

Contents



The **ctDate** object is a visual representation of a monthly calendar.

This control presents a month of dates at a time and then allows the user to view and optionally select one of those dates. Options in the control allow for independent weekend, and date colors, monthly navigation buttons, and border options.

When a new date is selected on the calendar, the [Day](#), [Month](#), [Year](#), and [Date](#) properties will also change to reflect the value of the new date. Likewise, the appearance of the calendar can be changed by changing the value of any of those properties.

The calendar is divided into three main parts. The *Title* section is used to display the month, year, and monthly navigation buttons. The *Header* section includes the area used to display the [DayHeader](#) titles. The *Date* section is used to display the days of the month.



File Names

16 bit CTDAT16.OCX
32 bit CTDAT32.OCX

Class Name

ctDate

See Also :

[Changing Dates](#)

[Events](#)

[Methods](#)

[Properties](#)

Properties

ctDate Property	Data Type	Description
BackColor	long	Specifies the background color of the control.
BorderColor	long	Specifies the border color of the control.
BorderType	integer (enumerated)	Specifies the type of border to place around the control. Valid values include ... <ul style="list-style-type: none">• 0 - Regular• 1 - None• 2 - Raised• 3 - Lowered
Date	long	A numeric value representing the selected date. This value indicate the number of days since Jan. 1/1900.
DateBorder	integer (enumerated)	Specifies the type of border to place around the each date in the calendar. Valid values include ... <ul style="list-style-type: none">• 0 - Regular• 1 - None• 2 - Raised• 3 - Lowered
DateXSize	integer	Specifies the offset value used to calculate the display width of the calendar.
DateYSize	integer	Specifies the offset value used to calculate the display height of the calendar.
Day	integer	Specifies the selected day of the month on the calendar. (1 - 31)
DayHeader	string	Specifies the headers to be placed above each day of the week.
FirstDay	integer (enumerated)	Specifies the starting day for the calendar. Valid values include ... <ul style="list-style-type: none">• 0 - Sunday• 1 - Monday• 2 - Tuesday• 3 - Wednesday• 4 - Thursday• 5 - Friday• 6 - Saturday
FocusColor	long	Specifies the color used to paint the focus rectangle or background of the selected day of the month.

<u>FocusType</u>	integer (enumerated)	Specifies the border type used to place around the selected date. Valid values include ... <ul style="list-style-type: none"> • 0 - Regular • 1 - Flat • 2 - Raised • 3 - Lowered
<u>Font</u>	Font	Specifies the font used for all text within the control.
<u>ForeColor</u>	long	Specifies the color used to paint the text of the days of the month.
<u>KeyboardScan</u>	boolean	Specifies whether the dates can be changed by pressing a key on the keyboard.
<u>LevelColor</u>	long	Specifies the color used to paint the upper level of the calendar.
<u>LevelDepth</u>	integer (enumerated)	Specifies the depth for the <u>LevelColor</u> to be painted. Valid values include ... <ul style="list-style-type: none"> • 0 - None • 1 - Title • 2 - Header
<u>LevelOffset</u>	integer	Specifies the offset value to be placed on the calculated position of the <u>LevelDepth</u> .
<u>Month</u>	integer	Specifies the selected month of the year on the calendar (1 - 12)
<u>MonthButtons</u>	boolean	Specifies whether or not the monthly navigation buttons will be placed on the calendar.
<u>MonthNames</u>	string	Specifies the names for the different months.
<u>TitleAlign</u>	integer (enumerated)	Specifies the alignment for the title text. Valid values include ... <ul style="list-style-type: none"> • 0 - Left • 1 - Right • 2 - Center
<u>TitleBorder</u>	boolean	Specifies whether or not a border will be drawn around the calendars title.
<u>TitleColor</u>	long	Specifies the color used to paint the title text and monthly navigation buttons of the calendar.
<u>WeekendColor</u>	long	Specifies the color used to paint the background of the weekend dates.
<u>Year</u>	integer	Specifies the current year for the calendar.

Methods

ctDate Method	Return Type	Description
<u>Action</u>	void	Performs a certain action on the calendar.
<u>ClearDays</u>	void	Sets all day colors back to their default settings.
<u>DayColor</u>	long	Specifies a new background color for a certain day of the month.
<u>LastDay</u>	void	Set the current date back one day.
<u>LastMonth</u>	void	Set the current date back one month.
<u>LastYear</u>	void	Set the current date back one year.
<u>NextDay</u>	void	Set the current date forward one day.
<u>NextMonth</u>	void	Set the current date forward one month.
<u>NextYear</u>	void	Set the current date forward one year.
<u>Today</u>	void	Set the current date to the current system date.

Events

ctDate Event	Occurs
Click	Standard Event
<u>DateChange</u>	Occurs when a new date is selected
DbClick	Standard Event
KeyDown	Standard Event
KeyPress	Standard Event
KeyUp	Standard Event
MouseDown	Standard Event
MouseMove	Standard Event
MouseUp	Standard Event

Changing Dates

There are a variety of different ways that can be used to change the current date on the control. These include

- Use the mouse and click on the new date.

- If the [KeyboardScan](#) property is set to TRUE, the user can press one of the arrow or cursor keys when the calendar has focus. The arrow keys will move the current date forwards or backwards one day, the *page up/down* keys will move the date forwards or backwards one month, and the *home* and *end* keys will move the date forwards or backwards one year.
- Assign new values to the [Day](#), [Month](#), [Year](#), or [Date](#) properties.
- Assign a new value to the [Action](#) method, or use one of the other methods such as [LastDay](#), [LastMonth](#), [LastYear](#), [NextDay](#), [NextMonth](#), [NextYear](#), or [Today](#).

BackColor Property

Description

Specifies the background color of the control.

Syntax

```
[form.]ctDate.BackColor [ = color& ]
```

Data Type

Long (OLE_COLOR)

Example

```
ctDate.BackColor = RGB( 128, 128, 128 )
```

BorderColor Property

Description

Specifies the border color of the control.

Syntax

```
[form.]ctDate.BorderColor [ = color& ]
```

Data Type

Long (OLE_COLOR)

Example

```
ctDate.BorderColor = RGB( 128, 128, 128 )
```

BorderStyle Property

Description

Specifies the type of border to place around the control. Valid values include ...

- 0 - Regular : Regular border
- 1 - None : No border
- 2 - Raised : Raised 3D border
- 3 - Lowered : Lowered 3D border

Syntax

[*form*.]ctDate.**BorderStyle** [= *setting%*]

Data Type

Integer (enumerated)

Example

```
ctDate.BorderStyle = 2
```


Date Property

Description

A numeric value representing the selected date. This value indicate the number of days since Jan. 1/1900. The control can accept a a negative number if the calendar needs to display a date before Jan. 1 / 1900.

Syntax

```
[form.]ctDate.Date [ = date& ]
```

Data Type

Long

Example

```
ctDate.Date = 38232
```

DateBorder Property

Description

Specifies the type of border to place around the each date in the calendar. The *DateBorder* property is also used to determine the border around the title if the [TitleBorder](#) property is set to TRUE. Valid values include ...

- 0 - Regular : Regular border
- 1 - None : No border
- 2 - Raised : Raised 3D border
- 3 - Lowered : Lowered 3D border

Syntax

[*form.*]ctDate.**DateBorder** [= *setting%*]

Data Type

Integer (enumerated)

Example

ctDate.DateBorder = 2

DateXSize Property

Description

Specifies the offset value used to calculate the display width of the calendar. The drawing area for the height and width of the calendar are calculated based on the current size of the control. The programmer can change the calculated drawing width of the calendar by changing the value of this property.

Syntax

```
[form.]ctDate.DateXSize [ = setting% ]
```

Data Type

Integer

Example

```
ctDate.DateXSize = 4  
ctDate.DateXSize = -2
```

DateYSize Property

Description

Specifies the offset value used to calculate the display height of the calendar. The drawing area for the height and width of the calendar are calculated based on the current size of the control. The programmer can change the calculated drawing height of the calendar by changing the value of this property.

Syntax

```
[form.]ctDate.DateYSize [ = setting% ]
```

Data Type

Integer

Example

```
ctDate.DateYSize = 4  
ctDate.DateYSize = -2
```

Day Property

Description

Specifies the selected day of the month on the calendar. (1 - 31)

Syntax

```
[form.]ctDate.Day [ = setting% ]
```

Data Type

Integer

Example

```
ctDate.Day = 23
```

DayHeader Property

Description

Specifies the headers to be placed above each day of the week. Each day header should be separated by a semi-colon in order for the control to determine what text string belongs to what day. The first text string corresponds to Sunday, the second to Monday, etc.

Syntax

```
[form.]ctDate.DayHeader [ = string% ]
```

Data Type

string

Example

```
ctDate.DayHeader = "Sun;Mon;Tue;Wed;Thr;Fri;Sat"
```

FirstDay Property

Description

Specifies the starting day for the calendar. Valid values include ...

- 0 - Sunday
- 1 - Monday
- 2 - Tuesday
- 3 - Wednesday
- 4 - Thursday
- 5 - Friday
- 6 - Saturday

Syntax

```
[form.]ctDate.FirstDay [ = setting% ]
```

Data Type

Integer (enumerated)

Example

```
ctDate.FirstDay = 4
```

FocusColor Property

Description

Specifies the color used to paint the focus rectangle or background of the selected day of the month.

Syntax

```
[form.]ctDate.FocusColor [ = color& ]
```

Data Type

Long (OLE_COLOR)

Example

```
ctDate.FocusColor = RGB( 128, 128, 128 )
```


FocusType Property

Description

Specifies the border type used to place around the selected date. Valid values include ...

- 0 - Regular : A rectangular border
- 1 - Flat : No border, focus color paints the dates background.
- 2 - Raised : Raised border, focus color paints the dates background.
- 3 - Lowered : Lowered border, focus color paints the dates background.

Syntax

```
[form.]ctDate.FocusType [ = setting% ]
```

Data Type

Integer (enumerated)

Example

```
ctDate.FocusType = 1
```

Font Property

Description

Specifies the font used for all text within the control.

In order to set the font of the control, it must be either set using the font property page of the control, the property table of the host language (if available), or by assigning a font from another control with the same font type to this one.

Syntax

```
[form.]ctDate.Font [ = Font ]
```

Data Type

Font

Example

```
ctDate.Font = lbl_Text.Font
```

ForeColor Property

Description

Specifies the color used to paint the text of the days of the month.

Syntax

```
[form.]ctDate.ForeColor [ = color& ]
```

Data Type

Long (OLE_COLOR)

Example

```
ctDate.ForeColor = RGB( 128, 128, 128 )
```

KeyboardScan Property

Description

Specifies whether the dates can be changed by pressing a key on the keyboard.

Syntax

```
[form.]ctDate.KeyboardScan [ = { True | False } ]
```

Data Type

Boolean

Example

```
ctDate.KeyboardScan = TRUE
```

LevelColor Property

Description

Specifies the color used to paint the upper level of the calendar. This property will have no effect on the control if the [LevelDepth](#) property is set to 0.

Syntax

```
[form.]ctDate.LevelColor [ = color& ]
```

Data Type

Long (OLE_COLOR)

Example

```
ctDate.LevelColor = RGB( 128, 128, 128 )
```

LevelDepth Property

Description

Specifies the depth for the [LevelColor](#) to be painted. It is possible to use two different colors to paint the background of the calendar. The *LevelDepth* property is used to determine what portion of the calendar will be painted with the second color. Valid values include ...

- 0 - None : No extra coloring
- 1 - Title : Paint the background of the title
- 2 - Header : Paint the background of the title and header

Syntax

```
[form.]ctDate.LevelDepth [ = setting% ]
```

Data Type

Integer (enumerated)

Example

```
ctDate.LevelDepth = 1
```

LevelOffset Property

Description

Specifies the offset value to be placed on the calculated position of the [LevelDepth](#) height.

Syntax

```
[form.]ctDate.LevelOffset [ = setting% ]
```

Data Type

Integer

Example

```
ctDate.LevelOffset = 2
```

Month Property

Description

Specifies the selected month of the year on the calendar. (1 - 12)

Syntax

```
[form.]ctDate.Month [ = setting% ]
```

Data Type

Integer

Example

```
ctDate.Month = 12
```


MonthButtons Property

Description

Specifies whether or not the monthly navigation buttons will be placed on the calendar. These buttons will allow the user to select the next or previous months date page by clicking on one of them.

Syntax

```
[form.]ctDate.MonthButtons [= { True | False }]
```

Data Type

Boolean

Example

```
ctDate.MonthButtons = TRUE
```

MonthNames Property

Description

Specifies the names for the different months. Each month should be separated by a semi-colon in order for the control to determine what text string belongs to what month. The first text string corresponds to January, the second to February, etc.

Syntax

```
[form.]ctDate.MonthNames [ = string% ]
```

Data Type

string

Example

```
ctDate.MonthNames = "Jan;Feb;Mar;Apr;May;Jun;Jul;Aug;Sep;Oct;Nov;Dec"
```

TitleAlign Property

Description

Specifies the alignment for the title text. Valid values include ...

- 0 - Left
- 1 - Right
- 2 - Center

Syntax

```
[form.]ctDate.TitleAlign [ = setting% ]
```

Data Type

Integer (enumerated)

Example

```
ctDate.TitleAlign = 1
```

TitleBorder Property

Description

Specifies whether or not a border will be drawn around the calendars title. The border type is determined by the [DateBorder](#) property.

Syntax

```
[form.]ctDate.TitleBorder [= { True | False } ]
```

Data Type

Boolean

Example

```
ctDate.TitleBorder = TRUE
```

TitleColor Property

Description

Specifies the color used to paint the title text and monthly navigation buttons of the calendar.

Syntax

```
[form.]ctDate.TitleColor [ = color& ]
```

Data Type

Long (OLE_COLOR)

Example

```
ctDate.TitleColor = RGB( 128, 128, 128 )
```

WeekendColor Property

Description

Specifies the background color of each weekend date in the calendar. A value of -1 will cause the weekend dates to be colored with the same color as the [BackColor](#) property.

Syntax

```
[form.]ctDate.WeekendColor [ = color& ]
```

Data Type

Long (OLE_COLOR)

Example

```
ctDate.WeekendColor = RGB( 128, 128, 128 )
```

Year Property

Description

Specifies the current year for the calendar.

Syntax

```
[form.]ctDate.Year [ = setting% ]
```

Data Type

Integer

Example

```
ctDate.Year = 1996
```

Action Method

Description

Performs a certain action on the calendar. The actions that this method can trigger include the [ClearDays](#), [LastDay](#), [LastMonth](#), [LastYear](#), [NextDay](#), [NextMonth](#), [NextYear](#), and [Today](#) methods. Because this method simply triggers other methods, it is somewhat redundant, and has only been included for compatibility between this control, and its VBX equivalent.

An action method is triggered by assigning a number to the *Action* method. Valid values include...

- 1 - Move to the next day
- 2 - Move to the next month
- 3 - Move to the next year
- 4 - Move to the last day
- 5 - Move to the last month
- 6 - Move to the last year
- 7 - Move to the current system data
- 8 - Clear the date buffers

Return Type

Void

Parameters

None

Example

```
ctDate.Action = 1
```


ClearDays Method

Description

Sets all day colors back to their default settings. These are the colors that were set using the [DayColor](#) method.

Return Type

Void

Parameters

None

Example

```
ctDate.ClearDays
```

DayColor Method

Description

Specifies a new background color for a certain day of the month. This property method will take precedence over a date colored with the [WeekendColor](#) property, but not with selected dates colored with the [FocusColor](#) property.

While the *DayColor* method can be used in any script, the easiest way to color certain dates would be to place this method in the [DateChange](#) event.

Syntax

```
[form.]ctDate.DayColor( index% ) [ = color& ]
```

Return Type

Long (OLE_COLOR)

Parameters

- 1 - Day : (integer) The day of the month
- 2 - Color : (Long) The new background color for the date

Example

```
ctDate.DayColor( 5 ) = RGB( 128, 0, 128 )  
newColor = ctDate.DayColor( 14 )
```

The following example would be placed in the [DateChange](#) event to color Christmas ...

```
if ( nMonth = 12 ) Then  
    ctDate.DayColor( 25 ) = RGB( 0, 255, 0 )  
End If
```

LastDay Method

Description

Set the current date back one day. If the selected date is the first day of the month, the calendar will display a new date page when it moves to the last day.

Syntax

```
[form.]ctDate.LastDay
```

Return Type

Void

Parameters

None

Example

```
ctDate.LastDay
```

LastMonth Method

Description

Set the current date back one month. This event will cause a new date page to be drawn.

Syntax

```
[form.]ctDate.LastMonth
```

Return Type

Void

Parameters

None

Example

```
ctDate.LastMonth
```

LastYear Method

Description

Set the current date back one year. This event will cause a new date page to be drawn.

Syntax

```
[form.]ctDate.LastYear
```

Return Type

Void

Parameters

None

Example

```
ctDate.LastYear
```

NextDay Method

Description

Set the current date forward one day. If the selected date is the last day of the month, the calendar will display a new date page when it moves to the next day.

Syntax

```
[form.]ctDate.NextDay
```

Return Type

Void

Parameters

None

Example

```
ctDate.NextDay
```

NextMonth Method

Description

Set the current date forward one month. This event will cause a new date page to be drawn.

Syntax

```
[form.]ctDate.NextMonth
```

Return Type

Void

Parameters

None

Example

```
ctDate.NextMonth
```

NextYear Method

Description

Set the current date forward one year. This event will cause a new date page to be drawn.

Syntax

```
[form.]ctDate.NextYear
```

Return Type

Void

Parameters

None

Example

```
ctDate.NextYear
```


Today Method

Description

Set the current date to the current system date.

Syntax

```
[form.]ctDate.Today
```

Return Type

Void

Parameters

None

Example

```
ctDate.Today
```

DateChange Event

Description

Occurs when a new date is selected.

Parameters

This event has four parameters sent to it. They include ...

- 1 - nDOW (integer) The new day of the week
- 2 - nDay (integer) The new day of the month
- 3 - nMonth (integer) The new month of the year
- 4 - nYear(integer) The new year

Notes

This event is a good location to place code that will color different dates with the [DayColor](#) method.

