

(Forms)

omaras34@Hotmail.com:

	(Esc)	(Form)	-1
.(True) (KeyPreview)		(Properties)	.1
	-:	(KeyDown)	.2
<pre>Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer) If KeyCode = vbKeyEscape Then Unload Me End If End Sub</pre>			
		(Form)	-2
	-:	(Load)	.1
<pre>Private Sub Form_Load() Me.Move (Screen.Width - Me.Width) / 2, (Screen.Height - Me.Height) End Sub</pre>			
	(X)	(Form)	-3
	-:	(Unload)	.1
<pre>Private Sub Form_Unload(Cancel As Integer) Cancel = True End Sub</pre>			
		(Form)	-4
.(True) (KeyPreview)		(Properties)	.1
	-:	(KeyDown)	.2
<pre>Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer) If KeyCode = vbKeyF1 Then MsgBox "Hello,World!!", vbInformation, "Hello" End If End Sub</pre>			

(Form) -5

(Module) .1

```
Public Sub Color_Border(F As Form, C As ColorConstants)
    F.Cls
    Me.Line (Me.ScaleLeft, Me.ScaleTop)-_
    (Me.ScaleWidth - 10, Me.ScaleHeight - 10), C, B
End Sub
```

-. (Resize) (Load) .2

```
Private Sub Form_Resize()
    Color_Border Me, vbGreen
End Sub
```

```
Private Sub Form_Load()
    Color_Border Me, vbGreen
End Sub
```

.(True) (KeyPreview) (Properties) .3

(Forms) -6

(CommandButton) .1

-. (Click) .2

```
Private Sub Command1_Click()
    Dim F As Form
    For Each F In Forms
        Unload F
        Set F = Nothing
    Next F
End Sub
```

(Forms) -7

-.:

```
Private Sub Command1_Click()
    Form2.Show vbModal
End Sub
```

(Form)

-8

"C:\Program Files\Microsoft Visual Studio\VB98\Template\Forms"

.(Add Form)

(Form)

-9

-.:

.(Module)

.1

```
Public Function FormLoadedByName(S As String) As Boolean
    Dim i As Integer, FName As String
    FName = LCase$(S)
    FormLoadedByName = False
    For i = 0 To Forms.Count - 1
        If LCase$(Forms(i).Name) = FName Then
            FormLoadedByName = True
            Exit Function
        End If
    Next
End Function
```

-.:

.2

```
Private Sub Command1_Click()
    If FormLoadedByName("Form2") = True Then
        MsgBox "The Form is loaded"
    Else
        MsgBox "The Form is not loaded"
    End If
End Sub
```

-(Forms)

(Variables)

-10

(Public)

(General)

.1

-.:



-.:

.2

```
Private Sub Form_Load()
    Form1.MyVar = "My Name"
End Sub
```

(Module)

(Form) -11

(Title Bar)

:-

.(API)

(FlashWindow())

(Module)

.1

```
Declare Function FlashWindow Lib "user32" _
    (ByVal hwnd As Long, ByVal bInvert As Long) As Long
```

:-

(Timer)

.2

Interval = 100 ■

Enabled = False ■

:-

(Timer)

.3

```
Private Sub Timer1_Timer()
    FlashWindow Me.hwnd, True
End Sub
```

:-

(Load)

.4

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

(Form) -12

:-

(Unload)

```
Private Sub Form_Unload(Cancel As Integer)
    Me.WindowState = vbMinimized
End Sub
```

-(Title Bar)

(Form)

-13

:-

(Properties)

```

Caption = "" ■
ControlBox = False ■
MinButton = False ■
MaxButton = False ■

```

-(Title Bar) (Form) -14

```

(ReleaseCapture()) (SendMessage()) (Module) .1
-: (API)

```

```

Declare Function SendMessage Lib "User32" Alias "SendMessageA" _
(ByVal hwnd As Long, ByVal wParam As Long, ByVal lParam _
As Long, lParam As Any) As Long

```

```

Declare Sub ReleaseCapture Lib "User32" ()

```

```

Public Const WM_NCLBUTTONDOWN = &HA1
Public Const HTCAPTION = 2

```

-(MouseDown) .2

```

Private Sub Form_MouseDown(Button As Integer, _
Shift As Integer, x As Single, Y As Single)
Dim ReturnValue As Long
If Button = 1 Then
Call ReleaseCapture
ReturnValue = SendMessage(Me.hwnd, WM_NCLBUTTONDOWN, HTCAPTION, 0&)
End If
End Sub

```

-(API) (Form) -15

```

-: (Module) .1

```

```

Sub PaintForm(FormName As Form, Orientation%, RStart%, _
GStart%, BStart%, RInc%, GInc%, BInc%)
    On Error Resume Next
    Dim x As Integer, y As Integer, z As Integer, Cycles As Integer
    Dim R%, G%, B%
    R% = RStart%: G% = GStart%: B% = BStart%
    If Orientation% = 0 Then
        Cycles = FormName.ScaleHeight \ 100
    Else
        Cycles = FormName.ScaleWidth \ 100
    End If
    For z = 1 To 100
        x = x + 1
        Select Case Orientation
            Case 0: 'Top to Bottom
                If x > FormName.ScaleHeight Then Exit For
                FormName.Line (0, x)-(FormName.Width, x + Cycles - 1), _
                RGB(R%, G%, B%), BF
            Case 1: 'Left to Right
                If x > FormName.ScaleWidth Then Exit For
                FormName.Line (x, 0)-(x + Cycles - 1, FormName.Height), _
                RGB(R%, G%, B%), BF
        End Select
        x = x + Cycles
        R% = R% + RInc%: G% = G% + GInc%: B% = B% + BInc%
        If R% > 255 Then R% = 255
        If R% < 0 Then R% = 0
        If G% > 255 Then G% = 255
        If G% < 0 Then G% = 0
        If B% > 255 Then B% = 255
        If B% < 0 Then B% = 0
    Next z
End Sub

```

-: (Load) .2

```

Private Sub Form_Load()
    PaintForm Me, 1, 100, 0, 255, 1, 0, -1
End Sub

```

-: (Form) -16

(General) (API) (GetSystemMetrics()) .1
-: (Const)

```
Private Declare Function GetSystemMetrics Lib "user32" _
    (ByVal nIndex As Long) As Long
```

```
Private Const SM_CXSCREEN = 0
Private Const SM_CYSCREEN = 1
```

-: (Load) .2

```
Private Sub Form_Load()
    Dim wid As Long
    Dim hgt As Long

    wid = GetSystemMetrics(SM_CXSCREEN)
    hgt = GetSystemMetrics(SM_CYSCREEN)

    Move 0, 0, _
        ScaleX(wid, vbPixels, vbTwips), _
        ScaleY(hgt, vbPixels, vbTwips)
End Sub
```

(Form) -17

-:

-: (Module) .1

```
Public Sub FormPosition_Get(F As Form)
    Dim buf As String
    Dim l As Integer, t As Integer
    Dim h As Integer, w As Integer
    Dim pos As Integer
    buf = GetSetting(App.EXENAME, "FormPosition", F.Tag, "")
    If buf = "" Then
        F.Move (Screen.Width - F.Width) \ 2, (Screen.Height - F.Height) \ 2
    Else
        pos = InStr(buf, ",")
        l = CInt(Left(buf, pos - 1))
        buf = Mid(buf, pos + 1)
        pos = InStr(buf, ",")
        t = CInt(Left(buf, pos - 1))
        buf = Mid(buf, pos + 1)
        pos = InStr(buf, ",")
        w = CInt(Left(buf, pos - 1))
        h = CInt(Mid(buf, pos + 1))
        F.Move l, t, w, h
    End If
End Sub
```

```
Public Sub FormPosition_Put(F As Form)
    Dim buf As String
    buf = F.Left & "," & F.Top & "," & F.Width & "," & F.Height
    SaveSetting App.EXENAME, "FormPosition", F.Tag, buf
End Sub
```

-: (Load) .2

```
Sub Form_Load()
    FormPosition_Get Me
End Sub
```

-: (Unload) .3

```
Sub Form_Unload()
    FormPosition_Put Me
End Sub
```

(Form) -18

-: (Deactivate)

```
Private Sub Form_Deactivate()
    Me.SetFocus
End Sub
```

(Form) -19

(General) (API) (PaintDesktop()) .1

-:

```
Private Declare Function PaintDesktop Lib "user32" _
    (ByVal hdc As Long) As Long
```

-: (Paint) .2

```
Private Sub Form_Paint()
    PaintDesktop Me.hdc
End Sub
```

(Form) "?" -20

-:

BorderStyle = 2-Sizable ■

```
MaxButton = False ■  
MinButton = False ■  
WhatsThisButton = True ■  
WhatsThisHelp = True ■
```

(Form) -21

```
-: (Load)
```

```
Private Sub Form_Load()  
    Form2.Show , Me  
End Sub
```

(Form) -22

```
-: (Query_Unload)
```

```
Private Sub Form_QueryUnload(Cancel As Integer, _  
    UnloadMode As Integer)  
    Cancel = (MsgBox("Do you want to quit?", _  
        vbQuestion + vbYesNo, "Exit") = vbNo)  
End Sub
```

(Form) -23

```
-: (Paint)
```

```
Private Sub Form_Paint()  
    Dim pic As StdPicture  
    Dim X As Long  
    Dim Y As Long  
    Set pic = LoadPicture("C:\icon_warning.gif")  
  
    For Y = 0 To ScaleHeight Step ScaleY(pic.Height)  
        For X = 0 To ScaleWidth Step ScaleX(pic.Width)  
            PaintPicture pic, X, Y  
        Next  
    Next  
End Sub
```

(Form) -24

<pre> -: (QueryUnload) Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer) Me.Picture = LoadPicture("") DoEvents Me.BackColor = vbBlack On Error Resume Next Dim i As Long For i = 1 To Me.Height Me.Top = Me.Top + 1 Me.Height = Me.Height - 2 Me.Left = Me.Left + 1 Me.Width = Me.Width - 2 Next DoEvents End Sub </pre>	<p>(Form) -25</p>
<pre> -: -: (General) </pre>	<p>.1</p>
<pre> Public captions As String Public counter As Integer -: (Load) </pre>	<p>.2</p>
<pre> Private Sub Form_Load() captions = "Hello From Visual Basic" End Sub 200 (Interval) (Timer) -: (Timer) </pre>	<p>.3 .4</p>
<pre> Private Sub Timer1_Timer() counter = counter + 1 If counter = Len(captions) Then counter = 0 End If Me.Caption = Right(captions, counter) End Sub </pre>	<p>(Form) -26</p>
<pre> (API) (GetSysColor()) (Module) -: (Const) </pre>	<p>.1</p>

```
Public Declare Function GetSysColor Lib "user32" _
  (ByVal nIndex As Long) As Long
```

```
Global Const COLOR_BACKGROUND = 1
```

-(Module) .2

```
Sub ColorFormBackground(frmForm As Form)
  Const COLOR_BACKGROUND = 1 'Desktop
  frmForm.BackColor = GetSysColor(COLOR_BACKGROUND)
End Sub
```

-(Load) .3

```
Private Sub Form_Load()
  ColorFormBackground Me
End Sub
```

(Form) -27

-(VScrollBar)

-(Load) (General) (Integer) .1

```
Dim oldPos As Integer
```

```
Private Sub Form_Load()
  Dim iFullFormHeight As Integer
  Dim iDisplayHeight As Integer

  iFullFormHeight = 3765
  iDisplayHeight = 1800

  Me.Height = iDisplayHeight

  With VScroll1
    .Height = Me.ScaleHeight
    .Min = 0
    .Max = iFullFormHeight - iDisplayHeight
    .SmallChange = Screen.TwipsPerPixelY * 10
    .LargeChange = .SmallChange
  End With
End Sub
```

-(Scroll) (Change) .2

```

Private Sub pScrollForm()
    Dim ctl As Control

    For Each ctl In Me.Controls
        If Not (TypeOf ctl Is VScrollBar) Then
            ctl.Top = ctl.Top + oldPos - VScroll11.Value
        End If
    Next

    oldPos = VScroll11.Value
End Sub

```

```

Private Sub VScroll11_Scroll()
    Call pScrollForm
End Sub

```

```

Private Sub VScroll11_Change()
    Call pScrollForm
End Sub

```

(MDIForm)

-28

-:

-:

.1

```

Private Sub mnuWindowArrangeIcons_Click()
    Me.Arrange vbArrangeIcons
End Sub

```

-:

.2

```

Private Sub mnuWindowCascade_Click()
    Me.Arrange vbCascade
End Sub

```

-:

.3

```

Private Sub mnuWindowTileHorizontal_Click()
    Me.Arrange vbTileHorizontal
End Sub

```

-:

.4

```

Private Sub mnuWindowTileVertical_Click()
    Me.Arrange vbTileVertical
End Sub

```

.5

```
Private Sub mnuWindowCloseAll_Click()  
Dim frm As Form  
  
    For Each frm In Forms  
        If Not (frm Is Me) Then  
            Unload frm  
        End If  
    Next frm  
End Sub
```

.6

```
Private Sub mnuWindowMinimizeAll_Click()  
Dim frm As Form  
  
    For Each frm In Forms  
        If Not (frm Is Me) Then  
            frm.WindowState = vbMinimized  
        End If  
    Next frm  
End Sub
```

.7

```
Private Sub mnuWindowRestoreAll_Click()  
Dim frm As Form  
  
    For Each frm In Forms  
        If Not (frm Is Me) Then  
            frm.WindowState = vbNormal  
        End If  
    Next frm  
End Sub
```

.8

```
Private Sub mnuWindowNewWindow_Click()  
Dim frm As New Form1  
  
    frm.Show  
End Sub
```

(MDIForm)

-29

(Resize)

(Properties)

```
Private Sub Form_Resize()
    Me.WindowState = vbMaximized
End Sub
```

(Form) -30

```
(SetWindowRgn()) (CreateRoundRectRgn()) .1
-: (General)
```

```
Private s_X As Long
Private s_Y As Long
```

```
Private Declare Function CreateRoundRectRgn Lib "gdi32" _
    (ByVal X1 As Long, ByVal Y1 As Long, _
    ByVal X2 As Long, ByVal Y2 As Long, _
    ByVal X3 As Long, ByVal Y3 As Long) As Long
```

```
Private Declare Function SetWindowRgn Lib "User32" _
    (ByVal hwnd As Long, ByVal hRgn As Long, _
    ByVal bRedraw As Boolean) As Long
```

-: (Load) .2

```
Private Sub Form_Load()
    Dim Y As Long
    Dim n As Long
    X = Me.Width / Screen.TwipsPerPixelX
    Y = Me.Height / Screen.TwipsPerPixelY
    n = 120
    SetWindowRgn Me.hwnd, _
    CreateRoundRectRgn(0, 0, X, Y, n, n), True
End Sub
```

omaras34@Hotmail.com:

www.alafari.com