10  
Picture1.BackColor = QBColor(Rnd  
    (* 15  
Picture1.ForeColor = QBColor(Rnd  
    (* 10
```

```
.....  
End Sub  
.....
```

```
.....  
2
```

```
()Private Sub Form_Click
```

```
Static intX As Integer
```

```
Select Case intX
```

```
Case 0
```

```
Set Command1.Container =
```

```
Picture1
```

```
Command1.Top= 0
```

```
Command1.Left= 0
```

```
Case 1
```

```
Set Command1.Container =
```

```
Frame1
```

```
Command1.Top= 0
```

```
Command1.Left= 0
```

```
Case 2  
.....
```

```
Set Command1.Container =  
    Form1  
    Command1.Top= 0  
    Command1.Left= 0  
    End Select  
    intX = intX + 1
```

```
.....  
End Sub  
.....
```

```
.....  
3
```

```
Private Sub Form_DragDrop (Source  
(As Control, X As Single, Y As Single  
    .Declare variable ' Dim Pic  
    .control Source.Move X, Y  
    Pic =  
' "ICONS\OFFICE\CRDFLE01.ICO"  
' If Source.DragIcon = False Then  
    Source.DragIcon =  
    Else  
    Source.DragIcon =  
    End If
```

```
.....  
End Sub  
.....
```

```
.....  
4  
.....
```

```
() Private Sub Form_Click  
If Command1.DragMode =  
vbManual Then  
Command1.DragMode =  
vbAutomatic  
Else  
Command1.DragMode = vbManual  
End If  
.....  
End Sub  
.....
```

**5**

```
Const winToolbox = 21001  
Const winCodeWindow = 21005  
() Private Sub Form_Load  
"App.HelpFile = "VB.HLP  
Frame1.HelpContextID =  
winColorPalette  
Text1.HelpContextID = winToolbox  
Form1.HelpContextID =  
winCodeWindow  
.....  
End Sub  
.....
```

**6**

```
    () Private Sub Form_Load  
Open "README.TXT" For Input  
    (Text1.Text = Input$(LOF(1), 1  
        Close 1  
        Form2.Visible = True  
  
Form1.Move 0, 1050, Screen.Width /  
    2, Screen.Height  
Form2.Move Screen.Width / 2, 1050,  
    Screen.Width / 2, Screen.Height  
Text1.Move 0, 0, ScaleWidth,  
    ScaleHeight  
.....  
End Sub  
.....
```

**7**

**.Declaration of a Windows routine '**  
**This statement should be placed in the '**  
**.module**

```
Declare Function SetWindowPos Lib  
"user32" Alias_ "SetWindowPos"  
(ByVal hwnd As Long, ByVal_  
hWndInsertAfter As Long, ByVal x As  
Long, ByVal y As_ Long, ByVal cx As
```





```
SetWindowPos hWnd,  
conHwndNoTopmost, 0, 0, 0, 0,  
conSwpNoActivate Or  
conSwpShowWindow  
End If  
End Sub
```

**This example automatically drops down the list portion of a ComboBox control whenever the ComboBox receives the focus. To try this example, create a new form containing a ComboBox control and an OptionButton control (used only to receive the focus). Create a new module using the Add Module command on the Project menu. Paste the Declare statement into the Declarations section of the new module, being sure that the statement is on one line with no break or wordwrap. Then paste the Sub procedure into the Declarations section of the form, and press F5. Use the TAB key to move the .focus to and from the ComboBox**

.....

**Declare Function SendMessage Lib  
"user32" Alias "SendMessageA"  
(ByVal hwnd As Long, ByVal wParam As Long, ByVal lParam As Long, ByVal wMsg As Long) As Long**

**() Private Sub Combo1\_GotFocus  
Const CB\_SHOWDROPDOWN =  
    &H14F  
    Dim Tmp  
    Tmp =  
        SendMessage(Combo1.hWnd,  
(&CB\_SHOWDROPDOWN, 1, ByVal 0  
    End Sub  
8**

**Select Case Button.Key  
    .Case Is = "open" ' Open file  
    Add code to Open a file here '  
    .Case Is = "save" ' Save file  
    Add code to Save a file here '**

**Case Else  
    If any other button is pressed '**

**.....  
    End Select  
.....  
.....**

**9**

```
() Private Sub Form_Click  
Set ' Width = Screen.Width * .75  
    .width of form  
Set ' Height = Screen.Height * .75  
    .height of form  
' Left = (Screen.Width - Width) / 2  
    .Center form horizontally  
' Top = (Screen.Height - Height) / 2  
    .Center form vertically  
.....  
End Sub  
.....
```

**10**

```
() Private Sub Form_Load  
.Put some items in the ListBox '  
"List1.AddItem "Selection 1  
"List1.AddItem "Selection 2  
"List1.AddItem "Selection 3  
"List1.AddItem "Selection 4  
"List1.AddItem "Selection 5  
End Sub  
.....
```



```
.Display all forms ' Form2.Show  
Form3.Show  
Form2.AutoRedraw = True  
Form3.AutoRedraw = True  
End Sub
```

```
() Private Sub Command1_Click  
' ReadCheckBox Form2.Check1  
Call procedure in other module  
and ' ReadCheckBox Form3.Check1  
.send control as argument  
End Sub
```

```
.Enter this code into Module1 '  
Sub ReadCheckBox (Source As  
(Control  
If Source.Value Then  
Clear ' Source.Parent.Cls  
.parent form  
Source.Parent.Print "CheckBox is ON."  
' Display on parent form  
Else
```

.....

```
Clear ' Source.Parent.Cls
        .parent form
Source.Parent.Print "CheckBox
.is OFF." ' Display on parent form
End If
.....
End Sub
.....
```

12

```
Private Sub Option1_Click(Index As
(Integer
If RichTextBox1.SelLength > 0 Then
RichTextBox1.SelAlignment =
Index
End If
.....
End Sub
.....
```

13

```
()Private Sub RichTextBox1_SelChange
If IsNull(RichTextBox1.SelBullet) =
True Then
Check1.Value = vbGrayed
ElseIf RichTextBox1.SelBullet =
True Then
.....
```

```
Check1.Value = vbChecked
ElseIf RichTextBox1.SelBullet =
    False Then
Check1.Value = vbUnchecked
End If
.....
End Sub
.....
```

14

```
() Private Sub VScroll1_Change
    VScroll1.Max =
    RichTextBox1.SelFontSize
(VScroll1.Min = -(VScroll1.Max
RichTextBox1.SelCharOffset =
    VScroll1.Value
.....
End Sub
.....
```

15

```
()Private Sub Command1_Click
CommonDialog1.ShowColor
    RichTextBox1.SelColor =
    CommonDialog1.Color
.....
End Sub
.....
```

16

```
() Private Sub Command1_Click  
CommonDialog1.Flags = cdlCFBoth  
CommonDialog1.ShowFont  
With RichTextBox1  
SelFontName = .  
CommonDialog1.FontName  
SelFontSize = .  
CommonDialog1.FontSize  
SelBold = .  
CommonDialog1.FontBold  
SelItalic = .  
CommonDialog1.FontItalic  
SelStrikethru = .  
CommonDialog1.FontStrikethru  
SelUnderline = .  
CommonDialog1.FontUnderline  
End With  
.....  
End Sub  
.....
```

17

```
() Private Sub Command1_Click  
CommonDialog1.Flags = Both  
.....
```



**CommonDialog1.ShowFont  
With RichTextBox1**

**SelFontName = .**

**CommonDialog1.FontName**

**SelFontSize = .**

**CommonDialog1.FontSize**

**SelBold = .**

**CommonDialog1.FontBold**

**SelItalic = .**

**CommonDialog1.FontItalic**

**SelStrikethru = .**

**CommonDialog1.FontStrikethru**

**SelUnderline = .**

**CommonDialog1.FontUnderline**

**End With**

.....  
**End Sub**  
.....

.....  
**18**  
.....

**()Private Sub Command1\_Click**

**Dim Margins As Integer**

**(Margins = CInt(Text1.Text**

**With RichTextBox1**

**SelStart = 1.**  
.....

```
        SelLength = .  
        (Len(RichTextBox1.Text  
        SelIndent = Margins.  
        SelRightIndent = Margins.  
        End With  
.....  
        End Sub  
.....
```

19

```
    () Private Sub Form_Load  
        Text1.Text = "Two of the peak  
        "human experiences  
        Text1.Text = Text1.Text & " are  
        ".good food and classical music  
        End Sub  
    () Private Sub Form_Click  
        Declare ' Dim Search, Where  
        .variables  
        .Get search string from user '  
        Search = InputBox("Enter text to be  
        (":found  
        ' Where = InStr(Text1.Text, Search)  
        .Find string in text  
        ,If found ' If Where Then  
.....
```

```
set ' Text1.SelStart = Where - 1
      selection start and
' Text1.SelLength = Len(Search)
      .set selection length

      Else
' MsgBox "String not found."
      .Notify user
      End If
      End Sub
```

**This example shows how the Clipboard object is used in cut, copy, paste, and delete operations. To try this example, create a form with a TextBox control and use the Menu Editor to create an Edit menu (for each of the commands, set the Caption property = Cut, Copy, Paste, and Delete, respectively; set the Name property = EditCut, EditCopy, .(EditPaste, and EditDelete, respectively**

```
() Private Sub EditCut_Click
```

.....

```
Clear the contents of the '
    .Clipboard
    Clipboard.Clear
.Copy selected text to Clipboard '
    Clipboard.SetText
    Screen.ActiveControl.SetText
.Delete selected text '
''' = Screen.ActiveControl.SetText
    End Sub
```

```
() Private Sub EditCopy_Click
Clear the contents of the '
    .Clipboard
    Clipboard.Clear
.Copy selected text to Clipboard '
    Clipboard.SetText
    Screen.ActiveControl.SetText
    End Sub
```

```
() Private Sub EditPaste_Click
Place text from Clipboard into '
    .active control
```

.....

```
Screen.ActiveControlSelText =  
    () Clipboard.GetText  
    End Sub
```

```
() Private Sub EditDelete_Click  
    .Delete selected text '
```

```
"" = Screen.ActiveControlSelText
```

```
.....  
End Sub  
.....
```

```
.....  
20
```

```
() Private Sub Command1_Click  
Open "mytext.rtf" For Output As 1  
Print #1, RichTextBox1.SelRTF  
Close 1
```

```
.....  
End Sub  
.....
```

```
.....  
21
```

```
()Private Sub Command1_Click  
    With RichTextBox1  
        SelTabCount = 5.  
For X = 0 To .SelTabCount - 1  
    SelTabs(X) = 5 * X.  
.....
```

**Next X**

**End With**

.....  
**End Sub**  
.....

.....  
**22**  
.....

**() Private Sub Form\_Click**

**.Declare variables ' Dim I, X**

**Reverse tab order by setting start '**

**.value of X**

**If CommandX(0).TabIndex = 0**

**Then X = 4 Else X = 1**

**For I = 0 To 3**

**' CommandX(I).Caption = X**

**.Set caption**

**CommandX(I).TabIndex = X**

**.Set tab order ' - 1**

**If CommandX(0).TabIndex =**

**3 Then**

**.Decrement X ' X = X - 1**

**Else**

**.Increment X ' X = X + 1**

**End If**

**Next I**  
.....

**End Sub**

---

**23**

```
() Private Sub Form_Load  
    Picture1.Tag =  
""ICONS\ARROWS\POINT03.ICO  
    Picture2.Tag =  
""ICONS\ARROWS\POINT04.ICO  
End Sub
```

```
Private Sub Picture3_DragOver (Source  
As Control, X As Single, Y As Single,  
(State As Integer  
    If State = vbEnter Then  
        Select based on each '  
            .PictureBox's Name property  
        Select Case Source.Name  
            "Case "Picture1  
                .Load icon for Picture1 '  
                Source.DragIcon =  
Case            LoadPicture(Picture1.Tag)  
                ""Picture2  
                .Load icon for Picture2 '
```

---

```
Source.DragIcon =  
(LoadPicture(Picture2.Tag  
End Select  
ElseIf State = vbLeave Then  
When source isn't over '  
.Picture3, unload icon  
Source.DragIcon = LoadPicture  
()  
End If  
.....  
End Sub  
.....
```

24

```
()Private Sub Form_Load  
Dim nodX As Node ' Declare an  
.object variable for the Node  
Dim i As Integer ' Declare a  
.variable for use as a counter  
  
Add one Node to the TreeView '  
control, and call it the first node  
()Set nodX = TreeView1.Nodes.Add  
"nodX.Text = "First Node  
.....
```



## **Populate the ListBox'**

```
List1.AddItem "Node1" ' Add  
    .each item to list  
"List1.AddItem "Node2  
"List1.AddItem "Node3  
"List1.AddItem "Node4  
"List1.AddItem "Node5
```

```
"List1.AddItem "Node6  
"List1.AddItem "Node7
```

**Add child nodes to the first Node ' object. Use the**

**.ListBox to populate the control '**

```
For i = 0 To List1.ListCount - 1
```

```
Set nodX =
```

```
(TreeView1.Nodes.Add(1, tvwChild
```

```
(nodX.Text = List1.List(i
```

```
Next i
```

```
Treeview1.Nodes(1).Selected = True
```

**Make sure the ' nodX.EnsureVisible**

**.node is visible**

.....

**End Sub**

**Private Sub**

**TreeView1\_NodeClick(ByVal Node As  
Node**

**Display the clicked Node object's  
.Text property**

**Label1.Caption = Node.Text**

**End Sub**

**25**

**() Private Sub Form\_Load**

**.Declare variable ' Dim I**

**Set ' FileCab(0).BorderStyle = 0  
.BorderStyle**

**FileCab(1).BorderStyle = 0**

**.Load icons into picture boxes '**

**FileCab(1).Picture =**

**LoadPicture('ICONS\OFFICE\FILES0  
("3B.ICO**

```
FileCab(0).Picture =
LoadPicture("ICONS\OFFICE\FILES0
           ("3A.ICO
           For I = 0 To 1
' FileCab(I).Move 400, 400
           .Place graphics at same spot
           Next I
Set to ' FileCab(1).Visible = False
           .invisible
Set to ' FileCab(0).Visible = True
           .visible

           End Sub

Private Sub FileCab_Click (Index As
           (Integer
           .Declare variable ' Dim I
           For I = 0 To 1
Switch the visibility for both '
           .graphics
FileCab(I).Visible = Not
FileCab(I).Visible
           Next I
.....
           End Sub
```

.....  
.....  
**26**

**() Private Sub Form\_Load**

**Turn on ' Picture1.AutoSize = -1  
.AutoSize**

**Make the ' Label1.Visible = 0  
.label invisible**

**File1.Pattern = "\*.BMP; \*.ICO;  
' Set file patterns \*.WMF"**

**End Sub**

**' Any Private Sub Dir1\_Change ()  
change in Dir1**

**is reflected ' File1.Path = Dir1.Path  
.in File1**

**End Sub**

**' Any Private Sub Drive1\_Change ()  
change in Drive1**

**is ' Dir1.Path = Drive1.Drive  
.reflected in Dir1**

**End Sub**  
.....

**Private Sub File1\_MouseDown (Button  
As Integer, Shift As Integer, X As  
Single, Y As Single**

**.Declare variable ' Dim DY  
' Get height DY = TextHeight("A")  
.of one line  
Label1.Move File1.Left, File1.Top +  
Y - DY /2, File1.Width, DY  
.Drag label outline ' Label1.Drag  
End Sub**

**Private Sub Dir1\_DragOver (Source As  
Control, X As Single, Y As Single, State  
(As Integer**

**.Change pointer to no drop '  
If State = 0 Then  
Source.MousePointer = 12  
.Use default mouse pointer '  
If State = 1 Then  
Source.MousePointer = 0  
End Sub**

.....

**Private Sub Drive1\_DragOver (Source  
As Control, X As Single, Y As Single,  
(State As Integer**

**.Change pointer to no drop '  
If State = 0 Then  
Source.MousePointer = 12  
.Use default mouse pointer '  
If State = 1 Then  
Source.MousePointer = 0  
End Sub**

**Private Sub Form\_DragOver (Source  
As Control, X As Single, Y As Single,  
(State As Integer**

**.Change pointer to no drop '  
If State = 0 Then  
Source.MousePointer = 12  
.Use default mouse pointer '  
If State = 1 Then  
Source.MousePointer = 0  
End Sub**

.....

**Private Sub File1\_DragOver (Source As Control, X As Single, Y As Single, State (As Integer**

**On Error Resume Next**

**If State = 0 And**

**Right\$(File1.FileName,4) = ".ICO"**

**Then**

**Label1.DragIcon =**

**LoadPicture(File1.Path + "\" +  
(File1.FileName**

**If Err Then MsgBox "The icon file  
".can't be loaded**

**ElseIf State = 1 Then**

**Label1.DragIcon = LoadPicture**

**.' Use no drag icon ()**

**End If**

**End Sub**

**Private Sub Picture1\_DragDrop (Source  
(As Control, X As Single, Y As Single**

**On Error Resume Next**

.....

```
Picture1.Picture =  
LoadPicture(File1.Path + "\" +  
File1.FileName
```

```
If Err Then MsgBox "The picture file  
".can't be loaded
```

```
.....  
End Sub  
.....
```

```
.....  
27
```

```
()Private Sub Command1_Click  
Dim FoundPos As Integer  
Dim FoundLine As Integer  
Find the text specified in the '  
.TextBox control  
FoundPos =  
RichTextBox1.Find(Text1.Text, , ,  
(rtfWholeWord
```

```
Show message based on whether '  
.the text was found or not
```

```
If FoundPos <> -1 Then  
.....
```



```
    Returns number of line '
        .containing found text
        FoundLine =
RichTextBox1.GetLineFromChar(Foun
        (dPos
MsgBox "Word found on line "
        (& CStr(FoundLine
        Else
        ".MsgBox "Word not found

        End If
```

```
.....
        End Sub
.....
```

```
.....
28
```

```
()Private Sub Command1_Click
    Dim FoundPos As Integer
    Dim FoundLine As Integer
    Find the text specified in the '
        .TextBox control
        FoundPos =
RichTextBox1.Find(Text1.Text, , ,
        (rtfWholeWord
```

```
.....
```

**Show message based on whether ' ' .the text was found or not**

```
    If FoundPos <> -1 Then  
        Returns number of line ' '  
            .containing found text  
            FoundLine =  
RichTextBox1.GetLineFromChar(Foun  
            (dPos  
MsgBox "Word found on line "  
            (& CStr(FoundLine  
            Else  
".MsgBox "Word not found
```

**End If**

**End Sub**

**29**

```
    ()Private Sub Command1_Click  
CommonDialog1.Filter = "Rich Text  
    "Format files|*.rtf  
CommonDialog1.ShowOpen
```

**RichTextBox1.LoadFile  
CommonDialog1.FileName, rtfRTF**

**End Sub**

**30**

**() Private Sub Form\_Click**

**.Declare variables ' Dim Inch, Msg**

**Msg = "Choose OK to resize and  
" move this form by**

**Msg = Msg & "changing the value of  
".properties**

**.Display message ' MsgBox Msg**

**.Set inch in twips ' Inch = 1440**

**.Set width ' Width = 4 \* Inch**

**.Set height ' Height = 2 \* Inch**

**.Set left to origin ' Left = 0**

**.Set top to origin ' Top = 0**

**Msg = "Now choose OK to resize  
" and move this form**

**Msg = Msg & "using the Move  
".method**

**.Display message ' MsgBox Msg**

**Move Screen.Width - 2 \* Inch,  
Screen.Height - Inch, 2 \* Inch, Inch**

**End Sub**

**31**

**()Private Sub Command1\_Click  
CommonDialog1.ShowSave  
RichTextBox1.SaveFile**

**CommonDialog1.FileName, rtfRTF**

**End Sub**

**32**

**()Private Sub Command1\_Click**

**CommonDialog1.Flags =**

**cdlPDReturnDC + cdlPDNoPageNums**

**If RichTextBox1.SelLength = 0 Then**

**CommonDialog1.Flags =**

**CommonDialog1.Flags + cdlPDAllPages**

**Else**

**CommonDialog1.Flags =**

**CommonDialog1.Flags + cdlPDSelection**

**End If**

**CommonDialog1.ShowPrinter**

```
''' Printer.Print
RichTextBox1.SelPrint
CommonDialog1.hDC
```

```
.....
End Sub
.....
```

```
.....
33
```

```
Private ThisControl As Control
```

```
Private Sub
```

```
Command1_MouseUp(Button As
Integer, Shift As Integer, X As Single, Y
(As Single
```

```
If Button = vbRightButton Then
```

```
Set ThisControl =
```

```
Command1
```

```
PopupMenu
```

```
mnuBtnContextMenu
```

```
End If
```

```
Set ThisControl = Nothing
```

```
End Sub
```

```
()Private Sub mnuBtnWhatsThis_Click
ThisControl.ShowWhatsThis
```

```
.....
End Sub
```



**End Select**

**End If**

.....  
**End Sub**  
.....

.....  
**35**  
.....

**Private Sub RichTextBox1\_KeyUp**  
**((KeyCode As Integer, Shift As Integer**

**If Shift = vbAltMask Then**

**Select Case KeyCode**

**:If Alt+S '**

**Case vbKeyS**

**Move insertion point to '**

**.the end of the sentence**

**RichTextBox1.Upto**

**".?!:", True, False**

**:If Alt+W '**

**Case vbkeyW**

**Move insertion point to '**

**.the end of the word**

**RichTextBox1.Upto "**

**.?!:", True, False**

**End Select**

**End If**  
.....

**End Sub**

---

**36**

```
() Private Sub Form_Load  
' Add list List1.AddItem "John"  
    .box entries  
    "List1.AddItem "Paul  
    "List1.AddItem "George  
    "List1.AddItem "Ringo  
End Sub
```

```
() Private Sub List1_DblClick  
' Command1.Value = True  
    .Trigger Click event  
End Sub
```

```
() Private Sub Command1_Click  
Display ' Text1.Text = List1.Text  
    .selection
```

---

**End Sub**

---

**37**

---



```
() Private Sub Form_Load  
' Add list List1.AddItem "John"  
    .box entries  
    "List1.AddItem "Paul  
"List1.AddItem "George  
"List1.AddItem "Ringo  
End Sub
```

```
() Private Sub List1_DblClick  
' Command1.Value = True  
    .Trigger Click event  
End Sub
```

```
() Private Sub Command1_Click  
Display ' Text1.Text = List1.Text  
    .selection  
.....  
End Sub  
.....
```

**38**

```
Private Sub Picture1_DragOver (Source  
As Control, X As Single, Y As Single,  
(State As Integer  
    Select Case State  
.....
```

```
Case vbEnter
    .Load icon '
Source.DragIcon =
LoadPicture('ICONS\ARROWS\POIN
("T03.ICO
Case vbLeave
Source.DragIcon =
.' Unload iconLoadPicture()
End Select
End Sub
```

```
Private Sub Picture1_DragDrop (Source
    (As Control, X As Single, Y As Single
' Source.DragIcon = LoadPicture()
.Unload icon
.....
End Sub
.....
```

39

```
() Private Sub Form_Load
Label1.AutoSize = True
End Sub
.....
```

```
Private Sub OptionGroup_GotFocus  
    ((Index As Integer  
        Select Case Index  
            Case 0  
Label1.Caption = "Option 1  
            ".has the focus  
            Case 1  
Label1.Caption = "Option 2  
            ".has the focus  
        End Select  
    End Sub
```

```
Private Sub OptionGroup_LostFocus  
    ((Index As Integer  
        "" = Label1.Caption  
.....  
        End Sub  
.....
```

```
.....  
    40  
Private Sub Text1_KeyDown (KeyCode  
    (As Integer, Shift As Integer  
    Dim ShiftDown, AltDown,  
        CtrlDown, Txt  
.....
```

**ShiftDown = (Shift And  
vbShiftMask) > 0**  
**AltDown = (Shift And vbAltMask) >  
0**  
**CtrlDown = (Shift And vbCtrlMask)  
> 0**

**' If KeyCode = vbKeyF2 Then  
.Display key combinations**

**If ShiftDown And CtrlDown And  
AltDown Then  
Txt =  
"."SHIFT+CTRL+ALT+F2**  
**ElseIf ShiftDown And AltDown  
Then  
".Txt = "SHIFT+ALT+F2**  
**ElseIf ShiftDown And CtrlDown  
Then  
".Txt = "SHIFT+CTRL+F2**  
**ElseIf CtrlDown And AltDown Then  
".Txt = "CTRL+ALT+F2**  
**ElseIf ShiftDown Then  
".Txt = "SHIFT+F2**  
**ElseIf CtrlDown Then**

.....

```
    ".Txt = "CTRL+F2
ElseIf AltDown Then
    ".Txt = "ALT+F2
ElseIf SHIFT = 0 Then
    ".Txt = "F2
    End If
```

```
Text1.Text = "You pressed " & Txt
    End If
```

```
.....
    End Sub
.....
```

```
.....
41
```

```
Private Sub Text1_KeyPress (KeyAscii
    (As Integer
    (Char = Chr(KeyAscii
    ((KeyAscii = Asc(UCase(Char
.....
    End Sub
.....
```

```
.....
42
```

```
    Private Sub Text1_GotFocus
        .Show focus with red '
        (Text1.BackColor = RGB(255, 0, 0
.....
```

```
Label1.Caption = "Text1 has the  
".focus  
End Sub
```

```
() Private Sub Text1_LostFocus  
.Show loss of focus with blue '  
(Text1.BackColor = RGB(0, 0, 255  
Label1.Caption = "Text1 doesn't  
".have the focus
```

```
.....  
End Sub  
.....
```

```
.....  
43
```

```
Dim PaintNow As Boolean  
Private Sub Form_MouseDown (Button  
As Integer, Shift As Integer, X As  
(Single, Y As Single  
.Enable painting 'PaintNow = True  
End Sub
```

```
Private Sub Form_MouseUp (Button As  
Integer, Shift As Integer, X As Single, Y  
(As Single  
.....
```

```
Disable ' PaintNow = False  
.painting  
End Sub
```

```
Private Sub Form_MouseMove (Button  
As Integer, Shift As Integer, X As  
(Single, Y As Single  
If PaintNow Then  
' Draw a point PSet (X, Y)  
End If  
  
End Sub
```

```
() Private Sub Form_Load  
Use wider ' DrawWidth = 10  
.brush  
' Set ForeColor = RGB(0, 0, 255)  
.drawing color
```

```
.....  
End Sub  
.....
```

```
.....  
44  
.....
```

```
Declare ' Dim PaintNow As Boolean  
.variable  
.....
```

```
Private Sub Form_MouseDown (Button  
As Integer, Shift As Integer, X As  
Single, Y As Single  
.Brush on 'PaintNow = True  
End Sub
```

```
Private Sub Form_MouseUp (Button As  
Integer, X As Single, Y As Single  
Turn off ' PaintNow = False  
.painting  
End Sub
```

```
Private Sub Form_MouseMove (Button  
As Integer, Shift As Integer, X As  
Single, Y As Single  
If PaintNow Then  
' Draw a point PSet (X, Y)  
End If  
End Sub
```

```
() Private Sub Form_Load  
Use wider ' DrawWidth = 10  
.brush
```

---



**' Set ForeColor = RGB(0, 0, 255)  
.drawing color**

.....  
**End Sub**  
.....

.....  
**45**  
.....

**Declare ' Dim PaintNow As Boolean  
.variable**

**Private Sub Form\_MouseDown (Button  
As Integer, Shift As Integer, X As  
(Single, Y As Single  
.Brush on 'PaintNow = True  
End Sub**

**Private Sub Form\_MouseUp (Button As  
(Integer, X As Single, Y As Single  
Turn off ' PaintNow = False  
.painting  
End Sub**

**Private Sub Form\_MouseMove (Button  
As Integer, Shift As Integer, X As  
(Single, Y As Single  
If PaintNow Then**  
.....

```
. ' Draw a point      PSet (X, Y)  
      End If  
      End Sub
```

```
    () Private Sub Form_Load  
Use wider '      DrawWidth = 10  
      .brush  
' Set ForeColor = RGB(0, 0, 255)  
      .drawing color
```

```
.....  
      End Sub  
.....
```

```
.....  
46  
.....
```

```
()Private Sub RichTextBox1_SelChange  
If RichTextBox1.SelLength > 0 Then  
  EditCutMenu.Enabled = True  
  EditCopyMenu.Enabled = True  
  Else  
    EditCutMenu.Enabled = False  
    EditCopyMenu.Enabled = False  
  End If
```

```
.....  
      End Sub  
.....
```

```
.....  
47  
.....
```

```
()Private Sub Command1_Click  
    CommonDialog1.Flags =  
    cdlPDReturnDC + cdlPDNoPageNums  
If RichTextBox1.SelLength = 0 Then  
    CommonDialog1.Flags =  
CommonDialog1.Flags + cdlPDAllPages  
    Else  
    CommonDialog1.Flags =  
CommonDialog1.Flags + cdlPDSelection  
    End If  
    CommonDialog1.ShowPrinter  
    "" Printer.Print  
    RichTextBox1.SelPrint  
    CommonDialog1.hDC  
.....  
    End Sub  
.....
```

**48**

```
() Private Sub Command1_Click  
Open "mytext.rtf" For Output As 1  
Print #1, RichTextBox1.TextRTF  
Close 1
```

```
.....  
End Sub
```

.....  
.....  
**49**  
.....

```
()Private Sub Form_Load  
Declare object ' Dim pnlX As Panel  
.variable for Panel objects  
Dim I As Integer
```

```
Add 6 Panel objects to the single '   
.default Panel object  
.making 7 Panel objects '   
For I = 1 to 6  
Set pnlX =  
StatusBar1.Panels.Add  
Next I  
End Sub
```

```
()Private Sub Form_Click  
If the Count of the collection is 7, '   
.then clear the collection  
Otherwise, add one Panel and use '   
the collection's Count property  
.to set its Style '   
If StatusBar1.Panels.Count = 7 Then  
.....
```

.....

```
StatusBar1.Panels.Clear
Else
Dim pnlX As Panel
Set pnlX =
(StatusBar1.Panels.Add( , , "simple", 0
The Style property is '
enumerated from 0 to 6. Use the Panels
Count property -1 to set the '
.Style property for the new Panel
Display all panels regardless of '
.form width
pnlX.minwidth =
("TextWidth("simple
pnlX.AutoSize = sbrSpring
pnlX.Style =
StatusBar1.Panels.Count - 1
End If
End Sub
```

.....

50

```
Dim pnlX As Panel ' Declare object
variable for Panel objects.
Dim i As Integer
```

.....

**' Add 6 Panel objects to the single  
default Panel object,  
' making 7 Panel objects.**

**For i = 1 To 6  
Set pnlX =  
StatusBar1.Panels.Add(, , , i)  
pnlX.AutoSize = sbrSpring  
Next i  
End Sub**

**Private Sub Form\_Click()**

**' If the Count of the collection is 1,  
add 6 Panel objects.  
' Otherwise, remove the first panel  
from the collection.**

**If StatusBar1.Panels.Count = 1 Then**

**Dim sbrX As Panel**

**Dim i As Integer**

**For i = 1 To 6 ' Each panel has its  
style set by i.**

**Set sbrX =**

**StatusBar1.Panels.Add(, , , i)**

**sbrX.AutoSize = sbrSpring**

**Next i**

.....

**Else ' Remove the first panel.  
StatusBar1.Panels.Remove 1  
End If**

.....  
**End Sub**  
.....

.....  
**RichTextBox**  
.....

**Loai.rtf**

.....  
**VERSION 5.00**

**Object = "{F9043C88-F6F2-101A-  
A3C9-08002B2F49FB}#1.2#0";  
"Comdlg32.ocx"**

**Object = "{3B7C8863-D78F-101B-  
B9B5-04021C009402}#1.1#0";  
"RICHTEX32.OCX"**

**Begin VB.Form Form1**

**Caption = " "**

**ClientHeight = 9210**

**ClientLeft = 60**  
.....

```
ClientTop      = 345
ClientWidth    = 9645
LinkTopic      = "Form1"
RightToLeft    = -1 'True
ScaleHeight    = 9210
ScaleWidth     = 9645
StartPosition  = 3 'Windows
Default
Begin RichTextLib.RichTextBox
  RLoai
    Height      = 6000
    Left        = 90
    TabIndex    = 12
    Top         = 45
    Width       = 9510
    _ExtentX    = 16775
    _ExtentY    = 10583
    _Version    = 327681
    Enabled     = -1 'True
    TextRTF     =
    $"Form1.frx":0000
  End
Begin VB.CommandButton
  Command10
    Caption     = "  "
```

.....



```
        Height      = 600
        Left        = 180
    RightToLeft    = -1 'True
        TabIndex    = 11
        Top         = 7245
        Width       = 1185
            End
Begin VB.CommandButton
    Command9
        Caption     = "      "
        Height      = 690
        Left        = 135
    RightToLeft    = -1 'True
        TabIndex    = 10
        Top         = 8100
        Width       = 1185
            End
Begin VB.CommandButton
    Command8
        Caption     = "      "
        Height      = 555
        Left        = 1485
    RightToLeft    = -1 'True
        TabIndex    = 9
```

---

**Top = 7245**

**Width = 1680**

**End**

**Begin VB.CommandButton  
Command7**

**Caption = " "**

**Height = 555**

**Left = 3240**

**RightToLeft = -1 'True**

**TabIndex = 8**

**Top = 7245**

**Width = 1770**

**End**

**Begin VB.CommandButton  
Command6**

**Caption = " "**

**Height = 555**

**Left = 5220**

**RightToLeft = -1 'True**

**TabIndex = 6**

**Top = 7245**

**Width = 1455**

**End**

**Begin VB.VScrollBar VScroll1**

.....

**Height = 5010**  
**Left = 225**  
**RightToLeft = -1 'True**  
**TabIndex = 5**  
**Top = 990**  
**Width = 420**  
**End**

**Begin VB.CommandButton**  
**Command5**

**Caption = " "**  
**Height = 555**  
**Left = 6975**  
**RightToLeft = -1 'True**  
**TabIndex = 4**  
**Top = 7200**  
**Width = 2445**  
**End**

**Begin MSComDlg.CommonDialog**  
**CLoai**

**Left = 0**  
**Top = 6120**  
**\_ExtentX = 847**  
**\_ExtentY = 847**  
**\_Version = 393216**  
**End**

.....

**Begin VB.CommandButton  
Command4**

**Caption = " "**  
**Height = 690**  
**Left = 450**  
**TabIndex = 3**  
**Top = 6300**  
**Width = 2040**  
**End**

**Begin VB.CommandButton  
Command3**

**Caption = " "**  
**Height = 600**  
**Left = 2655**  
**TabIndex = 2**  
**Top = 6345**  
**Width = 1860**  
**End**

**Begin VB.CommandButton  
Command2**

**BackColor = &H000000FF&**  
**Caption = " "**  
**Height = 600**



**Left** = 4590  
**TabIndex** = 1  
**Top** = 6345  
**Width** = 1950  
**End**

**Begin VB.CommandButton  
Command1**

**Caption** = ""  
**Height** = 645  
**Left** = 6705  
**TabIndex** = 0  
**Top** = 6345  
**Width** = 1680  
**End**

**Begin VB.Label Label1**

**Alignment** = 1 'Right Justify  
**AutoSize** = -1 'True  
**BackStyle** = 0 'Transparent  
**Caption** = ""  
"

**BeginProperty Font**

**Name** = "Arabic  
Transparent"  
**Size** = 15.75

.....

**Charset = 178**  
**Weight = 700**  
**Underline = 0 'False**  
**Italic = 0 'False**  
**Strikethrough = 0 'False**  
**EndProperty**  
**Height = 375**  
**Left = 2205**  
**RightToLeft = -1 'True**  
**TabIndex = 7**  
**Top = 8370**  
**Width = 5490**

**End**

**End**

**Attribute VB\_Name = "Form1"**

**Attribute VB\_GlobalNameSpace =**  
**False**

**Attribute VB\_Creatable = False**

**Attribute VB\_PredeclaredId = True**

**Attribute VB\_Exposed = False**

**Private Sub Command1\_Click()**

**RLoai.SelText = 0**

.....

**End Sub**

**Private Sub Command10\_Click()  
    RLoai.SelText =  
    Clipboard.GetText(vbCFRTF)**

**End Sub**

**Private Sub Command2\_Click()  
    CLoai.ShowColor  
    RLoai.SelColor = CLoai.Color  
    End Sub**

**Private Sub Command3\_Click()**  
.....

```
CLoai.Flags = cdlCFBoth  
CLoai.ShowFont  
With RLoai  
.SelFontName = CLoai.FontName  
.SelFontSize = CLoai.FontSize  
.SelBold = CLoai.FontBold  
.SelItalic = CLoai.FontItalic  
.SelStrikeThru =  
CLoai.FontStrikethru  
.SelUnderline =  
CLoai.FontUnderline  
End With
```

```
End Sub
```

```
Private Sub Command4_Click()  
RLoai.SelText = " "
```

```
End Sub
```

```
Private Sub Command5_Click()  
Dim lo  
lo = App.Path  
Dim lm As String
```

.....



```
lm = lo & "/" & "Loai.rtf"  
RLoai.SaveFile lm, 0
```

**End Sub**

```
Private Sub Command7_Click()  
Clipboard.SetText RLoaiSelText
```

**End Sub**

```
Private Sub Command8_Click()  
Clipboard.GetText (vbCFRTF)
```

**End Sub**

```
Private Sub Command9_Click()  
Screen.ActiveControl.RLoai  
SendKeys "{^} A", True
```

**End Sub**

---

```
Private Sub Form_Load()  
    Dim lo  
    lo = App.Path  
    Dim lm As String  
    lm = lo & "/" & "Loai.rtf"  
    RLoai.filename = lm
```

```
End Sub
```

```
Private Sub VScroll1_Change()  
VScroll1.Max = RLoai.SelFontSize  
VScroll1.Min = -(VScroll1.Max)  
RLoai.SelCharOffset =  
VScroll1.Value  
End Sub
```

-:

33-:

-:

-:

-

-

-

-

32

-5

.....

**6318676-011-:**

**-:**

**Loai12345678@maktoob.com**

