

# Contents

## Overview

[General Information](#)  
[Properties and Events](#)  
[Installation](#)  
[Distribution](#)  
[Support](#)  
[Registering](#)

## Properties

[BackColor](#)  
[DataField](#)  
[DateAtPoint](#)  
[DateAtPointJul](#)  
[DateDispStyle](#)  
[DayAlignment](#)  
[DOWAlign](#)  
[DOWBackColor](#)  
[DOWBorder](#)  
[DOWDispStyle](#)  
[DOWFontBold](#)  
[DOWFontItalic](#)  
[DOWFontName](#)  
[DOWFontSize](#)  
[DOWFontStrikeThru](#)  
[DOWFontUnderline](#)  
[DOWForeColor](#)  
[EnableKeys](#)  
[FirstDOW](#)  
[FixedDayHeight](#)  
[ForeColor](#)  
[Julian](#)  
[LineColor](#)  
[MonAlign](#)  
[MonBackColor](#)  
[MonDispStyle](#)  
[MonFontBold](#)  
[MonFontItalic](#)  
[MonFontName](#)  
[MonFontSize](#)  
[MonFontStrikeThru](#)  
[MonFontUnderline](#)  
[MonForeColor](#)  
[OtherMonBackColor](#)  
[OtherMonForeColor](#)  
[PointX, PointY](#)  
[PrintAction](#)  
[PrintHeight,PrintWidth,PrintX,PrintY](#)  
[PrintHDC](#)  
[SelDayBackColor](#)  
[SelDayForeColor](#)  
[ShowAllDays](#)

ShowArrows  
ShowLines  
ShowSelection  
Text

## Events

ClickDay  
DblClickDay  
DrawDay  
DrawOnDay  
QueryChangeDay

## General Information

Kalendar is custom control for displaying a calendar. The primary purpose of writing Kalendar was to provide a level of flexibility to VB programmers that was not available in most other calendar custom controls.

With Kalendar, you can:

- Affect the way each day box is drawn.
- Draw anything within each day box.
- Print the calendar (with any custom drawing) either full page, or anywhere and any size on the page.

Check out the DrawDay and DrawOnDay events, and the PrintAction property to get a quick jump on these features.

Kalendar is also a bound control, supports drag and drop, and most of the standard events and properties.

# Properties and Events

The following tables list all of the properties and events for Kalendar. The properties and events that are specific to Kalendar are highlighted.

## Properties

(About)	<u>BackColor</u>	BorderStyle
<u>DataField</u>	DataSource	<u>DateAtPoint</u>
<u>DateAtPointJul</u>	<u>DayAlignment</u>	<u>DOWAlign</u>
<u>DOWBackColor</u>	<u>DOWBorder</u>	<u>DOWDispStyle</u>
<u>DOWFontBold</u>	<u>DOWFontItalic</u>	<u>DOWFontName</u>
<u>DOWFontSize</u>	<u>DOWFontStrikeThru</u>	<u>DOWFontUnderline</u>
<u>DOWForeColor</u>	DragIcon	DragMode
Enabled	<u>EnableKeys</u>	<u>FirstDOW</u>
<u>FixedDayHeight</u>	FontBold	FontItalic
FontName	FontSize	FontStrikeThru
FontUnderline	<u>ForeColor</u>	Height
HelpContextID	Index	<u>Julian</u>
Left	<u>LineColor</u>	<u>MonAlign</u>
<u>MonBackColor</u>	<u>MonDispStyle</u>	<u>MonFontBold</u>
<u>MonFontItalic</u>	<u>MonFontName</u>	<u>MonFontSize</u>
<u>MonFontStrikeThru</u>	<u>MonFontUnderline</u>	<u>MonForeColor</u>
MousePointer	Name	<u>OtherMonBackColor</u>
<u>OtherMonForeColor</u>	<u>PointX</u>	<u>PointY</u>
<u>PrintAction</u>	<u>PrintHDC</u>	<u>PrintHeight</u>
<u>PrintWidth</u>	<u>PrintX</u>	<u>PrintY</u>
<u>SelDayBackColor</u>	<u>SelDayForeColor</u>	<u>ShowAllDays</u>
<u>ShowArrows</u>	<u>ShowLines</u>	<u>ShowSelection</u>
TabIndex	TabStop	Tag
<u>Text</u>	Top	Visible
Width		

## Events

<u>ClickDay</u>	<u>DbClickDay</u>	DragDrop
DragOver	<u>DrawDay</u>	<u>DrawOnDay</u>
GotFocus	KeyDown	KeyPress
KeyUp	LostFocus	MouseDown
MouseMove	MouseMove	<u>QueryChangeDay</u>

## Installation

To install Kalendar, copy the following files to your \WINDOWS\SYSTEM directory.

- KALENDAR.VBX

- KALENDAR.HLP
- KALENDAR.LIC *For registered users only.*

## **Distribution**

Kalendar.VBX is royalty free after registering it. You can distribute it with any application that you develop with it. However, you MAY NOT distribute the Kalendar.LIC file.

If you do not register the control, you will not be able to create .EXEs that use the Kalendar.VBX.

One license is required for each developer using Kalendar.VBX. Call for quantity discounts.

# Support

Kalendar support is provided through the following channels:

**Compuserve:** CIS 73003,131

**Mail:** Parachute Software  
324 Debbie Drive  
Waukesha, WI 53186

**Phone:** (414) 524-1992

Priority is given to registered users of the product, but we will try to answer any questions you have during the evaluation of the product.

# Registering

When you register Kalendar, you will receive the distributable version of the custom control and free telephone support (toll-charges apply). Each copy of Kalendar.VBX is \$25.00. To order Kalendar, send check or money order to:

**Parachute Software**  
324 Debbie Drive  
Waukesha, WI 53186

Please specify diskette preference. Thank you.



# BackColor

## Usage

*Kalendar*.BackColor [= *colorval*]

## Description

Sets the background color of the day box.

## Data Type

Color

## Notes

This property can be set in the DrawDay event.

# DataField

## Usage

[form.] control.**DataField** [ = *fieldname* ]

## Description

Used to specify/retrieve the data field that the control is bound to. Kalendar can only be bound to Date/Time fields. Binding to any other field type will cause an error to be generated.

# DateAtPoint

Example

## Usage

*textvar* = *Kalendar*.DateAtPoint

Read only.

## Description

For some of the standard functions (MouseMove, MouseDown, DragOver, DragDrop), the coordinates of the mouse are returned. By using the **PointX** and **PointY** properties, you can determine what date (if any), the mouse is currently over. If the coordinates specified in **PointX** and **PointY** are not over a date, this property will return an empty string ("").

## Data Type

String

# DateAtPointJul

Example

## Usage

*longvar* = *Kalendar*.DateAtPointJul

Read only.

## Description

For some of the standard functions (MouseMove, MouseDown, DragOver, DragDrop), the coordinates of the mouse are returned. By using the **PointX** and **PointY** properties, you can determine what date (if any), the mouse is currently over. If the coordinates specified in **PointX** and **PointY** are not over a date, this property will return -1. This is a read-only property.

## Data Type

Long

## Notes

Use the DateAtPoint property to determine the validity of the date at a given point.

# DateDisplayStyle

## Usage

*Kalendar.DateDisplayStyle* [ = *interval* ]

## Description

Sets the manner in which the boxes around each day are displayed.

## Valid Values

Value	Description
0	Normal - The days are drawn with the currently specified colors and fonts.
1	3-D - The days are drawn as gray 3-D buttons. There is no other control over the appearance of the button when this is the display style.
2	User-Defined - For each day to be drawn, the DrawDay event is fired. See the DrawDay event for an example of this display style

## Data Type

Integer

# DayAlignment

## Usage

*Kalendar.DayAlignment* [ = *interval* ]

## Description

Sets the manner in which the day numbers are displayed within each day box.

## Data Type

Integer

## Valid Values

Value	Description
0	Upper-left
1	Upper-center
2	Upper-right
3	Mid-left
4	Mid-center
5	Mid-right
6	Lower-left
7	Lower-center
8	Lower-right

## Notes

This property can be set in the DrawDay event.

# DOWAlign

## Usage

*Kalendar.DOWAlign* [ = *integerval* ]

## Description

Sets the alignment of the text in the day of the week title.

## Data Type

Integer

## Valid Values

Value	Description
0	Align left
1	Align right
2	Align center

# DOWBackColor

## Usage

*Kalendar.DOWBackColor* [ = *colorval* ]

## Description

Sets the background color of the day of the week title.

## Data Type

Color

# DOWBorder

## Usage

*Kalendar.DOWBorder* [ = *interval* ]

## Description

Sets whether or not a border is drawn around the text in the day of the week title.

## Data Type

Integer (True | False)



# DOWDispStyle

## Usage

*Kalendar.DOWDispStyle* [ = *interval* ]

## Description

Sets the manner in which the day of the week title is shown.

## Valid Values

Value	Description
0	None - No day of week is shown.
1	Long - The day of the week is completely spelled out.
2	Medium - The day of the week is displayed as the first three characters.
3	Short - The day of the week is displayed as a single letter.

## Data Type

Integer

## Notes

The day of week descriptions are stored in a string table in the VBX file. If you need to change the text, use a resource editor (such as Microsoft's AppStudio or Borland's Resource Workshop) to modify the descriptions as required.

# DOWFontBold

## Usage

*Kalendar.DOWFontBold* [ = *interval* ]

## Description

This property is for setting the font in which to display the text in the day of the week title on the Kalendar.

# DOWFontItalic

## Usage

*Kalendar.DOWFontItalic* [ = *interval* ]

## Description

This property is for setting the font in which to display the text in the day of the week title on the Kalendar.

# DOWFontName

## Usage

*Kalendar.DOWFontName* [ = *string* ]

## Description

This property is for setting the font in which to display the text in the day of the week title on the Kalendar.

# DOWFontSize

## Usage

*Kalendar.DOWFontSize* [ = *numericval* ]

## Description

This property is for setting the font in which to display the text in the day of the week title on the Kalendar.

# DOWFontStrikeThru

## Usage

*Kalendar.DOWFontStrikeThru* [ = *integerval* ]

## Description

This property is for setting the font in which to display the text in the day of the week title on the Kalendar.

# DOWFontUnderline

## Usage

*Kalendar.DOWFontUnderline* = [*integer*]

## Description

This property is for setting the font in which to display the text in the day of the week title on the Kalendar.

# DOWForeColor

## Usage

*Kalendar.DOWForeColor* [ = *colorval* ]

## Description

Sets the foreground color of the day of the week title.

## Data Type

Color



# EnableKeys

## Usage

*Kalendar.EnableKeys* [ = *interval* ]

## Description

Enables/disables the use of keys to navigate Kalendar. When this value is true, the left and right arrow keys move the date selection back and forth one day. The up and down arrows move the date selection back and forth one week.

## Data Type

Integer (True | False)

# FirstDOW

## Usage

*Kalendar.FirstDOW* [ = *interval* ]

## Description

Sets the day of the week which should be shown as the first day on the Kalendar. Valid values are 1 = Sunday, to 7 = Saturday.

## Data Type

Integer

# FixedDayHeight

## Usage

*Kalendar.FixedDayHeight* [ = *interval* ]

## Description

This affects how the number of weeks for a particular month are drawn. Since all months will fit into 6 weeks, calendars are typically drawn with enough space reserved for those 6 weeks. For instance, June 1994 requires a total of five 7 day weeks to be displayed, whereas July 1994 needs six weeks.

Setting this property to True will cause Kalendar to dynamically size the height of each week, depending on how the month is laid out.

## Data Type

Integer (True | False)

# ForeColor

## Usage

*Kalendar.ForeColor* [= *colorval*]

## Description

Sets the color of the text being drawn in the day box.

## Data Type

Color

## Notes

This property can be set in the DrawDay event.

# Julian

## Usage

*Kalendar*.**Julian** = [*longval*]

## Description

Allows you to access the current Kalendar date as a Julian value. See the [Text](#) property to get the date as a text string.

## Data Type

Long

# LineColor

## Usage

*Kalendar.LineColor* = [colorval]

## Description

Sets the color of the line that is drawn around each day box.

## Data Type

Color

## Notes

This value can be changed in the DrawDay event.

# MonAlign

## Usage

*Kalendar.MonAlign* [ = *interval* ]

## Description

Sets the manner in which the month title is aligned.

## Valid Values

Value	Description
0	Align left.
1	Align right.
2	Align center.

## Data Type

Integer

# MonBackColor

## Usage

*Kalendar.MonBackColor*[ = *colorval*]

## Description

Sets the background color of the month title and any space not covered by a day box.

## Data Type

Color



# MonDispStyle

## Usage

*Kalendar.MonDisplayStyle* [ = *interval* ]

## Description

Sets the manner in which the month title shown.

## Valid Values

Value	Description
0	None - No title is shown.
1	Month - Just the month name is shown.
2	Month/Year - The month and the year are shown.

## Data Type

Integer

## Notes

The month descriptions are stored in a string table in the VBX file. If you need to change the display, use a resource editor (such as Microsoft's AppStudio or Borland's Resource Workshop) to modify the descriptions as required.

# MonFontBold

## Usage

Use in the same manner as the standard fonts.

## Description

This property is for setting the font in which to display the month title on the Kalendar.

# MonFontItalic

## Usage

*Kalendar.MonFontItalic* = [ *interval* ]

## Description

This property is for setting the font in which to display the month title on the Kalendar.

# MonFontName

## Usage

*Kalendar.MonFontName* = [ *stringval* ]

## Description

This property is for setting the font in which to display the month title on the Kalendar.

# MonFontSize

## Usage

*Kalendar.MonFontSize* = [ *numericval* ]

## Description

This property is for setting the font in which to display the month title on the Kalendar.

# MonFontStrikeThru

## Usage

*Kalendar.MonFontStrikeThru* = [ *interval* ]

## Description

This property is for setting the font in which to display the month title on the Kalendar.

# MonFontUnderline

## Usage

*Kalendar.MonFontUnderline* = [ *interval* ]

## Description

This property is for setting the font in which to display the month title on the Kalendar.

# MonForeColor

## Usage

*Kalendar.MonForeColor*[ = *colorval*]

## Description

Sets the foreground color of the month title.

## Data Type

Color



# OtherMonBackColor

## Usage

*Kalendar*.OtherMonBackColor[ = *colorval*]

## Description

Sets the background color of a day box for a day that is not in the current month. This property only takes affect when the ShowAllDays property is True.

## Data Type

Color

## Notes

This property can be set in the DrawDay event.

# OtherMonForeColor

## Usage

*Kalendar*.OtherMonForeColor[ = *colorval*]

## Description

Sets the foreground color of a day box that is not in the current month. This property only takes affect when the ShowAllDays property is True.

## Data Type

Color

## Notes

This property can be set in the DrawDay event.

# PointX, PointY

## Usage

*Kalendar*.PointX [= twips]

*Kalendar*.PointY [= twips]

## Description

For some of the standard functions (MouseMove, MouseDown, DragOver, DragDrop), the coordinates of the mouse are returned. By using the **PointX** and **PointY** properties, you can determine what date (if any), the mouse is currently over. See the **DateAtPoint** property for an example of using these properties.

## Data Type

Integer

# PrintAction

## Usage

*Kalendar.PrintAction* [ = *interval* ]

Write only.

## Description

This property allows you to print a Kalendar on the currently selected printer. There are three ways you can print a Kalendar, Portrait, Landscape and User-Defined. The first two actions print the specified Kalendar full page, using the specified orientation. Using the user defined action allows you to print the Kalendar anywhere and any size on your page.

## Data Type

Integer

## Valid Values

Value	Description
1	Print full page, portrait orientation
2	Print full page, landscape orientation
3	Print the Kalendar using the PrintHeight, PrintWidth, PrintX and PrintY properties.

# PrintHeight,PrintWidth,PrintX,PrintY

Example

## Usage

*Kalendar.PrintHeight* [ = *twips* ]

*Kalendar.PrintWidth* [ = *twips* ]

*Kalendar.PrintX* [ = *twips* ]

*Kalendar.PrintY* [ = *twips* ]

Write only.

## Description

Use these properties to specify where on the page you want a Kalendar printed. Use the PrintHDC and PrintAction properties to actually print it. The Kalendar is printed on the current printer.

## Data Type

Integer

# PrintHDC

Example

## Usage

*Kalendar.PrintHDC* [ = *Printer.HDC* ]

Write only.

## Description

Use this property to indicate the HDC on which to print a Kalendar. Generally, you will set this property to the Printer object's HDC (*Printer.HDC*).

## Data Type

Integer

## Note

This property should always be set just before using the *PrintAction* property.

# SelDayBackColor

## Usage

*Kalendar.SelDayBackColor*[ = *colorval*]

## Description

Sets the background color of the day box which is currently selected.

## Data Type

Color

## Notes

This property can be set in the DrawDay event.

# SelDayForeColor

## Usage

*Kalendar.SelDayForeColor*[ = *colorva*]

## Description

Sets the color of the text in the day box which is currently selected.

## Data Type

Color

## Notes

This property can be set in the DrawDay event.



# ShowAllDays

## Usage

*Kalendar.ShowAllDays*[ = *interval*]

## Description

When this value is set to False, only the days within the current month are shown (i.e. 1 through 31).  
When this value is set to True, the first and last week that are drawn in a month are 'padded' with days from the previous and next months.

## Data Type

Integer (True | False)

# ShowArrows

## Usage

*Kalendar.ShowArrows*[ = *interval*]

## Description

Sets whether or not the month navigation arrows are shown on the Kalendar. If the MonDispStyle is 0 (None), the arrows are not shown.

## Data Type

Integer (True | False)

# ShowLines

## Usage

*Kalendar.ShowLines*[ = *interval*]

## Description

Indicates whether or not lines are drawn around each day box.

## Data Type

Integer (True | False)

# ShowSelection

## Usage

*Kalendar.ShowSelection*[ = *interval*]

## Description

Indicates whether or not the currently selected day is shown in the SelDayBackColor and SelDayForeColor colors, or shown in the BackColor and ForeColor colors.

## Data Type

Integer (True | False)

# Text

## Usage

*Kalendar.Text*[ = *datetext*]

## Description

Sets the current date that the calendar is displaying. This property must be set with a string in the form "mm/dd/yyyy". An error is generated if the string is not a valid date string.

## Data Type

Text (in "mm/dd/yyyy" format)

# ClickDay

## Syntax

Sub ClickDay()

## Description

This event is fired whenever the currently selected day changes, either by the user clicking on a day , the Text property being set or the Julian property being set. The current date is automatically updated when the user clicks on a date. This event will not be fired if the user clicks on a part of the calendar that is not a day (e.g. on the month); however, the standard Click event will be fired.

# DblClickDay

## Syntax

```
Sub DblClickDay()
```

## Description

This event is fired whenever a user double clicks on a day. The current date is automatically updated when the user double clicks on a day. This event will not be fired if the user double-clicks on a part of the calendar that is not a day (e.g. on the month); however, the standard DblClick event will be fired.

# DrawDay

[Example 1](#)

[Example 2](#)

## Syntax

Sub Kalendar1\_DrawDay (hdc As Integer, State As Integer, theDay As Long, x As Single, y As Single, x2 As Single, y2 As Single, Cancel As Integer)

## Parameters

hdc	The device context on which to draw. As a general rule, if you are going to draw your own date box, you will need to use Window's API calls. Most of these calls require and hdc in order to do their magic.
State	0 - Normal 1 - Selected, no focus 2 - Selected, have focus 3 - Other month day.
theDay	This is the date that needs to be drawn. Use the Visual Basic function Format to extract any necessary information. For example, to get the day to draw use Format(theDay, "d").
x, y, x2, y2	Describes the rectangle that makes up the <u>day box</u> . These values are in twips. If you use GDI calls, you will need to convert these to pixels.
Cancel	Set this value to true if you have drawn the day box. If you do not set this value to true, the day will be drawn using the calendars property settings.

## Description

By responding to this event you can draw a day any way you want to. This event is fired only if the DateDisplay style is set to 2 (User).

There are two ways that you can respond to this event. The first way is the easiest and doesn't require any Windows API calls. When this event fires, you can change any of the properties that affect the display of a day box. For instance, you can change the BackColor property so that each day is drawn in a different color. The documentation indicates whether or not a property can be set during this event.

The Cancel should not be set to True if you draw the day box using this first method.

The second way to respond to this event is to draw the day box using Windows API calls and the hdc parameter. This is the most flexible way to draw your own Kalendar day box, providing you with the option of using any of the Windows GDI drawing calls. Refer to example 2 for one way of responding to this event.

When you use the second method of responding to this event, you must set the Cancel parameter to True, otherwise the VBX will draw over your drawing.

A third possible way to draw a day box is to ignore this event (thus letting the VBX draw each day box



using the property settings) and respond to the DrawOnDay event.

*Warning: Do not set break points in this function. If you need to debug the code in this event, try using `Debug.Print` statements.*

# DrawOnDay

## Example

## Syntax

Sub Kalendar1\_DrawOnDay (hdc As Integer, State As Integer, theDay As Long, x As Single, y As Single, x2 As Single, y2 As Single)

## Parameters

hdc	The device context on which to draw. As a general rule, if you are going to draw your own date box, you will need to use Window's API calls. Most of these calls require and hdc in order to do their magic.
State	0 - Normal 1 - Selected, no focus 2 - Selected, have focus 3 - Other month day.
theDay	This is the date that needs to be drawn. Use the Visual Basic function Format to extract any necessary information. For example, to get the day to draw use Format(theDay, "d").
x, y, x2, y2	Describes the rectangle that makes up the <u>day box</u> . These values are in twips. If you use GDI calls, you will need to convert these to pixels.

## Description

This event allows a you to draw information on a day box without having to draw it yourself (by responding to the DrawDay event). It is fired after the DrawDay event (when the DateDisplayStyle is User-Defined) and the DrawDay event does not set the Cancel parameter to True.

When this event fires, you can assume that the day box for theDay has been drawn (generally as a rectangle). You can then use Windows API calls with the hdc parameter to draw information on top of the day box.

# QueryChangeDay

Example

## Syntax

Sub Kalendar1\_QueryChangeDay(theDay as Long, Cancel as Integer)

## Description

This event is fired just before the ClickDay event. It allows you to stop the change to a new date from the current date. The parameter, theDay is the day that Kalendar wants to change to. By setting the Cancel parameter to True, you will prevent Kalendar from changing to the date theDay. This is useful if you want to limit the range of days a user can choose (i.e. >= today or weekdays only)

*Warning: Do not set break points in this function, it is not a pretty sight. If you need to debug the code in this event, try using Debug.Print statements.*

## DateAtPoint Example

Copy

This code responds to a MouseMove event, by determining which date the mouse is currently over. It determines the date by setting PointX and PointY to the mouse's X and Y coordinates, then referencing the DateAtPoint property.

```
Sub Kalendar1_MouseMove (Button As Integer, Shift As Integer,  
    X As Single, Y As Single)  
    Dim tmpDay As String  
    Kalendar1.PointX = X  
    Kalendar1.PointY = Y  
    tmpDay = Kalendar1.DateAtPoint  
    If tmpDay <> "" Then  
        Kalendar1.Text = Kalendar1.DateAtPoint  
    End If  
End Sub
```

# DrawDay Example 1

Copy

This example shows how to respond to the DrawDay Event by changing the background color of a day, depending on what day of the week the day is. Weekdays are colored in bright purple and weekends are colored in dark purple.

```
Sub Kalendar1_DrawDay (hdc As Integer, STATE As Integer,
    theDay As Long, x As Single,
    y As Single, x2 As Single,
    y2 As Single, Cancel As Integer)

Dim DOW As Integer
DOW = Format(theDay, "w")
Select Case STATE
Case KAL_STATE_NOT_SELECTED:
    If DOW = 1 Or DOW = 7 Then
        Kalendar1.DayBackColor = RGB(128, 0, 128)
    Else
        Kalendar1.DayBackColor = RGB(255, 0, 255)
    End If

    Kalendar1.DayForeColor = RGB(0, DOW * 30, 0)
Case KAL_STATE_SELECTED_WITH,
    KAL_STATE_SELECTED_WITHOUT:
    If DOW = 1 Or DOW = 7 Then
        Kalendar1.SelDayBackColor = RGB(255, 0, 0)
    Else
        Kalendar1.SelDayBackColor = RGB(192, 0, 0)
    End If
    Kalendar1.SelDayForeColor = RGB(0, DOW * 30, 0)
End Select
If DOW = 1 Or DOW = 7 Then
    Kalendar1.DayAlignment = 8
Else
    Kalendar1.DayAlignment = 0
End If
End Sub
```

## DrawDay Example 2

Copy

This example shows how to respond to the DrawDay Event by drawing the entire day box using Windows API calls. Instead of drawing rectangles for the days, it draws ellipses.

```
Sub Kalendar1_DrawDay (hdc As Integer, STATE As Integer, theDay As Long, x As Single, y As Single, x2 As Single, y2
As Single, Cancel As Integer)
Dim retval As Integer
Dim oldPen As Integer
Dim txtDay As String
Dim r As Rect
Dim oldBrush
Dim oldColor, oldTextColor
Dim lx As Long
txtDay = Format(theDay, "d")
r.left = x / Screen.TwipsPerPixelX
r.top = y / Screen.TwipsPerPixelY
r.right = x2 / Screen.TwipsPerPixelX
r.bottom = y2 / Screen.TwipsPerPixelY
oldPen = SelectObject(hdc, GetStockObject(BLACK_PEN))
Select Case STATE
Case KAL_STATE_SELECTED_WITHOUT:
oldBrush = SelectObject(hdc, GetStockObject(LTGRAY_BRUSH))
oldColor = SetBkColor(hdc, RGB(192, 192, 192))
oldTextColor = SetTextColor(hdc, 0)
Case KAL_STATE_SELECTED_WITH:
oldBrush = SelectObject(hdc, GetStockObject(LTGRAY_BRUSH))
oldColor = SetBkColor(hdc, RGB(192, 192, 192))
oldTextColor = SetTextColor(hdc, RGB(255, 0, 0))
Case KAL_STATE_NOT_SELECTED:
oldBrush = SelectObject(hdc, GetStockObject(WHITE_BRUSH))
oldColor = SetBkColor(hdc, RGB(255, 255, 255))
oldColor = SetTextColor(hdc, 0)
End Select
retval = Ellipse(hdc, r.left, r.top, r.right, r.bottom)
retval = DrawText(hdc, txtDay, Len(txtDay), r, DT_CENTER Or DT_VCENTER Or DT_SINGLELINE)
retval = SelectObject(hdc, oldPen)
retval = SelectObject(hdc, oldBrush)
lx = SetBkColor(hdc, oldColor)
lx = SetTextColor(hdc, oldTextColor)
Cancel = True
End Sub
```

## DrawOnDay Example

Copy

This code responds to a DrawOnDay event, by drawing one of seven bitmaps on the day to be drawn. To use this sample, copy this onto a form with a Kalendar on it, create an array of Picture controls with a different icon in each. Set the AutoRedraw property of the Picture controls to True.

```
Sub Kalendar1_DrawOnDay (hDC As Integer, STATE As Integer, theDay As Long, x As Single, y As Single, x2 As Single,
y2 As Single)
Dim retval As Integer
Dim r As Rect
Dim w As Long, h As Long
KalWindowAPIRect x, y, x2, y2, r
InflateRect r, -1, -1
r.top = r.bottom - Picture1(0).Height / Screen.TwipsPerPixelY
r.left = r.right - Picture1(0).Width / Screen.TwipsPerPixelX

w = Picture1(0).Width / Screen.TwipsPerPixelX
If w > r.right - r.left Then
    w = r.right - r.left
End If
h = Picture1(0).Height / Screen.TwipsPerPixelY
If h > r.bottom - r.top Then
    h = r.bottom - r.top
End If
retval = BitBlt(hDC, r.left, r.top, w, h, Picture1(theDay Mod 8).hDC, 0, 0, SRCAND)
End Sub
```

# QueryChangeDay Example

Copy

This code responds to the QueryChangeDay event by not allowing any day previous to today to be selected.

```
Sub Kalender2_QueryChangeDay (theDay As Long, Cancel As Integer)
    If theDay < Date Then
        Beep
        Cancel = True
    End If
End Sub
```



# Print Kalendar Example

Copy

This code shows how to print a Kalendar using the PrintX, PrintY, PrintWidth, PrintHeight and PrintHDC properties. To use it, create a form with a Kalendar and a button on it. Copy the following code into the buttons Click event.

```
Dim saveMonFontSize As Single
Dim saveBackColor As Long
    ' Save the current state of the Kalendar.
saveMonFontSize = Kalendar1.MonFontSize
saveBackColor = Kalendar1.MonBackColor
    ' Set some properties so that it looks good.
Kalendar1.MonFontSize = 14
Kalendar1.MonFontBold = True
Kalendar1.BorderStyle = 1
Kalendar1.MonBackColor = RGB(255, 255, 255)
    ' Position the paper two inches down and two inches
    ' in, in a 3" X 3" square.
Kalendar1.PrintX = 2880
Kalendar1.PrintY = 2880
Kalendar1.PrintWidth = 1440 * 3
Kalendar1.PrintHeight = 1440 * 3
Kalendar1.PrintHDC = Printer.hDC
Printer.Print "This is a report with a Kalendar"
Kalendar1.PrintAction = 3
Kalendar1.MonFontBold = False
Kalendar1.MonBackColor = saveBackColor
Kalendar1.BorderStyle = 0
Printer.EndDoc
```

# Glossary of Terms

day box

day of the week title

month navigation arrows

month title

## **day box**

The rectangle that makes up a single day on a Kalendar. This includes the bounding rectangle and the day number.

## **day of the week title**

The portion of the Kalendar that shows the days of the week (i.e. Monday, Tuesday, etc.)

## **month navigation arrows**

The arrows shown in the [month title](#) that allow a user to move to the next or previous month.

## **month title**

The portion of the Kalendar that shows the name of the current month and year.

The properties that you can change within this event are: DayAlignment, any color property for a day (e.g. ForeColor), and any font property for a day(e.g. FontName) .

