

```
'Calendar - Calendar demo program
'Copyright (c) 1997 SoftCircuits Programming (R)
'Redistributed by Permission.
,
'This example program demonstrates how to create a mini calendar in
'Visual Basic 5.0. It takes advantage of the changes made to VB in
'version 4 that allow forms to have public methods and properties.
'Although the Calendar form contains a fair amount of code, you can
'take advantage of all of its features by calling the single method,
'GetDate().
,
'This program may be distributed on the condition that it is
'distributed in full and unchanged, and that no fee is charged for
'such distribution with the exception of reasonable shipping and media
'charged. In addition, the code in this program may be incorporated
'into your own programs and the resulting programs may be distributed
'without payment of royalties.
,
'This example program was provided by:
' SoftCircuits Programming
' http://www.softcircuits.com
' P.O. Box 16262
' Irvine, CA 92623
Option Explicit

'Grid dimensions for days
Private Const GRID_ROWS = 6
Private Const GRID_COLS = 7

'Private variables
Private m_CurrDate As Date, m_bAcceptChange As Boolean
Private m_nGridWidth As Integer, m_nGridHeight As Integer

'Public function: If user selects date, sets UserDate to selected
'date and returns True. Otherwise, returns False.
Public Function GetDate(UserDate As Date, Optional Title) As Boolean
    'Store user-specified date
    m_CurrDate = UserDate
    'Use caller-specified caption if any
    If Not IsMissing(Title) Then
        Caption = Title
    End If
    'Display this form
    Me.Show vbModal
    'Return selected date
    If m_bAcceptChange Then
        UserDate = m_CurrDate
    End If
    'Return value indicates if date was selected
    GetDate = m_bAcceptChange
End Function

'Form initialization
Private Sub Form_Load()
    'Calculate calendar grid measurements
    m_nGridWidth = ((picMonth.ScaleWidth - Screen.TwipsPerPixelX) \ GRID_COLS)
    m_nGridHeight = ((picMonth.ScaleHeight - Screen.TwipsPerPixelY) \ GRID_ROWS)
    m_bAcceptChange = False
End Sub

'Process user keystrokes
Private Sub picMonth_KeyDown(KeyCode As Integer, Shift As Integer)
    Dim NewDate As Date

    Select Case KeyCode
        Case vbKeyRight
            NewDate = DateAdd("d", 1, m_CurrDate)
        Case vbKeyLeft
            NewDate = DateAdd("d", -1, m_CurrDate)
        Case vbKeyDown
            NewDate = DateAdd("ww", 1, m_CurrDate)
```

```

Case vbKeyUp
    NewDate = DateAdd("ww", -1, m_CurrDate)
Case vbKeyPageDown
    NewDate = DateAdd("m", 1, m_CurrDate)
Case vbKeyPageUp
    NewDate = DateAdd("m", -1, m_CurrDate)
Case vbKeyReturn
    m_bAcceptChange = True
    Unload Me
    Exit Sub
Case vbKeyEscape
    Unload Me
    Exit Sub
Case Else
    Exit Sub
End Select
SetNewDate NewDate
KeyCode = 0
End Sub

'Double-click accepts current date
Private Sub picMonth_DblClick()
    m_bAcceptChange = True
    Unload Me
End Sub

' Select the date by mouse
Private Sub picMonth_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
    Dim i As Integer, MaxDay As Integer

    'Determine which date is being clicked
    i = Weekday(DateSerial(Year(m_CurrDate), Month(m_CurrDate), 1)) - 1
    i = (((x \ m_nGridWidth) + 1) + ((y \ m_nGridHeight) * GRID_COLS)) - i
    'Get last day of current month
    MaxDay = Day(DateAdd("d", -1, DateSerial(Year(m_CurrDate), Month(m_CurrDate) + 1, 1)))
    If i >= 1 And i <= MaxDay Then
        SetNewDate DateSerial(Year(m_CurrDate), Month(m_CurrDate), i)
    End If
End Sub

'Click on ">>" goes to next month
Private Sub lblNext_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
    If Button And vbLeftButton Then
        SetNewDate DateAdd("m", 1, m_CurrDate)
    End If
End Sub

'Double-click has same effect
Private Sub lblNext_DblClick()
    SetNewDate DateAdd("m", 1, m_CurrDate)
End Sub

'Click on "<<" goes to previous month
Private Sub lblPrev_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
    If Button And vbLeftButton Then
        SetNewDate DateAdd("m", -1, m_CurrDate)
    End If
End Sub

'Double-click has same effect
Private Sub lblPrev_DblClick()
    SetNewDate DateAdd("m", -1, m_CurrDate)
End Sub

'Changes the selected date
Private Sub SetNewDate(NewDate As Date)
    If Month(m_CurrDate) = Month(NewDate) And Year(m_CurrDate) = Year(NewDate) Then
        DrawSelectionBox False
        m_CurrDate = NewDate
        DrawSelectionBox True
    Else

```

```

        m_CurrDate = NewDate
        picMonth_Paint
    End If
End Sub

'Here's the calendar paint handler; displays the calendar days
Private Sub picMonth_Paint()
    Dim i As Integer, j As Integer, x As Integer, y As Integer
    Dim NumDays As Integer, CurrPos As Integer, bCurrMonth As Boolean
    Dim MonthStart As Date, buffer As String

    'Determine if this month is today's month
    If Month(m_CurrDate) = Month(Date) And Year(m_CurrDate) = Year(Date) Then
        bCurrMonth = True
    End If
    'Get first date in the month
    MonthStart = DateSerial(Year(m_CurrDate), Month(m_CurrDate), 1)
    'Number of days in the month
    NumDays = DateDiff("d", MonthStart, DateAdd("m", 1, MonthStart))
    'Get first weekday in the month (0 - based)
    j = Weekday(MonthStart) - 1
    'Tweak for 1-based For/Next index
    j = j - 1
    'Show current month/year
    lblMonth = Format$(m_CurrDate, "mmmm yyyy")
    'Clear existing data
    picMonth.Cls
    'Display dates for current month
    For i = 1 To NumDays
        CurrPos = i + j
        x = (CurrPos Mod GRID_COLS) * m_nGridWidth
        y = (CurrPos \ GRID_COLS) * m_nGridHeight
        'Show date as bold if today's date
        If bCurrMonth And i = Day(Date) Then
            picMonth.Font.Bold = True
        Else
            picMonth.Font.Bold = False
        End If
        'Center date within "date cell"
        buffer = CStr(i)
        picMonth.CurrentX = x + ((m_nGridWidth - picMonth.TextWidth(buffer)) / 2)
        picMonth.CurrentY = y + ((m_nGridHeight - picMonth.TextHeight(buffer)) / 2)
        'Print date
        picMonth.Print buffer;
    Next i
    'Indicate selected date
    DrawSelectionBox True
End Sub

'Draw or clears the selection box around the current date
Private Sub DrawSelectionBox(bSelected As Boolean)
    Dim clrTopLeft As Long, clrBottomRight As Long
    Dim i As Integer, x As Integer, y As Integer

    'Set highlight and shadow colors
    If bSelected Then
        clrTopLeft = vbButtonShadow
        clrBottomRight = vb3DHighlight
    Else
        clrTopLeft = vbButtonFace
        clrBottomRight = vbButtonFace
    End If
    'Compute location for current date
    i = Weekday(DateSerial(Year(m_CurrDate), Month(m_CurrDate), 1)) - 1
    i = i + (Day(m_CurrDate) - 1)
    x = (i Mod GRID_COLS) * m_nGridWidth
    y = (i \ GRID_COLS) * m_nGridHeight
    'Draw box around date
    picMonth.Line(x, y + m_nGridHeight)-Step(0, -m_nGridHeight), clrTopLeft
    picMonth.Line-Step(m_nGridWidth, 0), clrTopLeft
    picMonth.Line-Step(0, m_nGridHeight), clrBottomRight

```

```
    picMonth.Line -Step(-m_nGridWidth, 0), clrBottomRight  
End Sub
```



```
'Calendar - Calendar demo program
'Copyright (c) 1997 SoftCircuits Programming (R)
'Redistributed by Permission.
,
'This example program demonstrates how to create a mini calendar in
'Visual Basic 5.0. It takes advantage of the changes made to VB in
'version 4 that allow forms to have public methods and properties.
'Although the Calendar form contains a fair amount of code, you can
'take advantage of all of its features by calling the single method,
'GetDate().
,
'This program may be distributed on the condition that it is
'distributed in full and unchanged, and that no fee is charged for
'such distribution with the exception of reasonable shipping and media
'charged. In addition, the code in this program may be incorporated
'into your own programs and the resulting programs may be distributed
'without payment of royalties.
,
'This example program was provided by:
' SoftCircuits Programming
' http://www.softcircuits.com
' P.O. Box 16262
' Irvine, CA 92623
Option Explicit

'Let user select date from frmCalendar
Private Sub cmdCalendar_Click()
    Dim UserDate As Date

    UserDate = CVDate(txtDate)
    If frmCalendar.GetDate(UserDate) Then
        txtDate = UserDate
    End If
End Sub

'Exit program
Private Sub cmdExit_Click()
    Unload Me
End Sub

'Initialize date on load
Private Sub Form_Load()
    'Default to today's date
    txtDate = Date
End Sub
```

Date:

Calendar...

Exit