

---

ComponentOne

# **EventsCalendar for ASP.NET Web Forms**

Copyright © 1987-2015 GrapeCity, Inc. All rights reserved.

**ComponentOne, a division of GrapeCity**

201 South Highland Avenue, Third Floor  
Pittsburgh, PA 15206 USA

**Website:** <http://www.componentone.com>  
**Sales:** [sales@componentone.com](mailto:sales@componentone.com)  
**Telephone:** 1.800.858.2739 or 1.412.681.4343 (Pittsburgh, PA USA Office)

**Trademarks**

The ComponentOne product name is a trademark and ComponentOne is a registered trademark of GrapeCity, Inc. All other trademarks used herein are the properties of their respective owners.

**Warranty**

ComponentOne warrants that the media on which the software is delivered is free from defects in material and workmanship, assuming normal use, for a period of 90 days from the date of purchase. If a defect occurs during this time, you may return the defective media to ComponentOne, along with a dated proof of purchase, and ComponentOne will replace it at no charge. After 90 days, you can obtain a replacement for the defective media by sending it and a check for \$25 (to cover postage and handling) to ComponentOne.

Except for the express warranty of the original media on which the software is delivered is set forth here, ComponentOne makes no other warranties, express or implied. Every attempt has been made to ensure that the information contained in this manual is correct as of the time it was written. ComponentOne is not responsible for any errors or omissions. ComponentOne's liability is limited to the amount you paid for the product. ComponentOne is not liable for any special, consequential, or other damages for any reason.

**Copying and Distribution**

While you are welcome to make backup copies of the software for your own use and protection, you are not permitted to make copies for the use of anyone else. We put a lot of time and effort into creating this product, and we appreciate your support in seeing that it is used by licensed users only.

# Table of Contents

Overview .....	15
Help with ASP.NET Web Forms Edition .....	15
Key Features.....	15
Top Tips.....	16
Quick Start .....	16
Step 1 of 3: Adding C1EventsCalendar to the Page .....	16
Step 2 of 3: Customizing the Events Calendar .....	17
Step 3 of 3: Viewing the Events Calendar .....	17
C1EventsCalendar Design-Time Support .....	18
Smart Tag .....	18
Context Menu .....	20
Appearance.....	20
C1EventsCalendar Elements .....	21
Built-in Themes .....	22
ThemeRoller Overview .....	24
Working with EventsCalendar for ASP.NET Web Forms .....	25
Views.....	25
Changing the View .....	27
Customizing the View .....	29
Data and DataBinding .....	31
Binding Data to a C1EventsCalendar Control Tutorial .....	34
Configuring the AccessDataSource Control .....	35
Mapping to the Data Storage.....	42
Export Service .....	45
Events.....	49
Working with the Client-Side .....	50
Client-Side Events .....	50
Task-Based Help.....	51
Working with Themes.....	52
Using a Built-In Theme .....	52
Using ThemeRoller for Visual Studio .....	54
Using a Custom Theme .....	54

Changing the Start Date of the C1EventsCalendar .....	55
Changing the Culture of a Schedule.....	56
Hiding the Header Bar.....	57
Hiding the Navigation Bar.....	58
Client-Side Reference.....	59
Using the Wijmo CDN .....	60
C1.Web.Wijmo.Controls.4 Assembly .....	61
C1.Web.Wijmo.Controls Namespace .....	61
Overview .....	61
Classes.....	62
C1BaseStateManager.....	62
Overview .....	62
Members.....	63
C1TargetControlBase .....	63
Overview .....	64
Members.....	65
Methods.....	70
GetProperty<V> Method .....	71
GetScriptDescriptors Method .....	72
GetScriptReferences Method .....	73
RegisterDesignTimeStyleSheet Method .....	73
SetProperty<V> Method .....	74
Properties.....	75
CDNDependencyPaths Property .....	78
CDNPath Property.....	79
DisplayVisible Property .....	80
EnableCombinedJavaScripts Property .....	81
Enabled Property .....	82
Theme Property .....	82
ThemeSwatch Property .....	83
UseCDN Property .....	84
WijmoControlMode Property .....	85
WijmoCssAdapter Property .....	86
WijChartNavigator .....	87

Overview .....	88
Members.....	88
WijChartNavigator Constructor .....	89
C1.Web.Wijmo.Controls.C1EventsCalendar Namespace .....	89
Overview .....	89
Classes.....	91
C1CalendarStorage .....	91
Overview .....	92
Members.....	93
C1CalendarStorage Constructor .....	94
Properties.....	94
DataMember Property.....	95
DataSourceID Property .....	96
Mappings Property .....	96
C1CalendarStorageMappings .....	98
Overview .....	98
Members.....	99
Properties.....	101
ColorMapping Property .....	102
DescriptionMapping Property .....	103
LocationMapping Property .....	104
NameMapping Property .....	105
PropertiesMapping Property .....	106
TagMapping Property .....	107
C1EventCollection.....	108
Overview .....	109
Members.....	109
C1EventCollection Constructor.....	112
Methods.....	112
Clear Method .....	114
C1EventsCalendar .....	115
Overview .....	116
Members.....	117
C1EventsCalendar Constructor .....	127

Methods.....	128
FindControl Method.....	130
LoadLayout Method.....	130
LoadLayout(String) Method .....	131
LoadLayout(Stream) Method.....	132
LoadLayout(String,LayoutType) Method .....	132
LoadLayout(Stream,LayoutType) Method .....	133
LoadPostData Method .....	134
RaisePostBackEvent Method .....	135
RaisePostDataChangedEvent Method .....	136
SaveLayout Method .....	136
SaveLayout(String) Method .....	136
SaveLayout(Stream) Method .....	137
Properties.....	138
Colors Property .....	145
Culture Property .....	146
CultureCalendar Property .....	147
DataStorage Property .....	148
DatePagerLocalization Property .....	149
DayHeaderFormat Property.....	150
DayViewHeaderFormat Property.....	151
EnableLogs Property .....	152
EnsureEventDialogOnBody Property .....	152
EventDialogTemplate Property.....	153
EventTemplate Property.....	154
EventTitleFormat Property .....	155
FirstRowDayHeaderFormat Property .....	156
HeaderBarVisible Property .....	157
Localization Property .....	157
NavigationBarVisible Property.....	158
OnClientBeforeAddEvent Property.....	159
OnClientBeforeDeleteCalendar Property .....	160
OnClientBeforeDeleteEvent Property.....	161
OnClientBeforeEditEventDialogShow Property .....	162

OnClientBeforeSaveCalendar Property .....	163
OnClientBeforeUpdateEvent Property .....	164
OnClientCalendarsChanged Property .....	165
OnClientEventCreating Property .....	165
OnClientInitialized Property.....	166
OnClientSelectedDatesChanged Property .....	167
OnClientViewTypeChanged Property .....	168
SelectedDate Property .....	168
SelectedDates Property .....	169
TimeInterval Property .....	170
TimeIntervalHeight Property .....	171
TimeRulerFormat Property .....	172
TimeRulerInterval Property .....	172
Views Property .....	173
VisibleCalendars Property .....	174
WebServiceUrl Property .....	175
C1EventsCalendarSerializer .....	176
Overview .....	176
Members.....	177
C1EventsCalendarSerializer Constructor .....	178
C1EventsCalendarStorage.....	178
Overview .....	179
Members.....	180
Methods.....	181
AddCalendar Method.....	183
AddEvent Method.....	183
AddEvent(Event) Method .....	184
AddEvent(Hashtable) Method .....	184
BeginUpdate Method.....	185
DeleteCalendar Method .....	186
DeleteEvent Method.....	186
DeleteEvent(Event) Method .....	187
DeleteEvent(Hashtable) Method .....	187
EndUpdate Method .....	188

Export Method .....	188
Export(String,FileFormatEnum) Method .....	189
Export(Stream,FileFormatEnum) Method .....	190
FindEventByID Method .....	191
Import Method .....	191
Import(String,FileFormatEnum) Method.....	192
Import(Stream,FileFormatEnum) Method.....	193
Initialize Method .....	194
LoadData Method .....	194
SaveData Method .....	194
ToString Method .....	195
UpdateCalendar Method .....	195
UpdateEvent Method .....	196
UpdateEvent(Event) Method .....	196
UpdateEvent(Hashtable) Method.....	197
Properties.....	198
Calendars Property .....	198
CalendarStorage Property.....	199
DataFile Property .....	200
Events Property.....	201
EventStorage Property.....	201
C1EventsCalendarXmlExchanger .....	202
Overview .....	203
Members.....	203
C1EventsCalendarXmlExchanger Constructor .....	204
Methods.....	205
Export Method .....	205
Export(String) Method .....	206
Export(Stream) Method.....	207
Export(String,IList<Event>) Method .....	207
Export(Stream,IList<Event>) Method .....	208
Import Method .....	209
Import(String) Method.....	210
Import(Stream) Method .....	210



ReadStringValue Method.....	211
Properties.....	212
Calendars Property .....	212
Events Property.....	212
C1EventStorage.....	213
Overview .....	214
Members.....	214
C1EventStorage Constructor.....	215
Properties.....	216
DataMember Property.....	216
DataSourceID Property .....	217
Mappings Property .....	218
C1EventStorageMappings.....	219
Overview .....	220
Members.....	221
C1EventStorageMappings Constructor.....	223
Properties.....	223
CalendarMapping Property.....	225
ColorMapping Property .....	226
DescriptionMapping Property .....	227
EndMapping Property .....	228
LocationMapping Property .....	229
PropertiesMapping Property .....	230
StartMapping Property .....	231
SubjectMapping Property .....	232
TagMapping Property .....	233
C1MappingsBase.....	234
Overview .....	235
Members.....	235
C1MappingsBase Constructor.....	236
C1MappingsBase Constructor() .....	237
C1MappingsBase Constructor(StateBag).....	237
Methods.....	238
Dispose Method .....	238

FindByMappingName Method.....	239
ToArray Method.....	240
Properties.....	240
IdMapping Property .....	240
Events.....	241
Disposed Event.....	242
DatePagerLocalizationOption .....	242
Overview .....	243
Members.....	243
DatePagerLocalizationOption Constructor .....	245
Methods.....	246
ToString Method .....	246
Properties.....	247
customViewDayTooltipFormat Property .....	249
customViewDayUnitTooltip2MonthesFormat Property.....	249
customViewMonthTooltip2YearsFormat Property .....	250
customViewMonthTooltipFormat Property .....	251
customViewSingleDayTooltipFormat Property .....	252
customViewSingleMonthTooltipFormat Property.....	253
customViewSingleYearTooltipFormat Property .....	254
customViewUnitDayLabelFormat Property .....	254
customViewUnitDayLabelFormat2Months Property .....	255
customViewUnitMonthLabelFormat Property .....	256
customViewUnitMonthLabelFormat2Years Property .....	257
customViewUnitSingleDayLabelFormat Property .....	258
customViewUnitSingleMonthLabelFormat Property.....	258
customViewUnitSingleYearLabelFormat Property .....	259
customViewUnitYearLabelFormat Property .....	260
customViewYearTooltipFormat Property .....	261
dayViewLabelFormat Property .....	262
dayViewTooltipFormat Property .....	262
monthViewLabelFormat Property .....	263
monthViewTooltipFormat Property .....	264
weekViewLabelFormat Property .....	265

weekViewTooltip2MonthesFormat Property .....	265
weekViewTooltipFormat Property.....	266
DayViewHeaderFormatOption .....	267
Overview .....	268
Members.....	268
DayViewHeaderFormatOption Constructor .....	269
Methods.....	270
ToString Method .....	270
Properties.....	271
Custom Property .....	271
Day Property .....	272
List Property .....	273
Week Property .....	274
Event .....	274
Overview .....	275
Members.....	276
Event Constructor .....	278
Event Constructor().....	278
Event Constructor(Hashtable) .....	279
Methods.....	280
CopyFrom Method .....	280
FromXml Method .....	281
PropsToXml Method .....	281
ToXml Method .....	282
Properties.....	282
Allday Property .....	284
Calendar Property .....	284
Color Property.....	285
Data Property.....	285
Description Property .....	286
End Property .....	286
Id Property .....	287
Location Property.....	288
ParentRecurrenceId Property .....	288

RecurrencePattern Property .....	289
RecurrenceState Property.....	289
Start Property.....	290
Subject Property .....	290
Tag Property.....	291
Fields .....	291
_allDayEvent Field.....	292
_color Field.....	292
_description Field.....	293
_end Field.....	293
_start Field .....	294
_subject Field .....	294
LocalizationOption .....	294
Overview .....	295
Members.....	296
LocalizationOption Constructor .....	300
Methods.....	300
ToString Method .....	301
Properties.....	301
activityCreatingCalendar Property.....	305
activityCreatingEvent Property .....	306
activityDeletingCalendar Property.....	307
activityDeletingEvent Property .....	307
activityLoading Property .....	308
activityUpdating Property.....	309
activityUpdatingCalendar Property .....	310
activityUpdatingEvent Property.....	310
agendaHeaderFullDateFormat Property .....	311
agendaLoadingMoreEvents Property .....	312
agendaMoreEventsFormat Property .....	313
agendaTimeFormat Property.....	313
buttonCancel Property.....	314
buttonClearAll Property.....	315
buttonClose Property.....	316

buttonDayView Property .....	317
buttonDelete Property.....	317
buttonListView Property .....	318
buttonMonthView Property .....	319
buttonOK Property.....	320
buttonSave Property .....	320
buttonToday Property .....	321
buttonWeekView Property .....	322
calendarNextTooltip Property .....	323
calendarPrevTooltip Property.....	323
calendarTitleFormat Property.....	324
calendarToolTipFormat Property.....	325
customMonthViewHeaderFormat Property.....	326
customMonthViewSecondTitleFormat Property.....	326
customSingleMonthViewHeaderFormat Property .....	327
customSingleWeekViewHeaderFormat Property .....	328
customSingleYearViewHeaderFormat Property .....	329
customWeekViewHeaderFormat Property.....	330
customYearViewHeaderFormat Property.....	331
customYearViewMonthFormat Property .....	331
customYearViewSecondTitleFormat Property.....	332
dayDetailsLabelFulldateFormat Property .....	333
dayDetailsLabelYearFormat Property .....	334
labelAllDay Property .....	335
labelCalendar Property .....	335
labelCalendarName Property.....	336
labelColor Property .....	337
labelDescription Property .....	338
labelEnds Property.....	338
labelLocation Property.....	339
labelName Property .....	340
labelRepeat Property .....	341
labelStarts Property .....	341
labelToday Property.....	342

logCouldntOpenLocalStorage Property .....	343
messageEndOccursBeforeStart Property .....	344
monthCellMoreEventsFormat Property .....	345
monthViewHeaderFormat Property .....	345
navigatorBarNextTooltip Property.....	346
navigatorBarPrevTooltip Property .....	347
promptOpenOccurrenceFormat Property .....	348
repeatDaily Property.....	348
repeatMonthly Property .....	349
repeatNone Property .....	350
repeatWeekly Property .....	351
repeatWorkDays Property .....	351
repeatYearly Property .....	352
textNewEvent Property .....	353
titleEditCalendar Property .....	354
weekViewHeaderFormat Property .....	354
weekViewHeaderFormat2Months Property.....	355
MappingInfo.....	356
Overview .....	357
Members.....	358
MappingInfo Constructor .....	359
Methods.....	360
ToString Method .....	360
Properties.....	361
DefaultValue Property .....	361
IsMapped Property .....	362
MappingName Property .....	362
PropertyName Property.....	363
Required Property.....	364
RecurrencePattern.....	364
Overview .....	365
Members.....	366
RecurrencePattern Constructor.....	369
RecurrencePattern Constructor() .....	369

RecurrencePattern Constructor(String,Hashtable).....	370
Properties.....	370
Data Property .....	373
DayOfMonth Property .....	374
dayOfWeekMask Property.....	375
EndTime Property .....	375
Exceptions Property .....	376
Instance Property.....	376
Interval Property .....	377
MonthOfYear Property .....	378
NoEndDate Property .....	378
Occurrences Property .....	379
ParentRecurrenceId Property .....	380
PatternEndDate Property .....	380
PatternStartDate Property.....	381
RecurrenceType Property .....	381
RemovedOccurrences Property .....	382
StartTime Property.....	382
View .....	383
Overview .....	384
Members.....	384
View Constructor .....	385
View Constructor(C1EventsCalendarViewType).....	386
View Constructor(C1EventsCalendarViewType,Boolean).....	386
View Constructor() .....	387
Properties.....	387
Count Property.....	388
IsActive Property.....	389
Name Property.....	390
Type Property.....	390
Unit Property .....	391
Enumerations.....	392
C1EventsCalendarCustomViewUnit.....	392
C1EventsCalendarViewType .....	393

FileFormatEnum.....	394
RecurrencePattern.RecurrenceTypeEnum .....	395
RecurrencePattern.WeekDaysEnum .....	396
RecurrencePattern.WeekOfMonthEnum .....	397



# Overview

**EventsCalendar for ASP.NET Web Forms** is a fully functional schedule that allows users to add, edit, and manage their appointments.

## Getting Started

To get started, review the following topics:

- [Quick Start](#)
- [Design-Time Support](#)
- [Task-Based Help](#)

# Help with ASP.NET Web Forms Edition

## Getting Started

For information on installing **ComponentOne Studio ASP.NET Web Forms Edition**, licensing, technical support, namespaces and creating a project with the control, please visit [Getting Started with Studio for ASP.NET Web Forms Edition](#).

# Key Features

The following are some of the main features of [C1EventsCalendar](#) that you may find useful:

- **Microsoft Outlook-style Appointments**  
Schedule one-time, all day, or recurring appointments with reminders, labels, and availability status. Stay organized with categories, resources, and contacts.
- **Built-in Data Views**  
Just set one property to change the data view for the scheduler. The data views include day, work week, week in vertical format, week in horizontal format, and month.
- **Built-in Data Layer**  
Quickly load and save data to and from your scheduling application. **EventsCalendar** provides a built-in data storage whether you are using standard ADO.NET databinding or EventsCalendar's generated XML file to save and load data.
- **Universal .NET Data Binding**  
EventsCalendar for ASP.NET Web Forms can bind to a variety of data sources including ADO.NET, OLEDB, and SQL Server with little or no code.
- **Rich Client-side Object Model**  
**EventsCalendar** uses client-side rendering technology to increase performance in your Web forms.
- **Importing and Exporting**  
Import and export data from and to iCalendar, XML, and PDF/Image formats using the [Import](#) and [Export](#) methods.
- **Localized Scheduling**  
Localize text and dates in your schedule as well as view tasks and calendars in the local or a defined time zone.

- **CSS Support**

Use a cascading style sheet (CSS) style to define custom skins. CSS support allows you to match the tab control to your organization's standards.

## Top Tips

The following tips may help you troubleshoot when working with EventsCalendar for ASP.NET Web Forms .

### **Tip 1: Upgrading C1EventsCalendar**

You need remove the **c1evcalservice.ashx** file from your project before upgrading **C1EventsCalendar** to a new version. A new version of this file will be automatically created when the control is initialized for the first time.

## Quick Start

In this quick start guide, you will explore the functionality of the [C1EventsCalendar](#) control by creating a simple customized events calendar.

### **See Also**

[Step 1 of 3: Adding C1EventsCalendar to the Page](#)

[Step 2 of 3: Customizing the Events Calendar](#)

[Step 3 of 3: Viewing the Events Calendar](#)

## Step 1 of 3: Adding C1EventsCalendar to the Page

In this step you will begin by adding the [C1EventsCalendar](#) control to the page.

To begin, complete the following steps:

1. Create a new ASP.NET Website project.
2. Click the **Design** tab located below the Document window to switch to Design view.
3. Navigate to the Visual Studio Toolbox and double-click **C1EventsCalendar** to add the control to the page.

### **See Also**

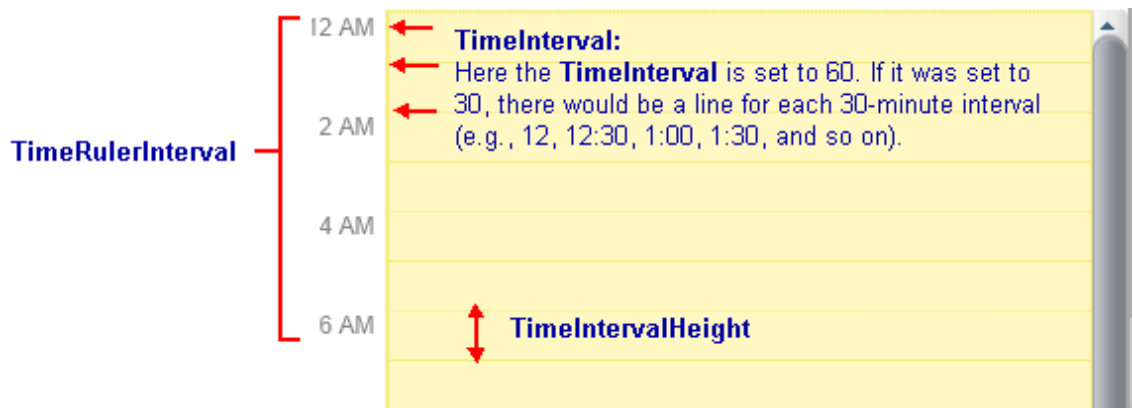
[Step 1 of 3: Adding C1EventsCalendar to the Page](#)

## Step 2 of 3: Customizing the Events Calendar

In this step you will set some properties to change the size of the events calendar and determine the time intervals shown on the calendar. Complete the following steps:

1. Click **C1EventCalendar1**'s smart tag to open the **C1EventsCalendar Tasks** menu and set the **Control width** property to **800px** and the **Control height** property to **500px**.
2. In the Visual Studio Properties window, set the following properties to the specified values:

Property	Value	Description
TimeInterval	60	This property specifies the time interval for the Day view, in minutes.
TimeIntervalHeight	25	This property specifies the Day view time interval row height, in pixels.
TimeRulerInterval	120	This property specifies the time ruler interval for the Day view, in minutes.



3.

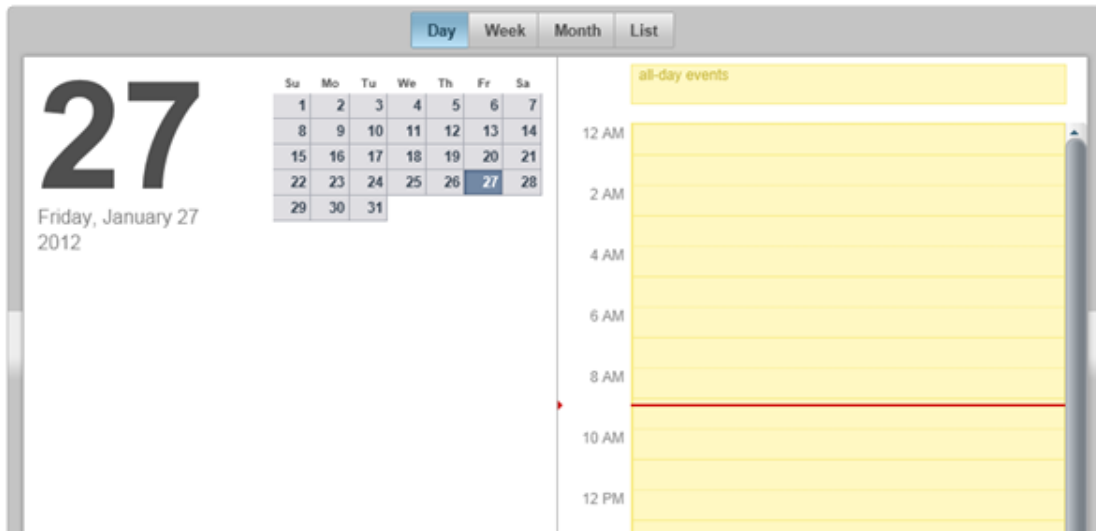
### See Also

[Step 2 of 3: Customizing the Events Calendar](#)

## Step 3 of 3: Viewing the Events Calendar

In this step you will run the application and view the events calendar where you can start adding events.

Press **F5** to run the application and view the events calendar.



Double-click a time to add a new event.

## See Also

[C1EventsCalendar Design-Time Support](#)

# C1EventsCalendar Design-Time Support


The following sections describe how to use **C1EventsCalendar's** design-time environment to configure the [C1EventsCalendar](#) control.

## See Also

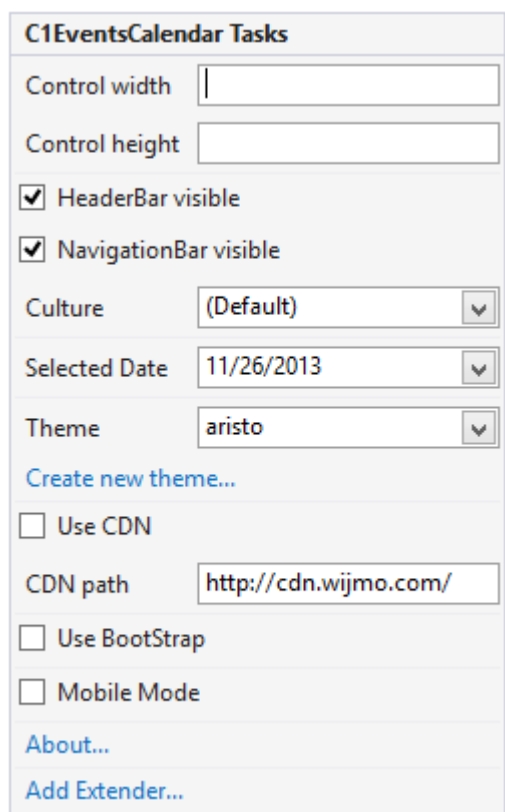
[Smart Tag](#)

[Context Menu](#)

## Smart Tag

The [C1EventsCalendar](#) control includes a smart tag (  ) in Visual Studio. A smart tag represents a shortcut tasks menu that provides the most commonly used properties in [C1EventsCalendar](#).

To access the **C1EventsCalendar Tasks** menu, click the smart tag in the upper-right corner of the [C1EventsCalendar](#) control.



The screenshot shows the 'C1EventsCalendar Tasks' menu. It contains the following elements:

- Control width**: A text input field.
- Control height**: A text input field.
- HeaderBar visible**: A checked checkbox.
- NavigationBar visible**: A checked checkbox.
- Culture**: A dropdown menu showing '(Default)'.
- Selected Date**: A dropdown menu showing '11/26/2013'.
- Theme**: A dropdown menu showing 'aristo'.
- Create new theme...**: A blue link.
- Use CDN**: An unchecked checkbox.
- CDN path**: A text input field containing 'http://cdn.wijmo.com/'.
- Use BootStrap**: An unchecked checkbox.
- Mobile Mode**: An unchecked checkbox.
- About...**: A blue link.
- Add Extender...**: A blue link.

The **C1EventsCalendar Tasks** menu operates as follows:

- **Control width**  
Enter a width, in pixels, for the calendar in the **Control width** textbox.
- **Control height**  
Enter a height, in pixels, for the calendar in the **Control height** textbox.
- **HeaderBar visible**  
Check the **HeaderBar visible** checkbox to make the header bar area of the calendar, which contains tabs for the calendar views, visible.
- **NavigationBar visible**  
Check the **NavigationBar visible** checkbox to make the navigation bar, which appears at the bottom of the calendar and allows you to jump to today's date or choose another date in the month, visible.
- **Culture**  
Click the **Culture** drop-down arrow to select the culture used to display the calendar.
- **Selected Date**  
Click the **Selected Date** drop-down arrow and select a date from the calendar. This is the date that will be shown in the [C1EventsCalendar](#) at run time.
- **Theme**  
Sets the theme to one of the built-in skins.
- **Create new theme...**  
The **Create new theme...** option opens **ThemeRoller for Visual Studio**. This allows you to create a customized theme without leaving your development environment. To find more information on using ThemeRoller in your application, see [ThemeRoller for Visual Studio](#).
- **Use CDN**  
Determines whether the control is using the CDN for the client-side reference.

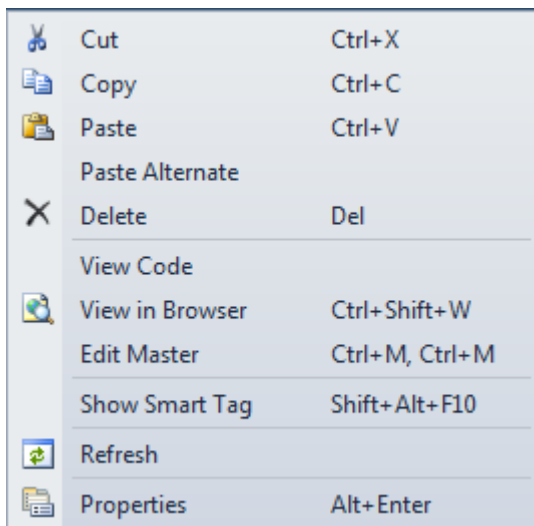
- **CDN Path**  
The path to the CDN library you are using.
- **Use Bootstrap**  
Selecting the **Use Bootstrap** option applies Bootstrap theming to your control. To find more information on using **Bootstrap** theming in your application, see [Bootstrap theming](#).
- **About**  
Clicking **About** opens a dialog box that displays the version number and licensing information for **ASP.NET Web Forms Edition**.

## See Also

[Context Menu](#)

# Context Menu

The **EventsCalendar for ASP.NET Web Forms** control provides a context menu for additional functionality to use at design time. Right-click the C1EventsCalendar control to open the following context menu:



## See Also

[Appearance](#)

# Appearance

The following topics show you the elements that make up the [C1EventsCalendar](#), its six premium built-in themes (Arctic, Midnight, Aristo, Rocket, Cobalt, and Sterling), and how to use ThemeRoller from jQuery UI to create your own customized theme!

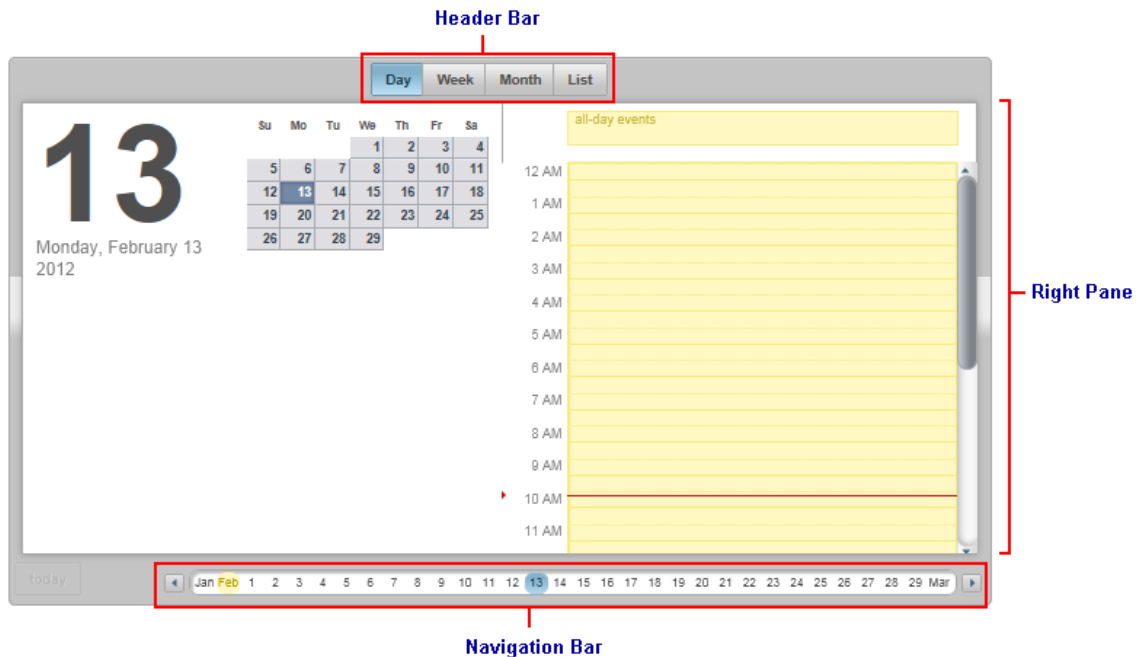
## See Also

[C1EventsCalendar Elements](#)

[Built-in Themes](#)[ThemeRoller Overview](#)

# C1EventsCalendar Elements

The [C1EventsCalendar](#) controls' user interface consists of the following areas:



## Header Bar

The **Header** bar appears at the top of the events calendar, by default, and allows users to switch between *Day*, *Week*, *Month*, and *List* views at run time.

## Navigation Bar

The **Navigation** bar appears at the bottom of the events calendar, by default, at run time. Users can advance through the days and months by clicking them in the **Navigation** bar or using the bar's previous and next arrows.

## Right Pane

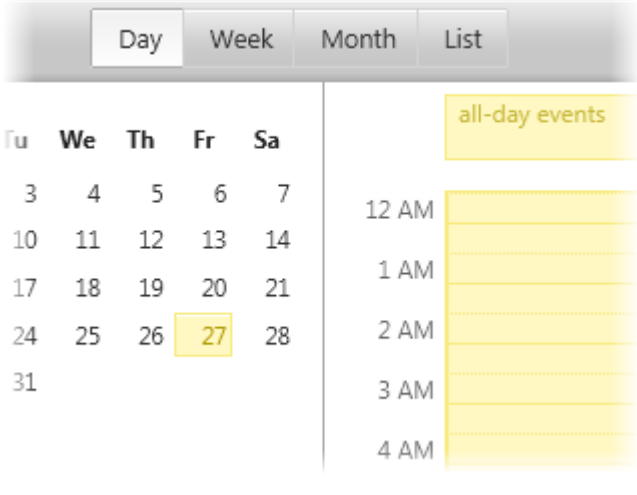
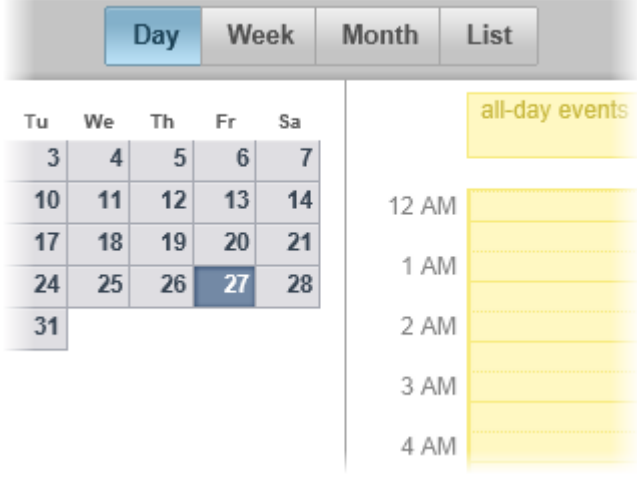
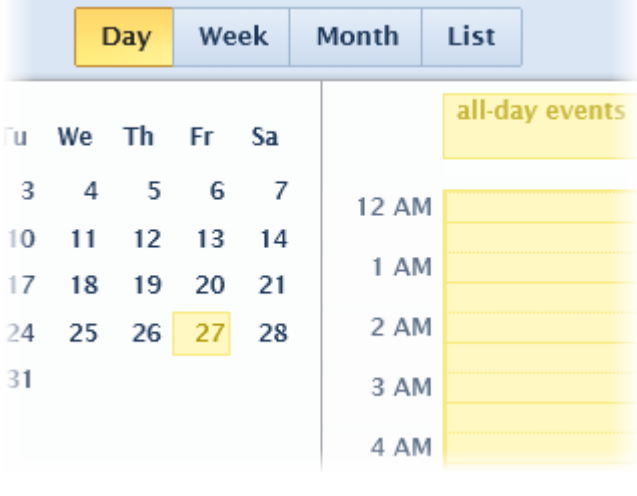
The **Right** pane is visible in the *Day* and *List* views and can be used to schedule events - just double-click the time for which you want to create an event.

## See Also

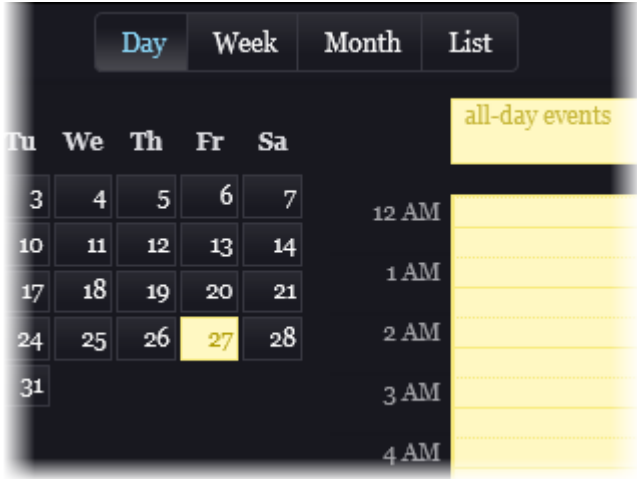
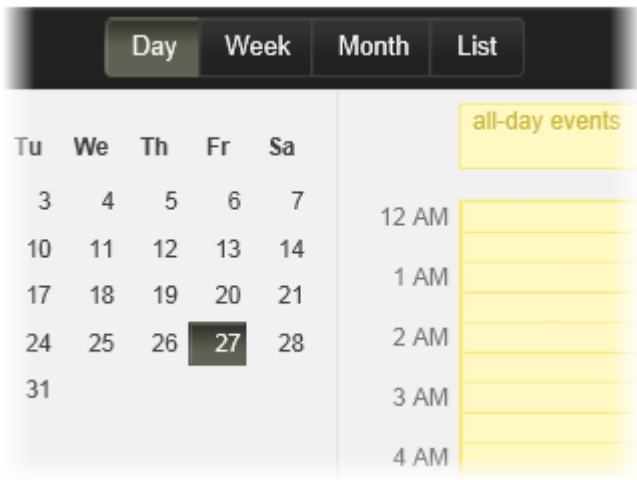
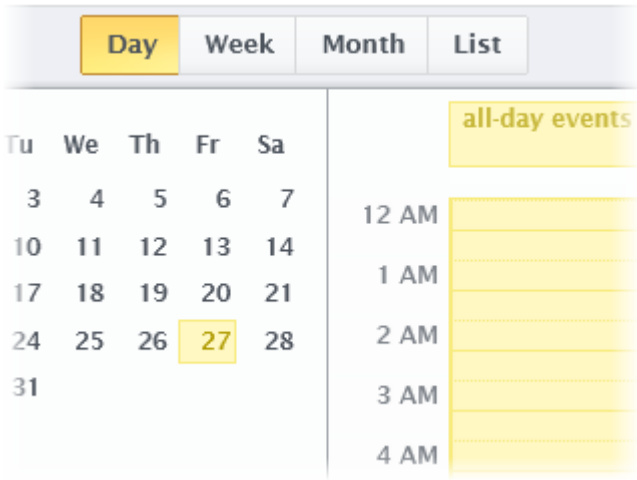
[Built-in Themes](#)

## Built-in Themes

The [C1EventsCalendar](#) control contains six built-in themes. When one of these themes is selected, all other **ASP.NET Web Forms Edition** controls on the page will be skinned accordingly. The themes will appear on the [C1EventsCalendar](#) control as follows:

<b>Arctic</b>	 <p>The Arctic theme features a light gray header with four buttons: 'Day' (selected), 'Week', 'Month', and 'List'. The calendar grid has a light gray background with white text for days of the week (Tu, We, Th, Fr, Sa) and dates. The date 27 is highlighted in yellow. The time slots on the right are labeled from 12 AM to 4 AM, with a yellow 'all-day events' bar at the top.</p>
<b>Aristo</b>	 <p>The Aristo theme features a light gray header with four buttons: 'Day' (selected), 'Week', 'Month', and 'List'. The calendar grid has a light gray background with white text for days of the week (Tu, We, Th, Fr, Sa) and dates. The date 27 is highlighted in blue. The time slots on the right are labeled from 12 AM to 4 AM, with a yellow 'all-day events' bar at the top.</p>
<b>Cobalt</b>	 <p>The Cobalt theme features a light blue header with four buttons: 'Day' (selected), 'Week', 'Month', and 'List'. The calendar grid has a light blue background with white text for days of the week (Tu, We, Th, Fr, Sa) and dates. The date 27 is highlighted in yellow. The time slots on the right are labeled from 12 AM to 4 AM, with a yellow 'all-day events' bar at the top.</p>



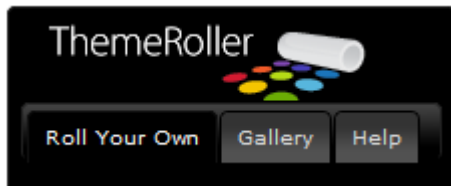
<b>Midnight</b>	 <p>The screenshot shows the C1EventsCalendar control in the 'Midnight' theme. The interface has a dark background. At the top, there are four buttons: 'Day' (highlighted in blue), 'Week', 'Month', and 'List'. Below these is a calendar grid with days of the week (Tu, We, Th, Fr, Sa) and dates (3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 17, 18, 19, 20, 21, 24, 25, 26, 27, 28, 31). The date 27 is highlighted in yellow. To the right of the grid is a vertical list of times from 12 AM to 4 AM. Above this list is a yellow box labeled 'all-day events'.</p>
<b>Rocket</b>	 <p>The screenshot shows the C1EventsCalendar control in the 'Rocket' theme. The interface has a light gray background. At the top, there are four buttons: 'Day' (highlighted in dark gray), 'Week', 'Month', and 'List'. Below these is a calendar grid with days of the week (Tu, We, Th, Fr, Sa) and dates (3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 17, 18, 19, 20, 21, 24, 25, 26, 27, 28, 31). The date 27 is highlighted in dark gray. To the right of the grid is a vertical list of times from 12 AM to 4 AM. Above this list is a yellow box labeled 'all-day events'.</p>
<b>Sterling</b>	 <p>The screenshot shows the C1EventsCalendar control in the 'Sterling' theme. The interface has a light gray background. At the top, there are four buttons: 'Day' (highlighted in yellow), 'Week', 'Month', and 'List'. Below these is a calendar grid with days of the week (Tu, We, Th, Fr, Sa) and dates (3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 17, 18, 19, 20, 21, 24, 25, 26, 27, 28, 31). The date 27 is highlighted in yellow. To the right of the grid is a vertical list of times from 12 AM to 4 AM. Above this list is a yellow box labeled 'all-day events'.</p>

To set the theme of the [C1EventsCalendar](#) control, simply set its [Theme](#) property to one of the built-in themes. This can be done in the C1EventsCalendar Tasks menu, in code, or in the Visual Studio Properties window. See [Using a Built-In Theme](#) for more information.

# ThemeRoller Overview

One of the most convenient features of Wijmo is its ThemeRoller compatibility. ThemeRoller is a Web application for creating unique jQueryUI themes to skin your Web app's widgets. With its simple interface and WYSIWYG preview, you can create a skin for all of your Wijmo widgets and other ThemeRoller-compatible widgets in less time than it would take to open a graphic editor.

We can find the ThemeRoller Web app at <http://jqueryui.com/themeroller/>. Once you've arrived at the app, take a gander at the column on the left-hand side of the page and observe that it holds three tabs: **Roll Your Own**, **Gallery**, and **Help**.



## The Roll Your Own Tab

**Roll Your Own** is where the magic happens. From this tab, we have the power to tweak your theme to perfection. To change an element, all we have to do is expand a node and get to work.

## The Gallery Tab

Clicking the **Gallery** contains a list and preview of ThemeRoller's premium themes. From here, we can preview, download, or edit one of the ready-made themes in the following ways:

- Clicking the snippet view of the theme loads an interactive preview to the right of the gallery.
- Clicking a theme's **Download** button will take us to the **Build Your Download** page. If we want to download one of the premade themes to skin your Wijmo widgets, all we have to do is navigate to the light orange panel, select our **Advanced Theme Settings** and **Version**, and click **Download**.
- Clicking **Edit** will load the chosen theme and return us to the **Roll Your Own** tab, where we can use the ThemeRoller to tweak the theme.

## The Help Tab

Clicking the **Help** tab will give us a quick reference and special information regarding the ThemeRoller, such as information for plugin developers and browser support notices.

See [Using a Custom Theme](#) for instructions on using the ThemeRoller to download a theme.

For more information on using the ThemeRoller and for a short tutorial, visit our Wijmo website.

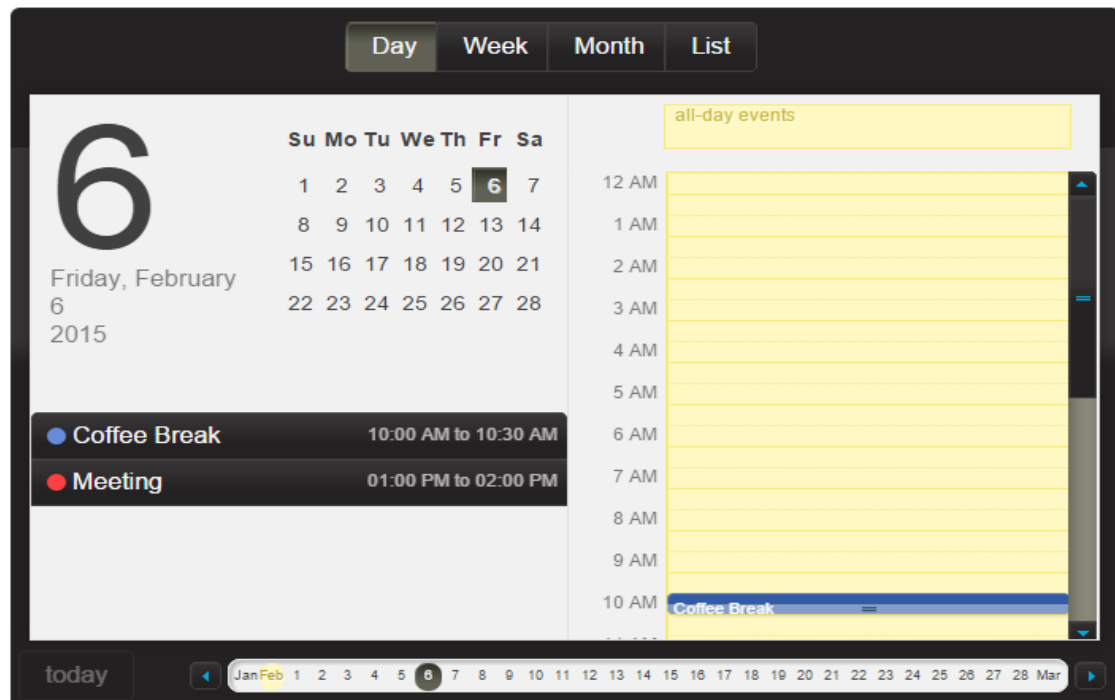
# Working with EventsCalendar for ASP.NET Web Forms

## Views

[C1EventsCalendar](#) includes several built-in data views. You can set the initial data view by using the Type property in the View Collection Editor. By default, it is set to **Day**. You can change its value to *Week*, *Month*, *List* and *Custom*.

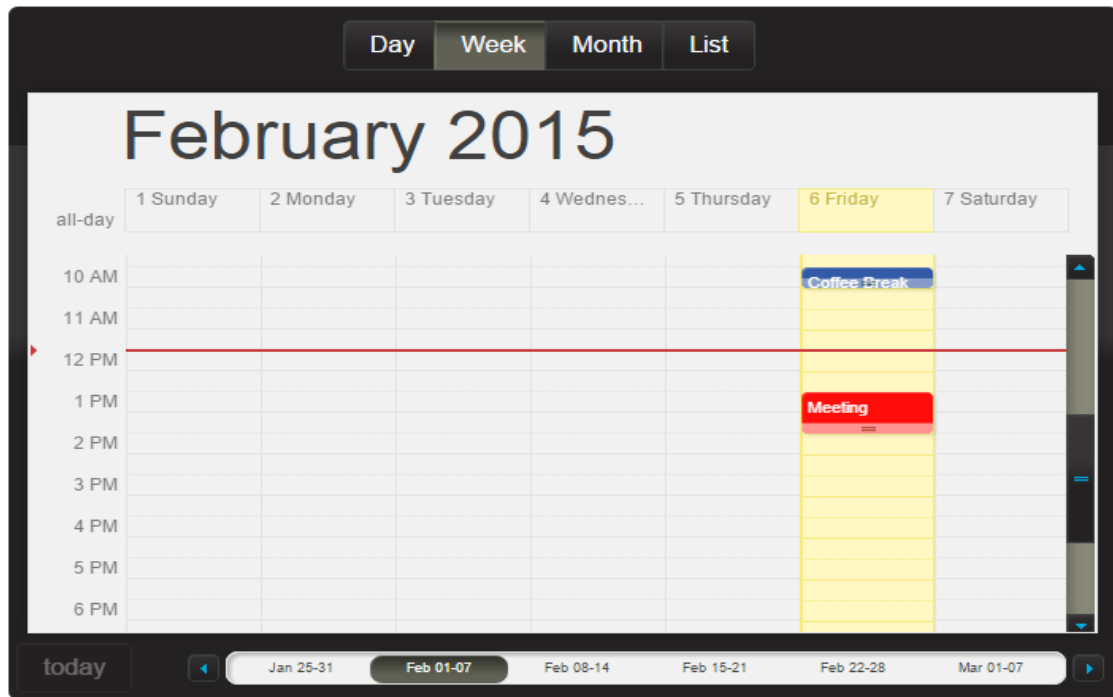
### Day

When the Type property is set to *Day*, the [C1EventsCalendar](#) control displays a small calendar for the month of the selected day and the selected day's schedule showing a 24-hour period starting from 12:00 AM. This is the default data view.



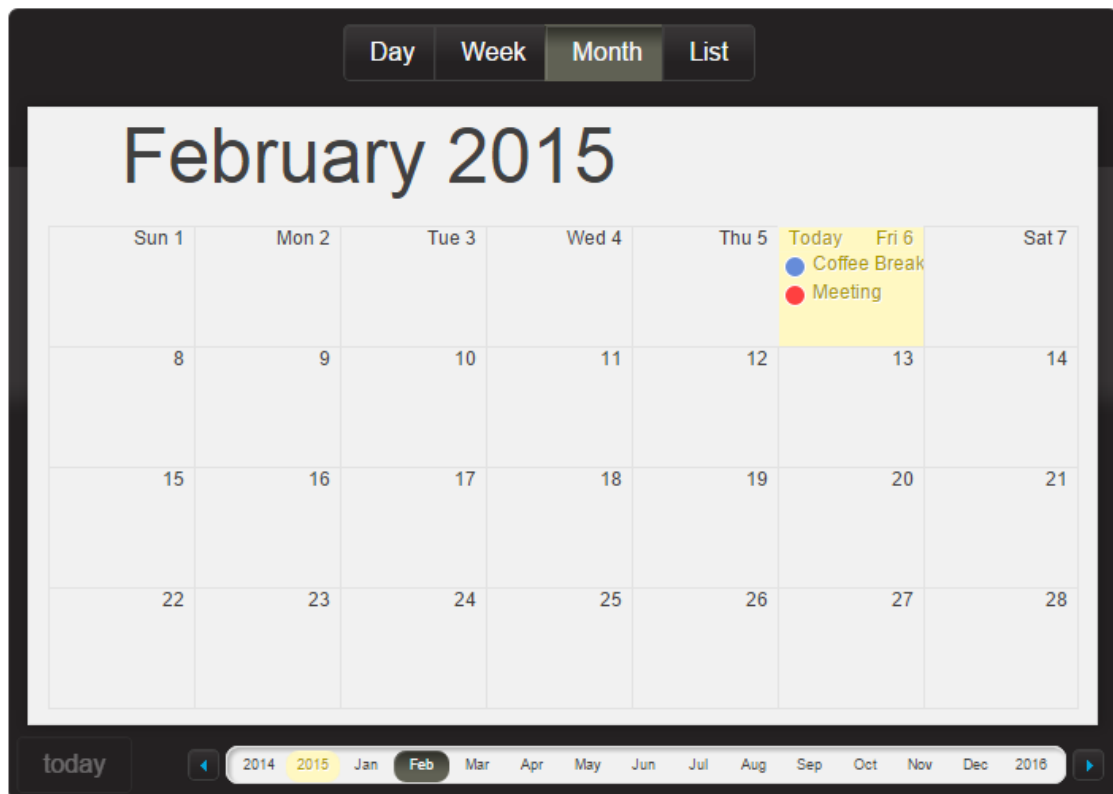
### Week

When the Type property is set to *Week*, the [C1EventsCalendar](#) control displays the calendar for the seven days of a week.



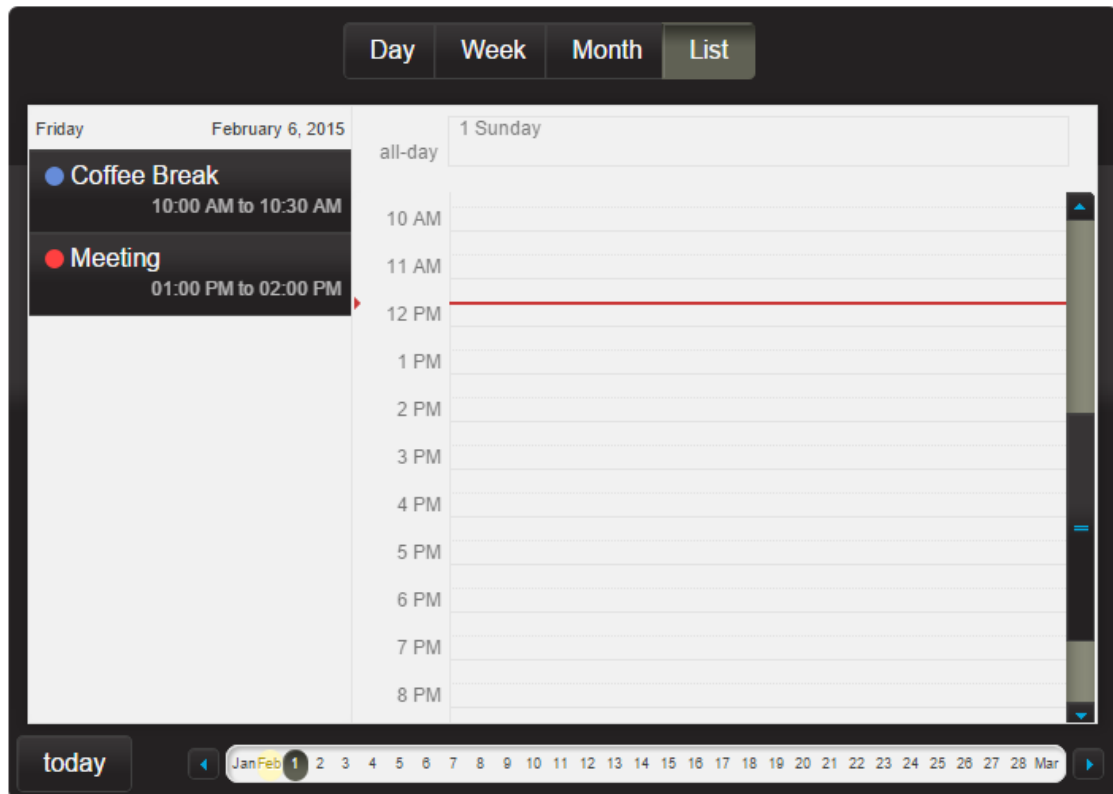
### Month

When the Type property is set to *Month*, the [C1EventsCalendar](#) control displays the calendar for the entire month, as well as days from the previous and following months, just as in a standard printed calendar. Note that in *Month* view mode, times are not displayed outside of events.



### List

When the Type property is set to *List*, the [C1EventsCalendar](#) control displays the schedule for the current or selected day and lists all scheduled events.



## See Also

[Changing the View](#)

## Changing the View

You can change the data view of a [C1EventsCalendar](#) control at design time using Source view, Properties window, and in Code.

Complete the following steps to set the [ViewType](#) property to **Month**:

### In Source View

Switch to Source view and add `ViewType="Month"` to the `<c1:C1EventsCalendar>` tag so that it appears similar to the following:

```
<c1:C1EventsCalendarID="C1EventsCalendar1" runat="server"
    ViewType="Month">
```

### In the Properties Window

1. Click the [C1EventsCalendar](#) control to select it.
2. Navigate to the Visual Studio Properties window and locate the View property.
3. Click the ellipsis button (...) next to **Views** property to open the View Collection Editor.
4. Go to member Months, set IsActive to **True**.

- Click the drop-down of **Type** property and select **Month**.

### In Code

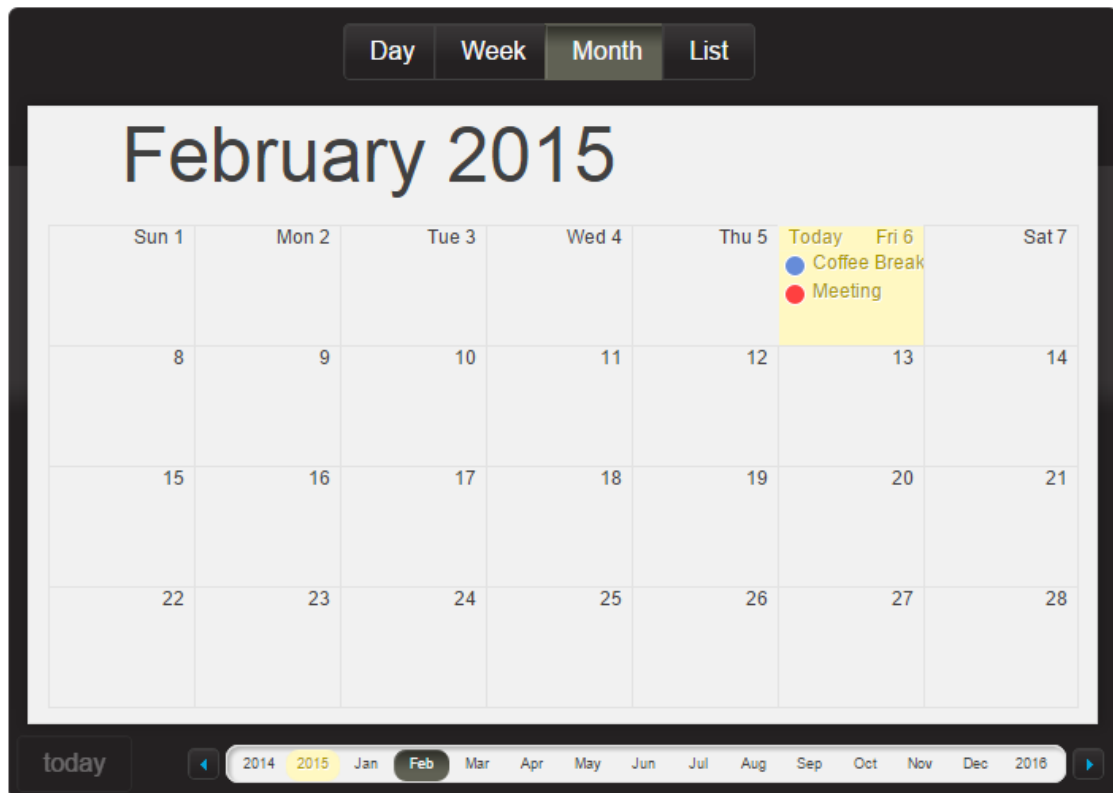
Add the following code to the **Page\_Load** event to set the calendar's **ViewType** to **Month**:

Visual Basic	Copy Code
<pre>C1EventsCalendar1.Views[2].IsActive = true</pre>	

C#	Copy Code
<pre>C1EventsCalendar1.Views[2].IsActive = true;</pre>	

### What You've Accomplished

When you run the project, EventsCalendar appears as shown in the image below.



## See Also

[Working with the Client-Side](#)

## Customizing the View

EventsCalendar shows [Day](#), [Week](#), [Month](#) and [List](#) views as default views. You can easily change the events calendar to show custom views by adding new views(such as two weeks or two months) having `ViewType` set to "Custom". You can customize the data view of the `C1EventsCalendar` control at design time using the Properties window, in Source view, and in Code.

Complete the following steps to add four custom views, **2 Days**, **2 Weeks**, **2 Months** and **2 Years**, and to set the active [ViewType](#) to 2 Months.

### In Source View

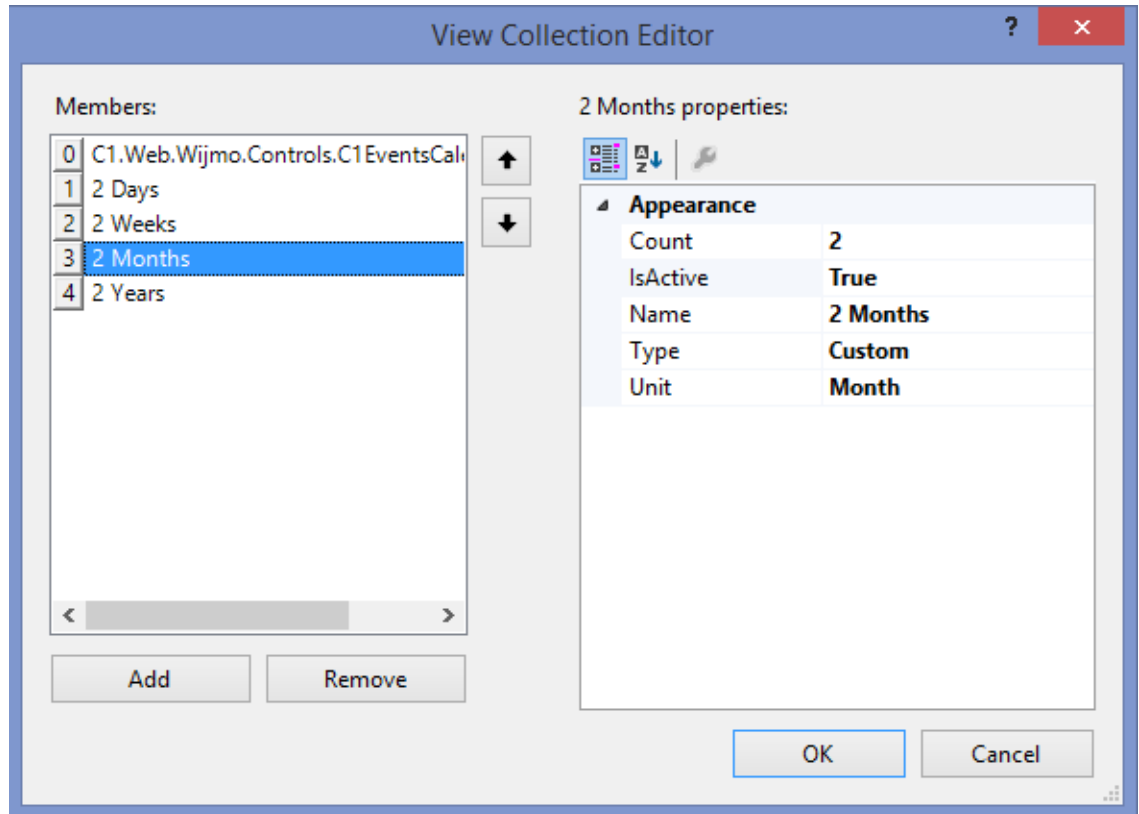
Switch to Source view and modify the `<cc1:C1EventsCalendar>` tags.

```
<cc1:C1EventsCalendar runat="server" ID="C1EventsCalendar1"
Width="60%" Height="475px">
  <Views>
    <cCl:View Type="Day">
    </cCl:View>
    <cCl:View Type="Custom" Name="2 Days" Unit="Day" Count="2">
    </cCl:View>
    <cCl:View Type="Custom" Name="2 Weeks" Unit="Week" Count="2">
    </cCl:View>
    <cCl:View Type="Custom" Name="2 Months" Unit="Month" Count="2"
IsActive ="true">
    </cCl:View>
    <cCl:View Type="Custom" Name="2 Years" Unit="Year" Count="2">
    </cCl:View>
  </Views>
</cc1:C1EventsCalendar>
```

### In the Properties Window

1. Right click the **C1EventsCalendar** control to open its Property window.
2. Navigate to the Visual Studio Properties window and locate the **Views** property.
3. Click the ellipsis button (...) next to **Views** property to open the **View Collection Editor**.
4. Click the **Add** button five times to add five new members in the header bar.
5. Set the **Type** of Member 1, 2, 3 and 4 to Custom. By default it is set to **Day**.
6. Set the Name property for each member you added as follows:
  - o Member 1: **2 Days**
  - o Member 2: **2 Weeks**
  - o Member 3: **2 Months**
  - o Member 4: **2 Years**
7. Set the **Count** property to **2** for all four custom members. By default it is set to 1.

8. Set **IsActive** to **True** for member 2 Months.



9. Click **OK** and press **F5** to run the project.

### In Code

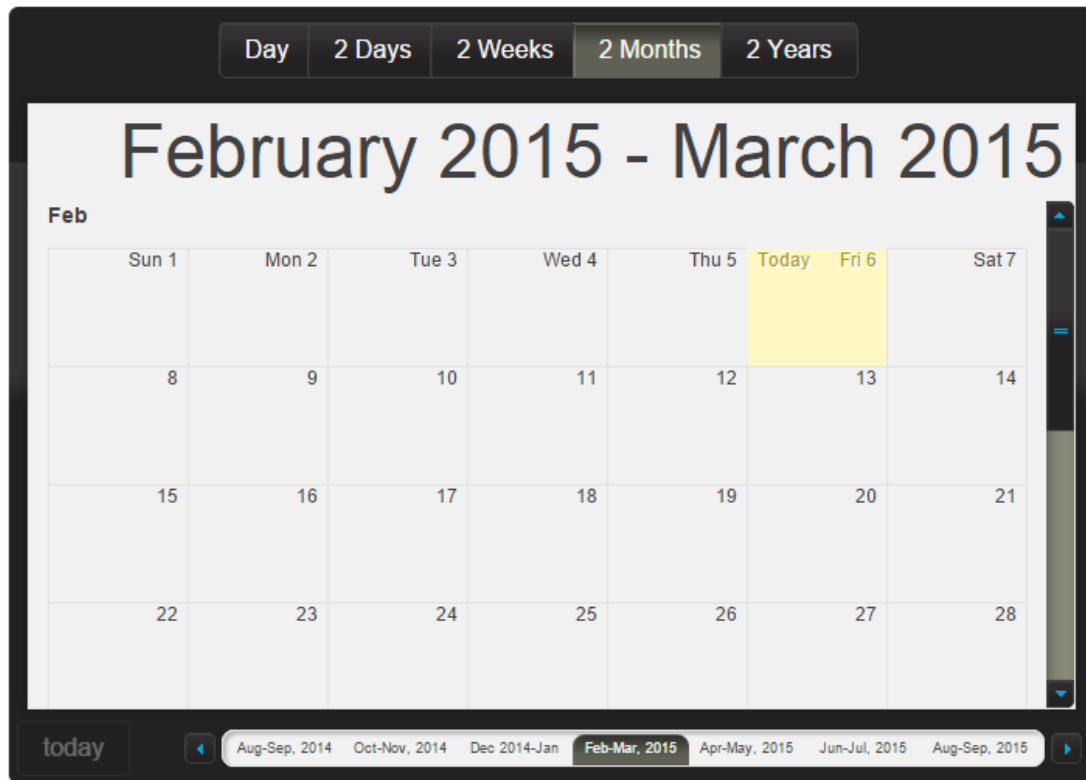
Visual Basic	Copy Code
<pre>C1EventsCalendar1.Views[2].IsActive = true</pre>	

C#	Copy Code
<pre>C1EventsCalendar1.Views[2].IsActive = true;</pre>	

### What You've Accomplished

When you run the project, EventsCalendar appears as shown in the image below.





## Data and DataBinding

**EventsCalendar for ASP.NET Web Forms** allows you to use standard ADO.NET databinding to bind the [C1EventsCalendar](#) control to a datasource.

Binding data to the [C1EventsCalendar](#) control involves two simple steps:

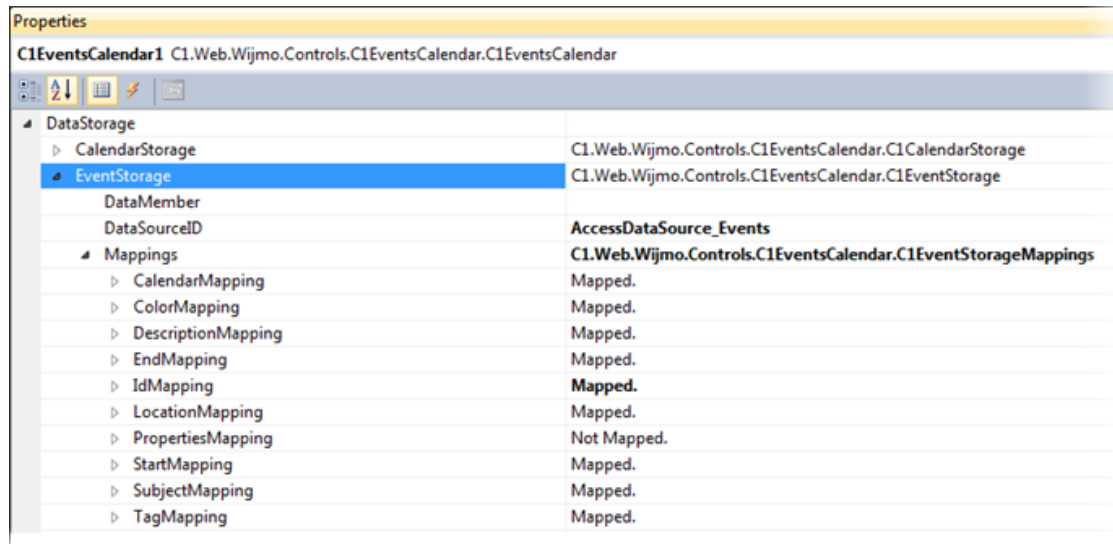
1. **Configure an AccessDataSource:** Choose the database to bind to and then specify the parameters that are mapped to the fields in the database table.
2. **Map to the Data Storage:** Map to the [C1EventsCalendar](#) data storage ([C1EventStorage](#) or [C1CalendarStorage](#)) by specifying the data source in the [C1EventStorage.DataSourceID](#)/[C1CalendarStorage.DataSourceID](#) property and then entering the names of the data fields using the **Mappings** properties. Each database table and its columns can be mapped to separate areas in the storage, and then calendars, subjects, colors, descriptions, start and end times, locations and more can be loaded from the database.

### C1EventStorage

The [C1EventStorage](#) class allows you to begin binding to a data source by specifying the data source in the [DataSourceID](#) property. You can then map the fields from the data source to the EventStorage using the [C1EventStorageMappings](#) properties.

C1EventStorageMappings Properties	Description
-----------------------------------	-------------

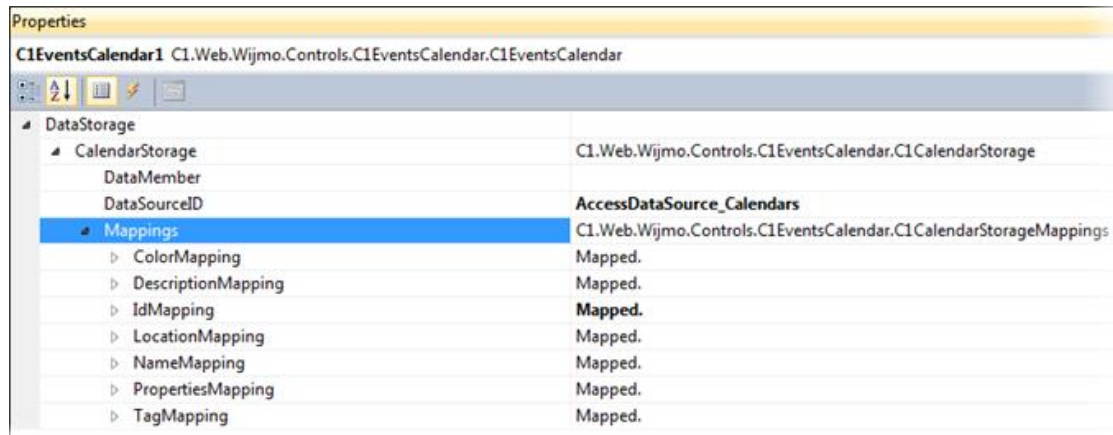
<a href="#">CalendarMapping</a>	Allows the calendar property to be bound to the appropriate field in the data source.
<a href="#">ColorMapping</a>	Allows the color property to be bound to the appropriate field in the data source.
<a href="#">DescriptionMapping</a>	Allows the description property to be bound to the appropriate field in the data source.
<a href="#">EndMapping</a>	Allows the end property to be bound to the appropriate field in the data source.
<a href="#">IdMapping</a>	Gets the Id property to be bound to the appropriate field in the data source.
<a href="#">LocationMapping</a>	Allows the location property to be bound to the appropriate field in the data source.
<a href="#">PropertiesMapping</a>	Allows other event properties to be bound to the appropriate field in the data source.
<a href="#">StartMapping</a>	Allows the start property to be bound to the appropriate field in the data source.
<a href="#">SubjectMapping</a>	Allows the subject property to be bound to the appropriate field in the data source.
<a href="#">TagMapping</a>	Allows the tag property to be bound to the appropriate field in the data source (the tag property can be used to store any user-defined information).



### C1CalendarStorage

The [C1CalendarStorage](#) class allows you to begin binding to a data source simply by specifying the data source in the [DataSourceID](#) property. You can then map the fields from the data source to the CalendarStorage using the [C1EventStorageMappings](#) properties.

C1CalendarStorageMappings Properties	Description
<a href="#">ColorMapping</a>	Allows the color property to be bound to the appropriate field in the data source.
<a href="#">DescriptionMapping</a>	Allows the description property to be bound to the appropriate field in the data source.
<a href="#">LocationMapping</a>	Allows the location property to be bound to the appropriate field in the data source.
<a href="#">NameMapping</a>	Allows the name property to be bound to the appropriate field in the data source.
<a href="#">PropertiesMapping</a>	Allows other calendar properties to be bound to the appropriate field in the data source.
<a href="#">TagMapping</a>	Allows the tag property to be bound to the appropriate field in the data source (the tag property can be used to store any user-defined information).



The [Binding Data to a C1EventsCalendar Control Tutorial](#) provides step-by-step instructions on configuring two **AccessDataSource** controls and mapping to the [C1EventStorage](#) and [C1CalendarStorage](#).

## See Also

[Binding Data to a C1EventsCalendar Control Tutorial](#)

# Binding Data to a C1EventsCalendar Control Tutorial

This tutorial will walk you through the steps of binding a [C1EventsCalendar](#) control to the **EventsCalendarNwind.mdb** database, which will be used to populate the control with some sample events and calendars. You will need to configure two **AccessDataSource** controls; one will map to the [C1EventStorage](#) and the other will map to the [C1CalendarStorage](#).

Before you begin, copy the database to the **App\_Data** folder of your project folder.

Start by creating a new Web application or Web site application and adding the **EventsCalendarNwind.mdb** database to the project. Then add a [C1EventsCalendar](#) and two **AccessDataSource** controls to the page. To do this, follow these steps:

1. Create a new Web application.
2. In the Solution Explorer, right-click the **App Data** folder and select **Add | Existing Item**.
3. In the **Add Existing Item** dialog box, select **EventsCalendarNwind.mdb** and click **Add**.
4. Add a [C1EventsCalendar](#) control and two Microsoft **AccessDataSource** data controls to your form. Change the ID for the first **AccessDataSource** control to **AccessDataSource\_Events**, and change the ID for the second **AccessDataSource** control to **AccessDataSource\_Calendars**.

## See Also

[Configuring the AccessDataSource Control](#)

[Mapping to the Data Storage](#)

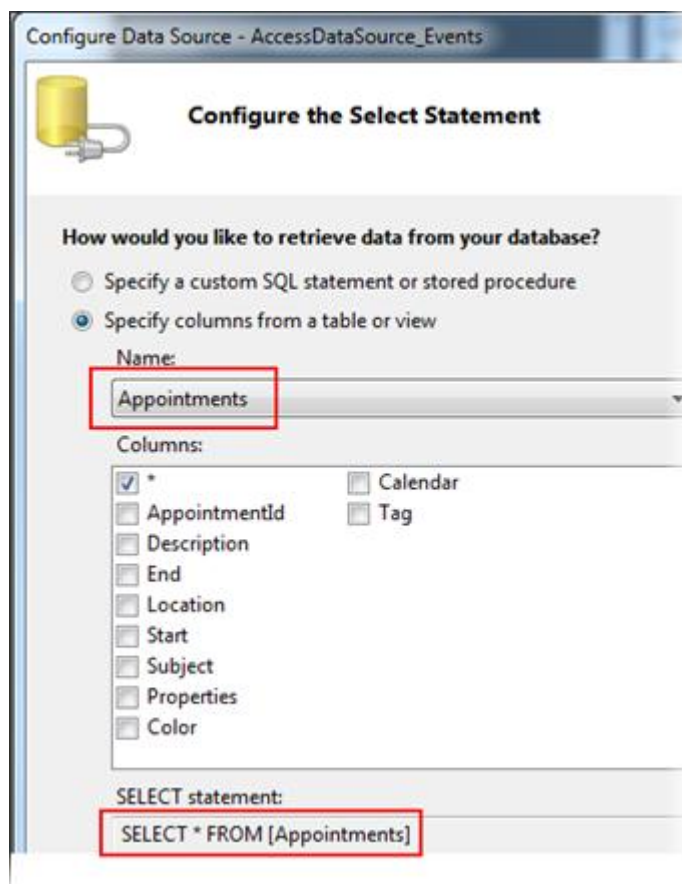
## Configuring the AccessDataSource Control

In this example, you have two **AccessDataSource** controls to configure: one for the *Appointments* table of the database (we'll call it **AccessDataSource\_Events**) and one for the *Calendars* table of the database (we'll call it **AccessDataSource\_Calendars**).

### Configuring the First AccessDataSource Control

Set up the first **AccessDataSource** control, which will be used to map to the fields in the *Appointments* table of the database. This table contains data for the start and end times of an appointment, the subject, where an appointment is an all day event, the Appointment ID, the color it will be, and the calendar to associate with it.

1. In Design view, select **AccessDataSource\_Events**, click the smart tag, and select **Configure Data Source**.
2. Select **EventsCalendarNwind.mdb** in the **App\_Data** folder and click **OK**.
3. Click **Next**.
4. Select **Appointments** under *Specify columns from a table or view*. The SELECT statement becomes **SELECT \* FROM [Appointments]**.



5. Click **Next** and then click **Finish** to close the dialog box.

### DeleteQuery Property

1. In Design view, select **AccessDataSource\_Events**, and in the Visual Studio Properties window, click the **ellipsis** button next to the **DeleteQuery** property.
2. In the **Command and Parameter Editor**, add the following DELETE command:

```
DELETE FROM [Appointments] WHERE [AppointmentId] = ?
```

3. Click the **Add Parameter** button and add a parameter named **AppointmentId**. Click the **Show advanced properties** link and select **Object** next to the **Type** property.
4. Click **OK** to close the dialog box.

### InsertQuery Property

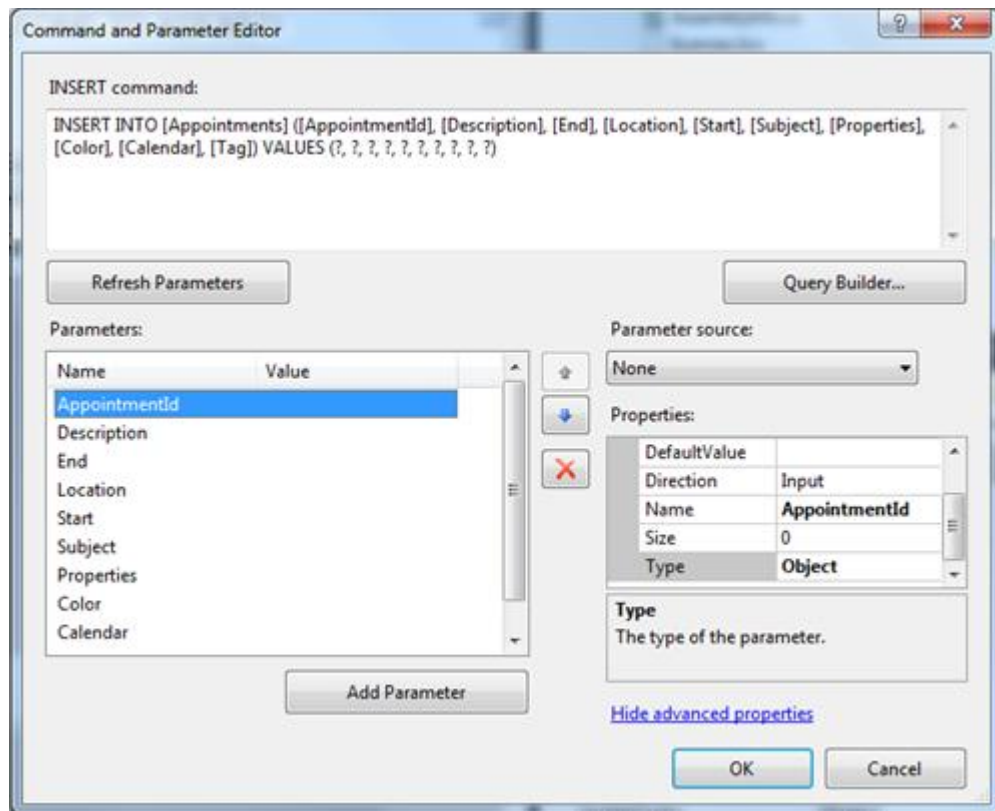
1. In Design view, select **AccessDataSource\_Events**, and in the Visual Studio Properties window, click the **ellipsis** button next to the **InsertQuery** property.
2. In the **Command and Parameter Editor**, add the following INSERT command:

```
INSERT INTO [Appointments] ([AppointmentId], [Description], [End], [Location], [Start],  
[Subject], [Properties], [Color], [Calendar], [Tag]) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)
```

3. Click the **Add Parameter** button to add the following parameters, specifying their **Names** and **Types**. Note that you must select the **Show advanced properties** link to view and specify the **Type** property.

Parameter name	Type
AppointmentId	Object
Description	String
End	DateTime
Location	String
Start	DateTime
Subject	String
Properties	String
Color	String
Calendar	String

Tag	String
-----	--------



- 4.
5. Click **OK** to close the dialog box.

### UpdateQuery Property

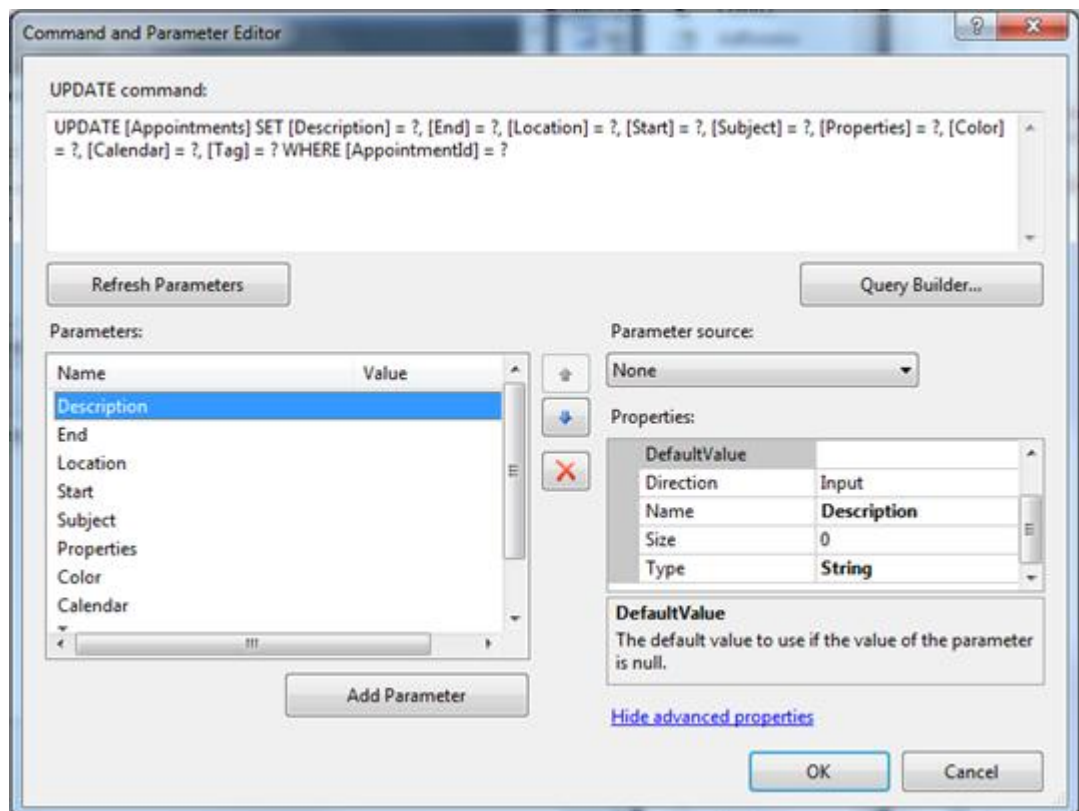
1. In Design view, select **AccessDataSource\_Events**, and in the Visual Studio Properties window, click the **ellipsis** button next to the **UpdateQuery** property.
2. In the **Command and Parameter Editor**, add the following UPDATE command:

```
UPDATE [Appointments] SET [Description] = ?, [End] = ?, [Location] = ?, [Start] = ?,
[Subject] = ?, [Properties] = ?, [Color] = ?, [Calendar] = ?, [Tag] = ? WHERE
[AppointmentId] = ?
```

3. Click the **Add Parameter** button to add the following parameters, specifying their **Names** and **Types**. Note that you must select the **Show advanced properties** link to view and specify the **Type** property.

Parameter name	Type
String	String
End	DateTime

Location	String
Start	DateTime
Subject	String
Properties	String
Color	String
Calendar	String
Tag	String
AppointmentId	Object



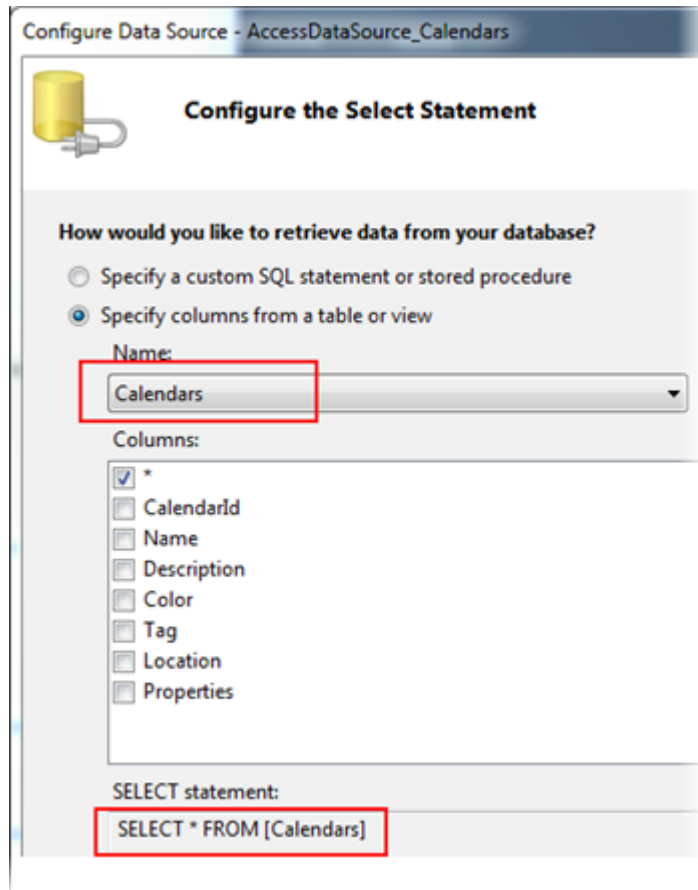
- 4.
5. Click **OK** to close the dialog box.

### Configuring the Second AccessDataSource Control

Set up the second **AccessDataSource** control, which will be used to map to the fields in the *Calendars* table of the database. This table contains data for the name, description, location, color and Calendar ID for two different calendars.



1. In Design view, select **AccessDataSource\_Calendars**, click the smart tag, and select **Configure Data Source**.
2. Select **EventsCalendarNwind.mdb** in the **App\_Data** folder and click **OK**.
3. Click **Next**.
4. Select **Calendars** under *Specify columns from a table or view*. The SELECT statement becomes **SELECT \* FROM [Calendars]**.



5. Click **Next** and then click **Finish** to close the dialog box.

### DeleteQuery Property

1. In Design view, select **AccessDataSource\_Calendars**, and in the Visual Studio Properties window, click the **ellipsis** button next to the **DeleteQuery** property.
2. In the **Command and Parameter Editor**, add the following DELETE command:

```
DELETE FROM [Calendars] WHERE [CalendarId] = ?
```

3. Click the **Add Parameter** button and add a parameter named **CalendarId**. Click the **Show advanced properties** link and select **String** next to the **Type** property.
4. Click **OK** to close the dialog box.

### InsertQuery Property

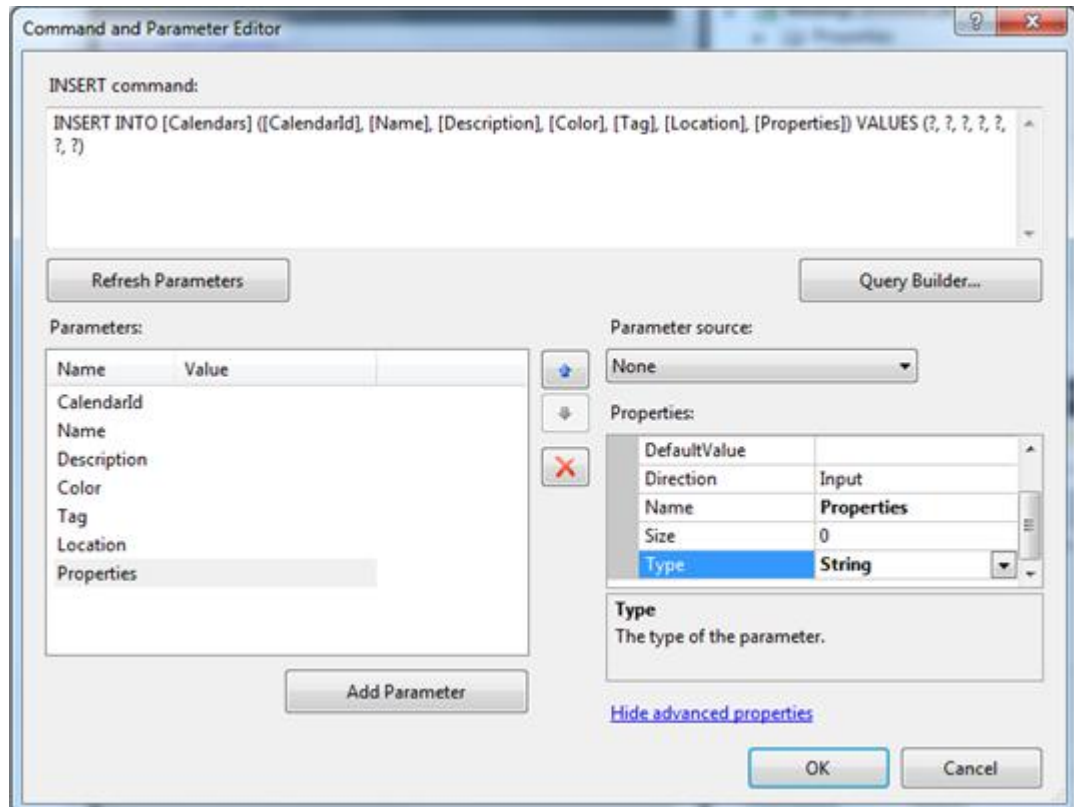
1. In Design view, select **AccessDataSource\_Calendars**, and in the Visual Studio Properties window, click the **ellipsis** button next to the **InsertQuery** property.

2. In the **Command and Parameter Editor**, add the following INSERT command:

```
INSERT INTO [Calendars] ([CalendarId], [Name], [Description], [Color], [Tag], [Location],  
[Properties]) VALUES (?, ?, ?, ?, ?, ?, ?)
```

3. Click the **Add Parameter** button to add the following parameters, specifying their **Names** and **Types**. Note that you must select the **Show advanced properties** link to view and specify the **Type** property.

Parameter name	Type
CalendarId	String
Name	String
Description	String
Color	String
Tag	String
Location	String
Properties	String



- 4.
5. Click **OK** to close the dialog box.

### UpdateQuery Property

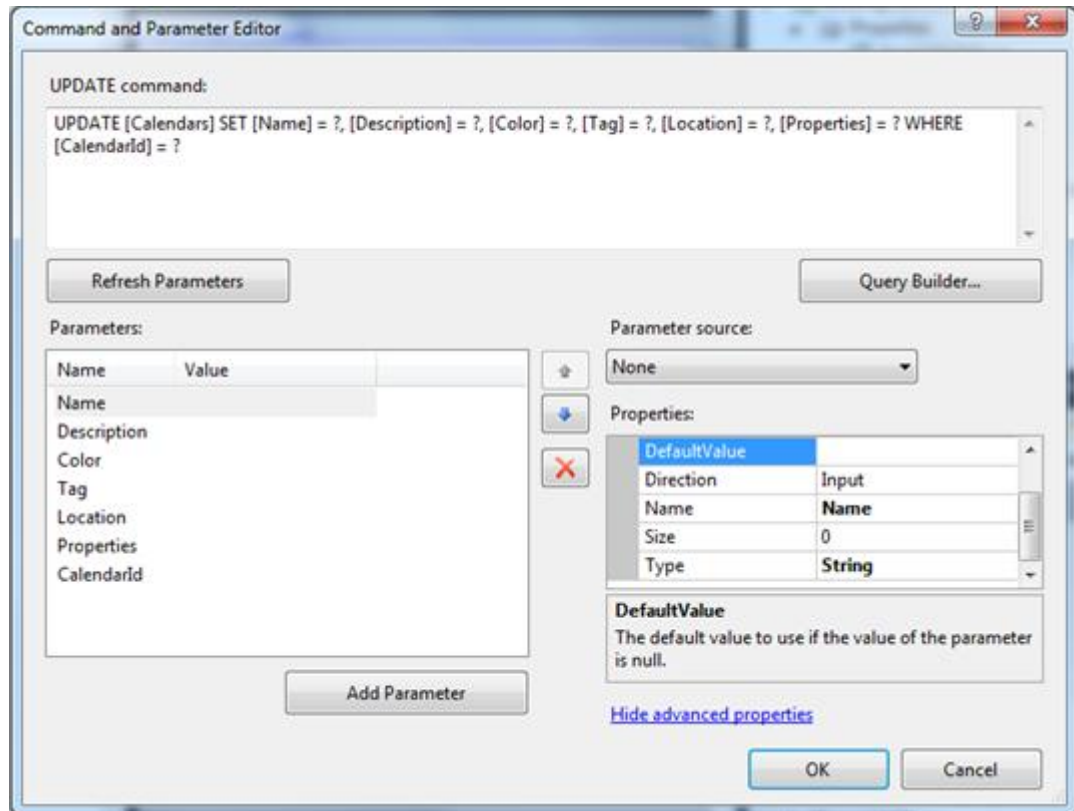
1. Select **AccessDataSource\_Events** and in the Visual Studio Properties window, click the **ellipsis** button next to the **UpdateQuery** property.
2. In the **Command and Parameter Editor**, add the following UPDATE command:

```
UPDATE [Calendars] SET [Name] = ?, [Description] = ?, [Color] = ?, [Tag] = ?, [Location] = ?, [Properties] = ? WHERE [CalendarId] = ?
```

3. Click the **Add Parameter** button to add the following parameters, specifying their **Names** and **Types**. Note that you must select the **Show advanced properties** link to view and specify the **Type** property.

Parameter Name	Type
Name	String
Description	String
Color	String
Tag	String

Location	String
Properties	String
CalendarId	String



- 4.
5. Click **OK** to close the dialog box.

## See Also

[Mapping to the Data Storage](#)

## Mapping to the Data Storage

Once the **AccessDataSource** controls are mapped to the database, you can begin mapping to the **C1EventsCalendar** data storage. This must be done in order for the events and calendars in the data source to appear in the events calendar in your application. Since there are both events and calendars stored in the **EventsCalendarNwind.mdb** and you configured the **AccessDataSource** controls to access these tables of the database, you can map to the **C1EventsCalendar** **C1EventStorage** and **C1CalendarStorage**.

To map to the **C1EventStorage**, follow these steps:

1. Select the **C1EventsCalendar** control and click **View | Properties Window**.
2. Expand the **DataStorage** node and then expand the **EventStorage** node.

3. Enter **AccessDataSource\_Events** next to the **DataSource** property.
4. Expand the **Mappings** node and then expand each node under this node. Enter the mapping properties as defined in the following table:

Mapping Node	MappingName Property
CalendarMapping	Calendar
ColorMapping	Color
DescriptionMapping	Description
EndMapping	End
IdMapping	AppointmentId
LocationMapping	Location
PropertiesMapping	N/A
StartMapping	Start
SubjectMapping	Subject
TagMapping	Tag

5. To map to the [C1CalendarStorage](#), follow these steps:
  1. Select the [C1EventsCalendar](#) control and click **View | Properties Window**.
  2. Expand the **DataStorage** node and then expand the **CalendarStorage** node.
  3. Enter **AccessDataSource\_Calendars** next to the **DataSource** property.
  4. Expand the **Mappings** node and then expand each node under this node. Enter the mapping properties as defined in the following table:

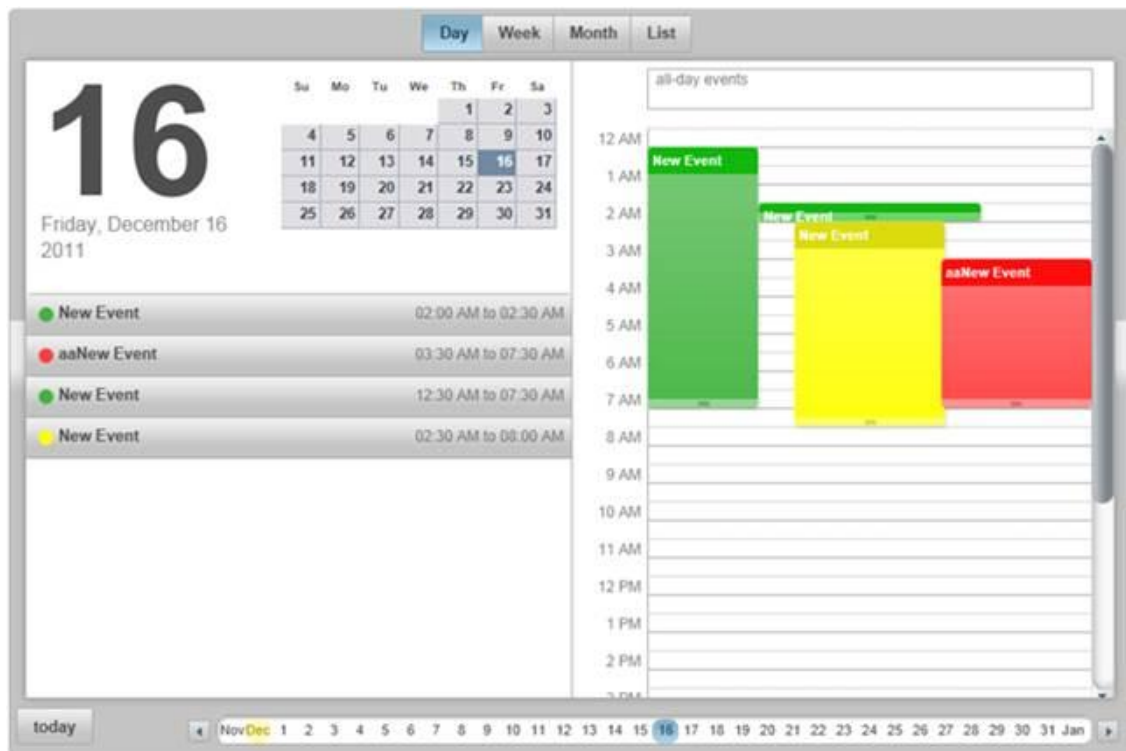
Mapping Node	MappingName Property
ColorMapping	Color
DescriptionMapping	Description
IdMapping	CalendarId

LocationMapping	Location
NameMapping	Name
PropertiesMapping	Properties
TagMapping	Tag

### Run the Project

Press **F5** to run the application.

Find **December 16, 2011** using the date bar at the bottom of the [C1EventsCalendar](#) in Day view.



Click the yellow event. Notice that the calendar property in the **Events** dialog box is mapped to **adeline's calendar**. The red event is mapped to **john's calendar**.

## See Also

[Events](#)

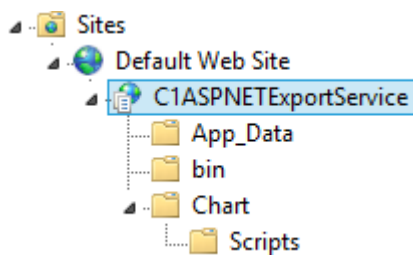
# Export Service

Use the ComponentOne ASP.NET Export Service, to export EventsCalendar to Excel, CSV and PDF, without creating a complex export application. This service resides on the application server.

The advantages of using ComponentOne ASP.NET Export Service are:

- Export your EventsCalendar, while preserving its formatting.
- Change exported file settings as per the requirement.

Export Service is a web-application that is deployed on Internet Information Services (IIS). Run the **C1ASPNETExportService** installer, placed in C:\Program Files\ComponentOne\ASP.NET Web Forms Edition folder. It installs the following files in IIS.



You can also locate these files at the following location:

C:\ProgramData\ComponentOne\C1ASPNET\C1ASPNETExportService

## System Requirements

Following are the system requirements of the service host:

- Microsoft Windows 7 or above.
- IIS 7.0 or above with ASP.NET 4.0 or above ( .NET framework 4.0 ).



You may update .Net framework 4.0 in **Microsoft Windows 7** or **Microsoft Windows Server 2008 R2**. See <http://support.microsoft.com/kb/2468871>, for more information.

## EventsCalendar Export Setting

Following are the settings, to download the eventscalendar in various formats:

### Export to Excel

- **FileFormat:** EventsCalendar can be exported to Excel sheets in xls or xlsx format.
- **Author:** Specifies the name of the person or organization responsible for writing the data.
- **Auto Row Height:** Selects whether the rows height will be adjusted based on its content or not.

- **Server URL:** Sets the URL of the server, from where the eventscalendar is to be exported. Enter ServerURL/exportapi/eventscalendar.
- **File Name:** Sets the name to be used for the exported file.

#### Export to CSV:

- **Server URL:** Sets the URL or the server, from where the eventscalendar is to be exported. Enter ServerURL/exportapi/eventscalendar.
- **File Name:** Sets the name to be used for the exported file.

#### Export to PDF

- **Repeat EventsCalendar Header:** Repeats the EventsCalendar header.
- **Auto Fit Width:** Enables auto-fit.
- **Landscape:** Enables landscape mode.
- **Margin Settings:**
  - - Top: Sets the top margin size in points.
    - Bottom: Sets the bottom margin size in points.
    - Right: Sets the right margin size in points.
    - Left: Sets the left margin size in points.
    - Paper Kind: Sets the paper kind. eg: Custom, Letter, LetterSmall, etc
- **Page Size:**
  - - Width: Sets the page width in points
    - Height: Sets the page height in points.
- **File Content:**
  - - Image Quality: Sets the image quality to Low, Medium or High
    - Compression: Sets the compression level to Default, None, Best Speed or Best Compression.
    - Font Type: Sets the Font type to True Type or Embedded.
- **Document Info:**
  - - Author: Sets the name of the person or organization that created the document.
    - Creator: Sets the name of the application that created the original document.
    - Subject: Sets the subject of the document.
    - Title: Sets document title in the title bar.
    - Producer: Sets the name of the application that created the PDF document.
    - Keywords: Sets the keywords associated with the PDF document that can be used to locate the document.
- **Document Security:**
  - - Encryption Type: Sets the Encryption Type to NotPermit, Standard40, Standard128 or Aes128.
    - Owner Password: Sets the password required to edit permissions for the document.
    - User Password: Sets the password required to open the document.
    - Allow Copy Content: Enables or disables copy content.
    - Allow Edit Annotations: Enables or disables users from editing annotations.



- Allow Edit Content: Enables or disables users from editing content in the document.
- Allow Print: Enables or disables printing for the document.
- **Configuration Setting:**
  - 
  - Server URL: Sets the server URL. Enter ServerURL/**exportapi/eventscalender**.
  - File Name: Sets the file name to be used for the exported PDF.

## Usage

The `exportEventsCalendar` method will be called to export the EventsCalendar content to Excel, CSV or PDF. These steps assume that you have added a button to the form, on whose click event you would call the export function. Add the following code within the `<head></head>` tags, to export the EventsCalendar to Excel.

```
<script src="http://code.jquery.com/jquery-1.9.1.min.js"
type="text/javascript"></script>
<asp:PlaceHolder runat="server">
    <script type="text/javascript">

        $(function () {
            $("#Button1").click(exportExcel);
        });

        // Export function
        function exportExcel() {
            var fileName = "ExportedEventsCalendar";
            var type = "Xls";
            var excelSetting = {
                autoRowHeight: true,
                author: "ComponentOne"
            };
            var url =
"http://demos.componentone.com/ASPNET/ExportService" +
"/exportapi/eventscalender";

$("#<%=C1EventsCalendar1.ClientID%>").c1eventscalender("exportEventsCalendar"
, fileName, type, excelSetting, url);
        }
    </script>
</asp:PlaceHolder>
```

Add the following code within the `<head></head>` tags, to export the EventsCalendar to a CSV file:

```
<script src="http://code.jquery.com/jquery-1.9.1.min.js"
type="text/javascript"></script>
<asp:PlaceHolder ID="PlaceHolder1" runat="server">
    <script type="text/javascript">

        $(function () {
            $("#Button1").click(exportCsv);
        });

        // Export function
```

```
function exportCsv() {
    var fileName = "ExportedEventsCalendar";
    var url =
"http://demos.componentone.com/ASPNET/ExportService" +
"/exportapi/eventscalender";

$("#<%=C1EventsCalendar1.ClientID%>").c1eventscalender("exportEventsCalendar"
, fileName, "csv", url);
}
</script>
</asp:Placeholder>
```

Add the following code within the <head></head> tags, to export the EventsCalendar to a PDF:

```
<script src="http://code.jquery.com/jquery-1.9.1.min.js"
type="text/javascript"></script>
<asp:Placeholder ID="Placeholder1" runat="server">
    <script type="text/javascript">

        $(function () {
            $("#Button1").click(exportPdf);

        });

        // pdf settings
        function getPdfSetting() {
            return {
                repeatHeader: true,
                landscape: true,
                autoFitWidth: true,
                pageSize: {
                    width: 300,
                    height: 400
                },
                paperKind: 'A4',
                margins: {
                    top: 50,
                    right: 50,
                    bottom: 50,
                    left: 50
                },
                imageQuality: 'Low',
                compression: 'BestCompression',
                fontType: 'TrueType',
                author: 'ComponentOne',
                creator: 'ComponentOne',
                subject: 'EventsCalendar Export',
                title: 'EventsCalendar Export',
                producer: 'ComponentOne',
                keywords: 'EventsCalendar, EventsCalendar,
Export, PDF',

                encryption: 'NotPermit',
                ownerPassword: '0000',
                userPassword: '00000',
                allowCopyContent: true,
                allowEditAnnotations: true,
                allowEditContent: true,
                allowPrint: true,
```

```

    }
}

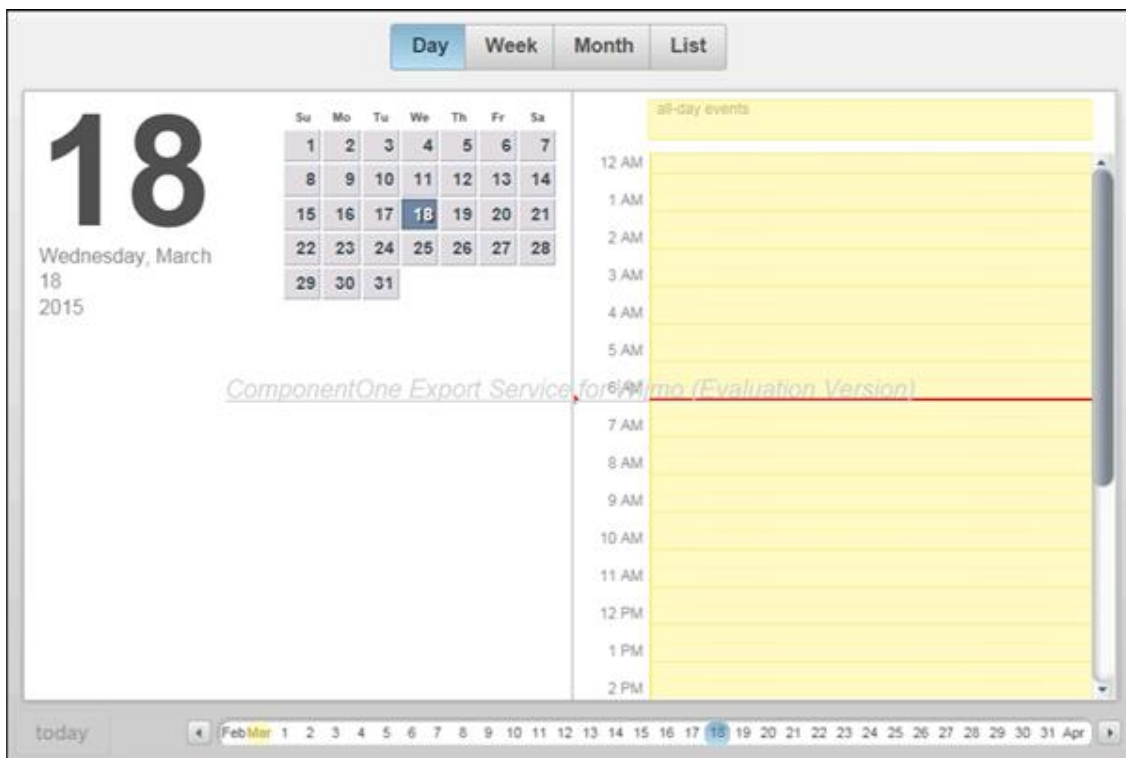
// Export function
function exportPdf() {
    var fileName = 'ExportEventsCalendar';
    var pdfSetting = getPdfSetting();
    var url =
"http://demos.componentone.com/ASPNET/ExportService" +
"/exportapi/eventscalendar";

$("#<%=C1EventsCalendar1.ClientID%>").c1eventscalendar("exportEventsCalendar"
, fileName, "pdf", pdfSetting, url);
}

</script>
</asp:Placeholder>

```

The following image displays the PDF generated:



In case the file is not downloaded on Internet Explorer, turn off Internet Explorer protected mode to export file or run Internet Explorer as administrator. To turn off the protected mode:

- Open Internet Explorer **Settings** and select **Internet Options**.
- In the **Security** tab, select **Internet** and uncheck "**Enable Protected Mode**".

## Events

An event represents a period of time and detailed information about something that will take place during that period of time.

Events can be added at run time by selecting the date on the calendar and then double-clicking the time slot (for *Day*, *Week*, or *List* view). If you are in *Month* view, double-click the date of the event, and check the **all-day** checkbox or enter a start and end time.

Once an event is added to the calendar, it can be removed, if necessary. Double-click the event and then click the **Delete** button.

## See Also

[Views](#)

# Working with the Client-Side

The **EventsCalendar for ASP.NET Web Forms** controls have a very rich client-side object model since most of their members are identical to the members in the server-side control.

When a **C1EventsCalendar** control is rendered, an instance of the client-side control will be created automatically. This means that you can enjoy the convenience of accessing properties and methods of the **C1EventsCalendar** control without having to postback to the server.

Using client-side code, you can implement many features in your Web page without the need to send information to the Web server, which takes time. Thus, using the client-side object model can increase the efficiency of your Web site.

## See Also

[Client-Side Events](#)

# Client-Side Events

**EventsCalendar for ASP.NET Web Forms** includes several client-side events that allow you to manipulate the **C1EventsCalendar** control when an action such as an invalid character is entered.

You can use the sever-side properties, listed in the Client Side Event table, to specify the name of the JavaScript function that will respond to a particular client-side event. For example, to assign a JavaScript function called **beforeDeleteCalendar** to respond before the calendar is deleted, you would set the **OnClientBeforeDeleteCalendar** property to **beforeDeleteCalendar**.

The following table lists the events that you can use in your client scripts. These properties are defined on the server side, but the actual events or the name you declare for each JavaScript function are defined on the client side.

Event Server-Side Property Name	Event Name	Description
---------------------------------	------------	-------------

OnClientBeforeDeleteCalendar	beforeDeleteCalendar	Occurs before the delete calendar action.
OnClientBeforeDeleteEvent	beforeDeleteEvent	Occurs before the delete action.
OnClientBeforeSaveCalendar		Occurs before the save calendar action.
OnClientBeforeSaveEvent		Occurs before the save event action.
OnClientCalendarsChanged	calendarsChanged	Occurs when the calendars option has been changed.
OnClientInitialized	initialized	Occurs when the events calendar is constructed and events data is loaded from an external or local data source.
OnClientSelectedDatesChanged	selectedDatesChanged	Occurs after the date value changed.
OnClientViewTypeChanged	viewTypeChanged	Occurs when the viewType option has been changed.

Descriptions and syntax examples for the **C1EventsCalendar** client-side events can also be found at [http://wijmo.com/wiki/index.php/Events\\_Calendar](http://wijmo.com/wiki/index.php/Events_Calendar).

## See Also

[Task-Based Help](#)

## Task-Based Help

The task-based help section assumes that you are familiar with programming in the Visual Studio ASP.NET environment and have a general understanding of the **EventsCalendar for ASP.NET Web Forms** control.

Each topic provides a solution for specific tasks using the [C1EventsCalendar](#) control. By following the steps outlined in each topic, you will be able to create projects using a variety of [C1EventsCalendar](#) features.

Each task-based help topic also assumes that you have created a new ASP.NET project.

## See Also

[Working with Themes](#)

[Changing the Start Date of the C1EventsCalendar](#)

[Changing the Culture of a Schedule](#)

[Hiding the Header Bar](#)

[Hiding the Navigation Bar](#)

# Working with Themes

The topics in this section illustrate how to utilize built-in themes and custom themes.

## See Also

[Using a Built-In Theme](#)

[Using ThemeRoller for Visual Studio](#)


[Using a Custom Theme](#)

# Using a Built-In Theme

A [C1EventsCalendar](#) control has six embedded themes that you can apply with just a few clicks. This topic illustrates how to change the theme in Design view and in code. For more information on themes, see [EventsCalendar for ASP.NET Web Forms Appearance](#).

### Changing the Theme in Design View

Complete the following steps:

1. Click the **C1EventsCalendar** smart tag () to open the **C1EventsCalendar Tasks** menu.
2. Click the **Theme** drop-down arrow and select a theme from the list. For this example, select **rocket**. The **rocket** theme is applied to the [C1EventsCalendar](#) control.

### Changing the Theme in Code

Complete the following steps:

1. Import the following namespace into your project:

To write code in Visual Basic

Visual Basic	Copy Code
<pre>Imports C1.Web.Wijmo.Controls</pre>	

To write code in C#

C#	Copy Code
<pre>using C1.Web.Wijmo.Controls;</pre>	

2. Add the following code, which sets the [Theme](#) property, to the **Page\_Load** event:

To write code in Visual Basic

Visual Basic	Copy Code
<pre>C1EventsCalendar1.Theme = "rocket"</pre>	

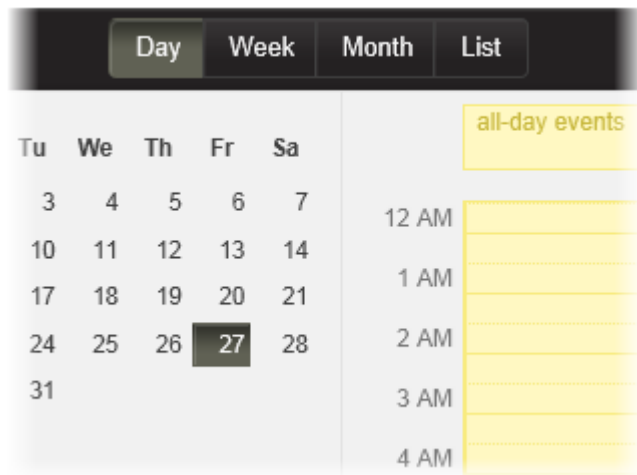
To write code in C#

C#	Copy Code
<pre>C1EventsCalendar1.Theme = "rocket";</pre>	

3. Run the program.

**This topic illustrates the following:**

The following image shows a **C1EventsCalendar** control with the **rocket** theme:



## See Also

[Using a Custom Theme](#)

## Using ThemeRoller for Visual Studio

The new **ThemeRoller for Visual Studio** makes designing beautiful themes for **ASP.NET Web Forms Edition** controls easy. For more information on creating and editing a **ThemeRoller for Visual Studio** theme, please see [ThemeRoller for Visual Studio](#).

## Using a Custom Theme

**EventsCalendar for ASP.NET Web Forms** provides six built-in themes, but if you prefer to use a different theme, you can choose an existing theme using a CDN URL or create your own custom theme with the jQuery ThemeRoller Web application. We will use [C1EventsCalendar](#) in the following examples.

### Using a CDN URL

Complete the following steps:

1. Click the [C1EventsCalendar](#) smart tag to open the **C1EventsCalendar Tasks** menu.
2. Select the **Use CDN** check box.
3. In the **Theme** property, enter a CDN URL to specify the theme; CDN URLs can be found at <http://blog.jqueryui.com/>. In this example, we'll use the *trontastic* theme: <http://ajax.googleapis.com/ajax/libs/jqueryui/1.8.17/themes/cupertino/jquery-ui.css>. This theme setting is stored in the `<appSettings>` of the **Web.config** file. In the Solution Explorer, double-click the **Web.config** file. Notice the `<appSettings>` tag contains a **WijmoTheme** key and value; this is where the CDN URL you added is specified.
4. Run the project and notice the theme is applied to [C1EventsCalendar](#).

### Using jQuery ThemeRoller

Complete the following steps:



1. Go to <http://jqueryui.com/themeroller/>.
2. On the **Roll Your Own** tab, change the settings to create a custom theme; you can customize fonts, colors, backgrounds, borders, and more. Or click the **Gallery** tab and select an existing theme.
3. Click the **Download** button and then click **Download** again on the **Build Your Download** page.
4. Save and unzip the theme .zip file to a folder within your Visual Studio project folder. In this example, we created a **customtheme** folder.
5. In the Solution Explorer, click **Show All Files** and then right-click the **customtheme** folder and select **Include in Project**.
6. Click the [C1EventsCalendar](#) smart tag to open the **Tasks** menu.
7. Select the **Use CDN** check box.
8. In the **Theme** property, enter the path to your custom theme .css; for example, **custom-theme\css\custom-theme/jquery-ui-1.8.17.custom.css**. This theme setting is stored in the `<appSettings>` of the **Web.config** file. In the Solution Explorer, double-click the **Web.config** file. Notice the `<appSettings>` tag contains a **WijmoTheme** key and value; this is where the custom theme you added is specified.
9. Run the project and notice the theme is applied to [C1EventsCalendar](#).

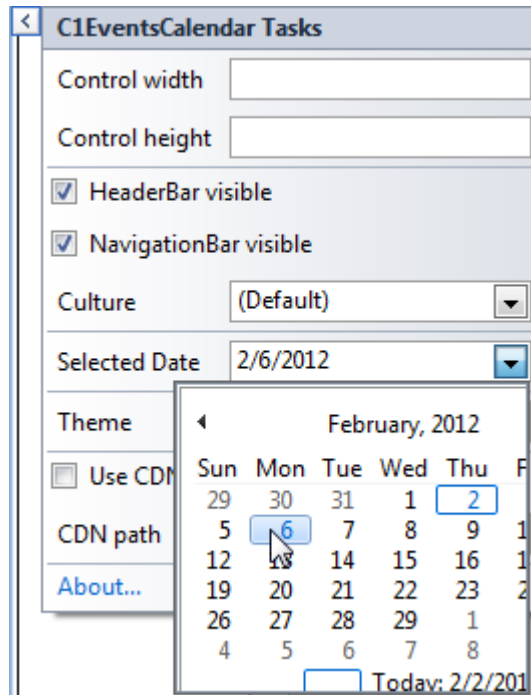
## See Also

[Changing the Start Date of the C1EventsCalendar](#)

# Changing the Start Date of the C1EventsCalendar

By default, the *Day* view of the [C1EventsCalendar](#) starts with the current date. You can change the date using the [SelectedDate](#) property.

1. Right-click the [C1EventsCalendar](#) control and then click **Show Smart Tag** to view the **C1EventsCalendar Tasks** menu.
2. Click the drop-down arrow next to the **Selected Date** item.
3. Select a date from the calendar.



This will set the [SelectedDate](#) property and the [C1EventsCalendar](#) will now begin on the date specified.

**Note:** Setting the [SelectedDate](#) property has no effect on the *Month* view. If you use the *Week* view, the start date of the calendar becomes the Sunday of the week of the [SelectedDate](#).

## See Also

[Changing the Culture of a Schedule](#)

# Changing the Culture of a Schedule

Each culture has different conventions and formats for displaying dates, time, numbers, and other information. This topic illustrates several different ways to set the culture of a [C1EventsCalendar](#) using the [Culture](#) property.

## Setting the Culture in Design View

To set the culture of a [C1EventsCalendar](#) control, complete the following steps:

1. Right-click the [C1EventsCalendar](#) control and then click **Show Smart Tag** to view the **C1EventsCalendar Tasks** menu.
2. Click the drop-down arrow next to the **Culture** item.
3. Select a culture from the list, for example **ja-JP**. Your [C1EventsCalendar](#) now uses the specified culture format, in this example, Japanese.

## Setting the Culture in Source View

To set the culture in Source view, add `Culture="ja-JP"` to the `<cc1:C1EventsCalendar>` tag. Once you've set this property, your markup will resemble the following:

[To write code in Source View](#)

```
<cc1:C1EventsCalendar ID="C1EventsCalendar1" runat="server" Culture="ja-JP">
```

### Setting the Culture Programmatically

To set the culture programmatically, add the following code into the **Page\_Load** event:

[To write code in Visual Basic](#)

Visual Basic	Copy Code
<pre>CC1EventsCalendar1.Culture = new System.Globalization.CultureInfo("ja-JP")</pre>	

[To write code in C#](#)

C#	Copy Code
<pre>C1EventsCalendar1.Culture = new System.Globalization.CultureInfo("ja-JP");</pre>	

## See Also

[Hiding the Header Bar](#)

# Hiding the Header Bar

By default the [C1EventsCalendar](#) control displays a header bar at the top of the control. The Header bar contains tabs for the calendar views, allowing you to change the current view at run time. You can disable the Header bar by setting the [HeaderBarVisible](#) property to **False**.

### Hiding the Header Bar in Design View

To hide the Header bar at design time, complete the following steps:

1. Right-click the [C1EventsCalendar](#) control and then click **Show Smart Tag** to view the **C1EventsCalendar Tasks** menu.
2. In the **C1EventsCalendar Tasks** menu, clear the **HeaderBar visible** check box. Your [C1EventsCalendar](#) control will no longer show a Header bar at run time.

### Hiding the Header Bar in Source View

To hide the Header bar in Source view, add `HeaderBarVisible="False"` to the `<cc1:C1EventsCalendar>` tag. Once you've set this property, your markup will resemble the following:

[To write code in Source View](#)

```
<cc1:C1EventsCalendar ID="C1EventsCalendar1" runat="server" HeaderBarVisible="False">
```

### Hiding the Header Bar in Code

To hide the Header bar programmatically, add the following code into the `Page_Load` event:

[To write code in Visual Basic](#)

Visual Basic	Copy Code
<pre>C1EventsCalendar1.HeaderBarVisible = False</pre>	

[To write code in C#](#)

C#	Copy Code
<pre>C1EventsCalendar1.HeaderBarVisible = false;</pre>	

## See Also

[Hiding the Navigation Bar](#)

# Hiding the Navigation Bar

By default the [C1EventsCalendar](#) control displays a navigation bar at the bottom of the control. The navigation bar notes the selected date and lets the user navigate to previous or following dates or months at run time. You can disable the navigation bar by setting the [NavigationBarVisible](#) property to **False**.

### Hiding the Navigation Bar in Design View

To hide the navigation bar at design time, complete the following steps:

1. Right-click the [C1EventsCalendar](#) control and then click **Show Smart Tag** to view the **C1EventsCalendar Tasks** menu.
2. In the **C1EventsCalendar Tasks** menu, clear the **NavigationBar visible** check box. Your [C1EventsCalendar](#) control will no longer show a navigation bar at run time.

### Hiding the Navigation Bar in Source View

To hide the navigation bar in Source view, add `NavigationBarVisible="False"` to the `<cc1:C1EventsCalendar>` tag. Once you've set this property, your markup will resemble the following:

[To write code in Source View](#)

```
<cc1:C1EventsCalendar ID="C1EventsCalendar1" runat="server" NavigationBarVisible="False">
```

### Hiding the Navigation Bar in Code

To hide the navigation bar programmatically, add the following code in the **Page\_Load** event:

[To write code in Visual Basic](#)

Visual Basic	Copy Code
<pre>C1EventsCalendar1.NavigationBarVisible = False</pre>	

[To write code in C#](#)

C#	Copy Code
<pre>C1EventsCalendar1.NavigationBarVisible = false;</pre>	

## Client-Side Reference

As part of the amazing [ComponentOne Web stack](#), the Wijmo jQuery UI widgets are optimized for client-side Web development and utilize the power of jQuery for superior performance and ease of use.

The ComponentOne Wijmo website at <http://wijmo.com/widgets/> provides everything you need to know about Wijmo widgets, including demos and samples, documentation, theming examples, support and more.

The client-side documentation provides an overview, sample markup, options, events, and methods for each widget. To get started with client-side Web development for **EventsCalendar for ASP.NET Web Forms**, click one of the external links to view our Wijmo wiki documentation. Note that the **Overview** topic for each of the widgets applies mainly to the widget, not to the server-side ASP.NET Web Forms control.

- [wijevcal documentation](#)
- [wijevcal API](#)

## See Also

[Using the Wijmo CDN](#)

# Using the Wijmo CDN

You can easily load the client-side Wijmo widgets into your web page using a Content Delivery Network (CDN). CDN makes it quick and easy to use external libraries, and deploy them to your users. A CDN is a network of computers around the world that host content. Ideally, if you're in the United States and you access a webpage using a CDN, you'll get your content from a server based in the US. If you're in India or China, and you access the SAME webpage, the content will come from a server a little closer to your location.

When web browsers load content, they commonly will check to see if they already have a copy of the file cached. By using a CDN, you can benefit from this. If a user had previously visited a site using the same CDN, they will already have a cached version of the files on their machine. Your page will load quicker since it doesn't need to re-download your support content.

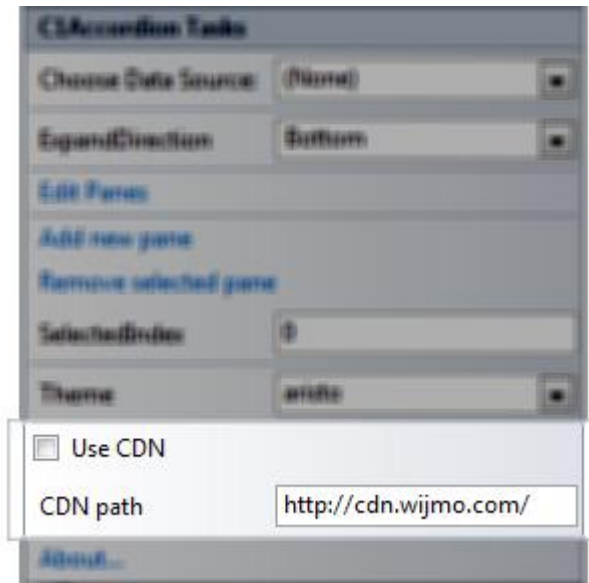
Wijmo has had CDN support from the very beginning. You can find the CDN page at <http://wijmo.com/downloads/cdn/>. The markup required for loading Wijmo into your page looks similar to this:

[To write code in Source View](#)

```
<!--jQuery References-->
<script src="http://ajax.aspnetcdn.com/ajax/jquery/jquery-1.7.1.min.js"
        type="text/javascript"></script>
<script src="http://ajax.aspnetcdn.com/ajax/jquery.ui/1.8.17/jquery-
ui.min.js"
        type="text/javascript"></script>
<!--Theme-->
<link href="http://cdn.wijmo.com/themes/rocket/jquery-wijmo.css"
      rel="stylesheet" type="text/css" title="rocket-jqueryui" />
<!--Wijmo Widgets CSS-->
<link href="http://cdn.wijmo.com/jquery.wijmo-complete.all.2.0.0.min.css"
      rel="stylesheet"
        type="text/css" />
<!--Wijmo Widgets JavaScript-->
<script src="http://cdn.wijmo.com/jquery.wijmo-open.all.2.0.0.min.js"
        type="text/javascript"></script>
<script src="http://cdn.wijmo.com/jquery.wijmo-complete.all.2.0.0.min.js"
        type="text/javascript"></script>
```

In this markup, you'll notice that some of the .js files are labeled as \*.min.js. These files have been minified - in other words, all unnecessary characters have been removed - to make the pages load faster. You will also notice that there are no references to individual .js files. The JavaScript for all widgets, CSS, and jQuery references have been combined into one file, respectively, such as wijmo-complete.2.0.0.min.js. If you want to link to individual .js files, see the **Dependencies** topic for each widget.

With the **ASP.NET Web Forms Edition** controls, you can click the **Use CDN** checkbox in the control's **Tasks** menu and specify the **CDN path** if you want to access the client-side widgets.






## **C1.Web.Wijmo.Controls.4 Assembly**

### C1.Web.Wijmo.Controls Namespace

### Overview

[Inheritance Hierarchy](#)

### Classes

	Class	Description
	<a href="#">C1BaseStateManager</a>	Base class to support view state management for a server control.
	<a href="#">C1TargetControlBase</a>	Class to be used as base for the wijmo extenders target controls. Class to be used as base for the wijmo target controls.
	<a href="#">WijChartNavigator</a>	

### See Also

#### Reference

[C1.Web.Wijmo.Controls.4 Assembly](#)

# Classes

## C1BaseStateManager

Base class to support view state management for a server control.

### Object Model

C1BaseStateManager

### Syntax

Visual Basic (Declaration)	
Public MustInherit Class C1BaseStateManager	
C#	
public abstract class C1BaseStateManager	

### Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1BaseStateManager**

### See Also

#### Reference

[C1BaseStateManager Members](#)

[C1.Web.Wijmo.Controls Namespace](#)

### Overview

Base class to support view state management for a server control.

### Object Model

C1BaseStateManager

### Syntax

Visual Basic (Declaration)	
Public MustInherit Class C1BaseStateManager	



C#	
----	--

public abstract class C1BaseStateManager
--

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1BaseStateManager**

## See Also

### Reference

[C1BaseStateManager Members](#)

[C1.Web.Wijmo.Controls Namespace](#)

## Members

The following tables list the members exposed by [C1BaseStateManager](#).

## See Also

### Reference

[C1BaseStateManager Class](#)

[C1.Web.Wijmo.Controls Namespace](#)

## C1TargetControlBase

Class to be used as base for the wijmo extenders target controls. Class to be used as base for the wijmo target controls.

## Object Model

C1TargetControlBase

## Syntax

Visual Basic (Declaration)	
----------------------------	--

<WidgetDependenciesAttribute(System.Object[])> Public MustInherit Class C1TargetControlBase Inherits System.Web.UI.WebControls.WebControl
---

C#	
----	--

[WidgetDependencies(System.Object[])]
---------------------------------------

```
public abstract class C1TargetControlBase :
System.Web.UI.WebControls.WebControl
```

## Inheritance Hierarchy

```
System.Object
  System.Web.UI.Control
    System.Web.UI.WebControls.WebControl
      C1.Web.Wijmo.Controls.C1TargetControlBase
        C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendar
```

## See Also

### Reference

[C1TargetControlBase Members](#)

[C1.Web.Wijmo.Controls Namespace](#)

## Overview

Class to be used as base for the wijmo extenders target controls. Class to be used as base for the wijmo target controls.

## Object Model

C1TargetControlBase

## Syntax

Visual Basic (Declaration)

```
<WidgetDependenciesAttribute(System.Object[])>
Public MustInherit Class C1TargetControlBase
    Inherits System.Web.UI.WebControls.WebControl
```

C#

```
[WidgetDependencies(System.Object[])]
public abstract class C1TargetControlBase :
System.Web.UI.WebControls.WebControl
```

## Inheritance Hierarchy

```
System.Object
  System.Web.UI.Control
    System.Web.UI.WebControls.WebControl
```

**C1.Web.Wijmo.Controls.C1TargetControlBase**[C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendar](#)

## See Also











### Reference















[C1TargetControlBase Members](#)[C1.Web.Wijmo.Controls Namespace](#)













## Members





The following tables list the members exposed by [C1TargetControlBase](#).

### Public Properties

	Name	Description
	<a href="#">AccessKey</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">AppRelativeTemplateSourceDirectory</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )
	<a href="#">Attributes</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">CDNDependencyPaths</a>	A string array value specifies all css/js references that wijmo control depends on.
	<a href="#">CDNPath</a>	Content Delivery Network path.
	<a href="#">ClientID</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )
	<a href="#">ClientIDMode</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )
	<a href="#">Controls</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )
	<a href="#">ControlStyle</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">ControlStyleCreated</a>	(Inherited from











		System.Web.UI.WebControls.WebControl)
	CssClass	(Inherited from System.Web.UI.WebControls.WebControl)
	DisplayVisible	Gets or sets a value that indicates whether a server control is rendered as UI on the page.
	EnableCombinedJavaScripts	Enable JavaScripts files combined into one file. In order to combined enabled, you must register the WijmoHttpHandler in web.config.
	Enabled	Overridden. Gets or sets a value indicating whether the Web server control is enabled.
	EnableTheming	(Inherited from System.Web.UI.WebControls.WebControl)
	EnableViewState	(Inherited from System.Web.UI.Control)
	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	Height	(Inherited from System.Web.UI.WebControls.WebControl)
	ID	(Inherited from System.Web.UI.Control)
	NamingContainer	(Inherited from System.Web.UI.Control)
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	Site	(Inherited from System.Web.UI.Control)

	<a href="#">SkinID</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">Style</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">SupportsDisabledAttribute</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">TabIndex</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">TemplateControl</a>	(Inherited from System.Web.UI.Control)
	<a href="#">TemplateSourceDirectory</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Theme</a>	Name of the theme that will be used to style the widgets. Available embedded themes include: aristo, midnight, and ui-lightness. Note that only one theme can be used for the whole page at one time. Please make sure that all widget extenders have the same Theme value.
	<a href="#">ThemeSwatch</a>	A value that indicates the theme swatch of the control, this property only works when WijmoControlMode property is Mobile.
	<a href="#">ToolTip</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">UniqueID</a>	(Inherited from System.Web.UI.Control)
	<a href="#">UseCDN</a>	Determines whether the widget extender must load client resources from CDN (Content Delivery Network) path given by property CDNPath.
	<a href="#">ValidateRequestMode</a>	(Inherited from System.Web.UI.Control)

	<a href="#">ViewStateMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Width</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">WijmoControlMode</a>	A value that indicates mode of the control, whether it is a mobile or web control. Note that only one value can be used for the whole website or project.
	<a href="#">WijmoCssAdapter</a>	Indicates the control applies the theme of JQuery UI or Bootstrap.

[Top](#)

## Public Methods



	Name	Description
	<a href="#">ApplyStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">ApplyStyleSheetSkin</a>	(Inherited from System.Web.UI.Control)
	<a href="#">CopyBaseAttributes</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">DataBind</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Dispose</a>	(Inherited from System.Web.UI.Control)
	<a href="#">FindControl</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Focus</a>	(Inherited from System.Web.UI.Control)
	<a href="#">GetPropertyValue&lt;V&gt;</a>	Gets the property value by property name.
	<a href="#">GetRouteUrl</a>	Overloaded. (Inherited from System.Web.UI.Control)
	<a href="#">GetScriptDescriptors</a>	When overridden in a derived class, registers the WidgetDescriptor objects for the control.

⇒	<a href="#">GetScriptReferences</a>	When overridden in a derived class, registers an additional script libraries for the extender control.
⇒	<a href="#">GetUniqueIDRelativeTo</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">HasControls</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">MergeStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	<a href="#">RegisterDesignTimeStyleSheet</a>	Register design time stylesheet
⇒	<a href="#">RenderBeginTag</a>	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	<a href="#">RenderControl</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">RenderEndTag</a>	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	<a href="#">ResolveClientUrl</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">ResolveUrl</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">SetPropertyValue&lt;V&gt;</a>	Sets the property value by property name.
⇒	<a href="#">SetRenderMethodDelegate</a>	(Inherited from System.Web.UI.Control)

[Top](#)

## Public Events

	Name	Description
⚡	<a href="#">DataBinding</a>	(Inherited from System.Web.UI.Control)
⚡	<a href="#">Disposed</a>	(Inherited from System.Web.UI.Control)
⚡	<a href="#">Init</a>	(Inherited from System.Web.UI.Control)
⚡	<a href="#">Load</a>	(Inherited from System.Web.UI.Control)

	<a href="#">PreRender</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Unload</a>	(Inherited from System.Web.UI.Control)

[Top](#)

## See Also

### Reference











[C1TargetControlBase Class](#)

[C1.Web.Wijmo.Controls Namespace](#)

## Methods

For a list of all members of this type, see [C1TargetControlBase members](#).

## Public Methods

	Name	Description
	<a href="#">ApplyStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">ApplyStyleSheetSkin</a>	(Inherited from System.Web.UI.Control)
	<a href="#">CopyBaseAttributes</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">DataBind</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Dispose</a>	(Inherited from System.Web.UI.Control)
	<a href="#">FindControl</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Focus</a>	(Inherited from System.Web.UI.Control)
	<a href="#">GetPropertyValue&lt;V&gt;</a>	Gets the property value by property name.
	<a href="#">GetRouteUrl</a>	Overloaded. (Inherited from System.Web.UI.Control)
	<a href="#">GetScriptDescriptors</a>	When overridden in a derived class, registers the WidgetDescriptor objects for the control.



⇒	<a href="#">GetScriptReferences</a>	When overridden in a derived class, registers an additional script libraries for the extender control.
⇒	<a href="#">GetUniqueIDRelativeTo</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">HasControls</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">MergeStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	<a href="#">RegisterDesignTimeStyleSheet</a>	Register design time stylesheet
⇒	<a href="#">RenderBeginTag</a>	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	<a href="#">RenderControl</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">RenderEndTag</a>	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	<a href="#">ResolveClientUrl</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">ResolveUrl</a>	(Inherited from System.Web.UI.Control)
⇒	<a href="#">SetPropertyValue&lt;V&gt;</a>	Sets the property value by property name.
⇒	<a href="#">SetRenderMethodDelegate</a>	(Inherited from System.Web.UI.Control)

[Top](#)

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1.Web.Wijmo.Controls Namespace](#)

## GetPropertyValue<V> Method

Gets the property value by property name.

## Syntax

Visual Basic (Declaration)

```
Public Function GetPropertyValue(Of V)( _
    ByVal propertyName As System.String, _
    ByVal nullValue As V _
) As V
```

C#

```
public V GetPropertyValue<V>(
    System.string propertyName,
    V nullValue
)
```

### Parameters

*propertyName*

Name of the property.

*nullValue*

The null value.

### Type Parameters

*V*

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## GetScriptDescriptors Method

When overridden in a derived class, registers the WidgetDescriptor objects for the control.

### Syntax

Visual Basic (Declaration)

```
Public Overridable Function GetScriptDescriptors() As
System.Collections.Generic.IEnumerable(Of ScriptDescriptor)
```

C#

```
public virtual System.Collections.Generic.IEnumerable<ScriptDescriptor>
GetScriptDescriptors()
```

## Return Value

An enumeration of WidgetDescriptor objects.

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## GetScriptReferences Method

When overridden in a derived class, registers an additional script libraries for the extender control.

## Syntax

Visual Basic (Declaration)	
Public Overridable Function GetScriptReferences() As System.Collections.Generic.IEnumerable(Of ScriptReference)	
C#	
public virtual System.Collections.Generic.IEnumerable<ScriptReference> GetScriptReferences()	

### Return Value

An object that implements the **System.Collections.IEnumerable** interface and that contains ECMAScript (JavaScript) files that have been registered as embedded resources.

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## RegisterDesignTimeStyleSheet Method

The **System.Web.UI.HtmlTextWriter** object that receives the server control content.

resource name of stylesheet file

Register design time stylesheet

## Syntax

## Visual Basic (Declaration)

```
Public Sub RegisterDesignTimeStyleSheet( _
    ByVal writer As System.Web.UI.HtmlTextWriter, _
    ByVal cssResourceName As System.String _
)
```

## C#

```
public void RegisterDesignTimeStyleSheet(
    System.Web.UI.HtmlTextWriter writer,
    System.string cssResourceName
)
```

**Parameters***writer*

The **System.Web.UI.HtmlTextWriter** object that receives the server control content.

*cssResourceName*

resource name of stylesheet file

**See Also****Reference**

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

**SetPropertyValue<V> Method**

Sets the property value by property name.

**Syntax**

## Visual Basic (Declaration)

```
Public Sub SetPropertyValue(Of V)( _
    ByVal propertyName As System.String, _
    ByVal value As V _
)
```

## C#

```
public void SetPropertyValue<V>(
    System.string propertyName,
```

```

    V value
)

```

### Parameters

*propertyName*

Name of the property.

*value*

The value.

### Type Parameters

*V*

## See Also

### Reference






[C1TargetControlBase Class](#)















[C1TargetControlBase Members](#)














## Properties









For a list of all members of this type, see [C1TargetControlBase members](#).

## Public Properties

	Name	Description
	<a href="#">AccessKey</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">AppRelativeTemplateSourceDirectory</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Attributes</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">CDNDependencyPaths</a>	A string array value specifies all css/js references that wijmo control depends on.
	<a href="#">CDNPath</a>	Content Delivery Network path.

	<a href="#">ClientID</a>	(Inherited from System.Web.UI.Control)
	<a href="#">ClientIDMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Controls</a>	(Inherited from System.Web.UI.Control)
	<a href="#">ControlStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">ControlStyleCreated</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">CssClass</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">DisplayVisible</a>	Gets or sets a value that indicates whether a server control is rendered as UI on the page.
	<a href="#">EnableCombinedJavaScripts</a>	Enable JavaScripts files combined into one file. In order to combined enabled, you must register the WijmoHttpHandler in web.config.
	<a href="#">Enabled</a>	Overridden. Gets or sets a value indicating whether the Web server control is enabled.
	<a href="#">EnableTheming</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">EnableViewState</a>	(Inherited from System.Web.UI.Control)
	<a href="#">HasAttributes</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">Height</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">ID</a>	(Inherited from System.Web.UI.Control)

	NamingContainer	(Inherited from System.Web.UI.Control)
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	Site	(Inherited from System.Web.UI.Control)
	SkinID	(Inherited from System.Web.UI.WebControls.WebControl)
	Style	(Inherited from System.Web.UI.WebControls.WebControl)
	SupportsDisabledAttribute	(Inherited from System.Web.UI.WebControls.WebControl)
	TabIndex	(Inherited from System.Web.UI.WebControls.WebControl)
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Theme	Name of the theme that will be used to style the widgets. Available embedded themes include: aristo, midnight, and ui-lightness. Note that only one theme can be used for the whole page at one time. Please make sure that all widget extenders have the same Theme value.
	ThemeSwatch	A value that indicates the theme swatch of the control, this property only works when WijmoControlMode property is Mobile.

	<a href="#">ToolTip</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">UniqueID</a>	(Inherited from System.Web.UI.Control)
	<a href="#">UseCDN</a>	Determines whether the widget extender must load client resources from CDN (Content Delivery Network) path given by property CDNPath.
	<a href="#">ValidateRequestMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">ViewStateMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Width</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">WijmoControlMode</a>	A value that indicates mode of the control, whether it is a mobile or web control. Note that only one value can be used for the whole website or project.
	<a href="#">WijmoCssAdapter</a>	Indicates the control applies the theme of JQuery UI or Bootstrap.

[Top](#)

## See Also

### Reference

[C1TargetControlBase Class](#)[C1.Web.Wijmo.Controls Namespace](#)

## CDNDependencyPaths Property

A string array value specifies all css/js references that wijmo control depends on.

## Syntax

Visual Basic (Declaration)

```

<C1CategoryAttribute("Behavior")>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>

```



```
<C1DescriptionAttribute(Key="C1Base.CDNDependencyPaths", Description="A
string array value specifies all css/js references that wijmo control depends
on.")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Hidden)>
Public Overridable Property CDNDependencyPaths As System.String()
```

C#

```
[C1Category("Behavior")]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[C1Description(Key="C1Base.CDNDependencyPaths", Description="A string array
value specifies all css/js references that wijmo control depends on.")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
public virtual System.string[] CDNDependencyPaths {get; set;}
```

## Remarks

Using this property, user can specify the CDN dependencies(such as jQuery, jQuery UI, jQuery mobile, bootstrap). If some dependencies are set in this property, wijmo controls will find dependencies path in this property. If found, use the path to load dependency, otherwise, use the default CDN path. For example, if user wants to specify the dependencies of jQuery and jQuery UI, he can set the value like this: ["http://ajax.aspnetcdn.com/ajax/jQuery/jquery-1.11.1.min.js", "http://ajax.aspnetcdn.com/ajax/jquery.ui/1.11.1/jquery-ui.js"]

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## CDNPath Property

Content Delivery Network path.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
```

```
<C1DescriptionAttribute(Key="C1Base.CDNPath", Description="Content Delivery
Network path.")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Hidden)>
Public Overridable Property CDNPath As System.String
```

C#

```
[C1Category("Behavior")]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[C1Description(Key="C1Base.CDNPath", Description="Content Delivery Network
path.")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
public virtual System.string CDNPath {get; set;}
```

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## DisplayVisible Property

Gets or sets a value that indicates whether a server control is rendered as UI on the page.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1Controls.DisplayVisible", Description="Gets or
sets a value that indicates whether a server control is rendered as UI on the
page.")>
<C1CategoryAttribute("Behavior")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property DisplayVisible As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
```

```
[WidgetOption()]
[C1Description(Key="C1Controls.DisplayVisible", Description="Gets or sets a
value that indicates whether a server control is rendered as UI on the
page.")]
[C1Category("Behavior")]
[Layout(LayoutType.Behavior)]
public System.bool DisplayVisible {get; set;}
```

## Remarks

Note that a server control is created and invisible if DisplayVisible is set to false.

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## EnableCombinedJavaScripts Property

Enable JavaScripts files combined into one file. In order to combined enabled, you must register the WijmoHttpHandler in web.config.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Base.EnableCombinedJavaScripts",
Description="Enable JavaScripts files combined into one file.")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Hidden)>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
Public Overridable Property EnableCombinedJavaScripts As System.Boolean
```

C#

```
[C1Description(Key="C1Base.EnableCombinedJavaScripts", Description="Enable
JavaScripts files combined into one file.")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
public virtual System.bool EnableCombinedJavaScripts {get; set;}
```

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## Enabled Property

Gets or sets a value indicating whether the Web server control is enabled.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1Base.Enabled", Description="Indicates whether
control is enabled.")>
<C1CategoryAttribute("Behavior")>
Public Overrides Property Enabled As System.Boolean
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1Base.Enabled", Description="Indicates whether control
is enabled.")]
[C1Category("Behavior")]
public override System.bool Enabled {get; set;}
```

### Property Value

true if control is enabled; otherwise, false. The default is true.

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## Theme Property

Name of the theme that will be used to style the widgets. Available embedded themes include: aristo, midnight, and ui-lightness. Note that only one theme can be used for the whole page at one time. Please make sure that all widget extenders have the same Theme value.

## Syntax

### Visual Basic (Declaration)

```
<TypeConverterAttribute("C1.Web.Wijmo.Controls.WijmoThemeNameConverter,
C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Hidden)>
<C1DescriptionAttribute(Key="C1Base.Theme", Description="Name of the theme
that will be used to style widgets. Available themes: aristo / midnight / ui-
lightness. Note, only one theme can be used for the whole page at one time.
Please, make sure that all widget extenders have the same Theme value.")>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
Public Overridable Property Theme As System.String
```

### C#

```
[TypeConverter("C1.Web.Wijmo.Controls.WijmoThemeNameConverter,
C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
[C1Description(Key="C1Base.Theme", Description="Name of the theme that will
be used to style widgets. Available themes: aristo / midnight / ui-lightness.
Note, only one theme can be used for the whole page at one time. Please, make
sure that all widget extenders have the same Theme value.")]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
public virtual System.string Theme {get; set;}
```

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## ThemeSwatch Property

A value that indicates the theme swatch of the control, this property only works when WijmoControlMode property is Mobile.

## Syntax

### Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Hidden)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1Base.ThemeSwatch", Description="A value that indicates the theme swatch of the control, this property only works when WijmoControlMode property is Mobile.")>
<TypeConverterAttribute("C1.Web.Wijmo.Controls.WijmoThemeSwatchConverter, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral, PublicKeyToken=9b75583953471eea")>
Public Overridable Property ThemeSwatch As System.String
```

### C#

```
[C1Category("Appearance")]
[WidgetOption()]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Hidden)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1Base.ThemeSwatch", Description="A value that indicates the theme swatch of the control, this property only works when WijmoControlMode property is Mobile.")]
[TypeConverter("C1.Web.Wijmo.Controls.WijmoThemeSwatchConverter, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral, PublicKeyToken=9b75583953471eea")]
public virtual System.string ThemeSwatch {get; set;}
```

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## UseCDN Property

Determines whether the widget extender must load client resources from CDN (Content Delivery Network) path given by property CDNPath.

## Syntax

## Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Base.UseCDN", Description="Determines whether
this extender loads client script references from CDN.")>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Hidden)>
Public Overridable Property UseCDN As System.Boolean
```

## C#

```
[C1Description(Key="C1Base.UseCDN", Description="Determines whether this
extender loads client script references from CDN.")]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
public virtual System.bool UseCDN {get; set;}
```

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## WijmoControlMode Property

A value that indicates mode of the control, whether it is a mobile or web control. Note that only one value can be used for the whole website or project.

## Syntax

## Visual Basic (Declaration)

```
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1Base.WijmoControlMode", Description="A value
that indicates mode of the control, whether it is a mobile or web control.
Note that only one value can be used for the whole website or project.")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
```

```
lizationVisibility.Hidden)>
Public Overridable Property WijmoControlMode As WijmoControlMode
```

C#

```
[C1Category("Behavior")]
[C1Description(Key="C1Base.WijmoControlMode", Description="A value that
indicates mode of the control, whether it is a mobile or web control. Note
that only one value can be used for the whole website or project.")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
public virtual WijmoControlMode WijmoControlMode {get; set;}
```

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

## WijmoCssAdapter Property

Indicates the control applies the theme of JQuery UI or Bootstrap.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1Base.WijmoCssAdapter", Description="Indicates
the control applies the theme of JQuery UI or Bootstrap.")>
<System.ComponentModel.DefaultValueAttribute()>
<TypeConverterAttribute("C1.Web.Wijmo.Controls.WijmoCssAdapterConverter,
C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Hidden)>
Public Overridable Property WijmoCssAdapter As System.String
```

C#

```
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
```



```
[C1Category("Behavior")]
[C1Description(Key="C1Base.WijmoCssAdapter", Description="Indicates the
control applies the theme of JQuery UI or Bootstrap.")]
[System.ComponentModel.DefaultValue()]
[TypeConverter("C1.Web.Wijmo.Controls.WijmoCssAdapterConverter,
C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
public virtual System.string WijmoCssAdapter {get; set;}
```

## See Also

### Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

# WijChartNavigator

## Object Model

WijChartNavigator

## Syntax

Visual Basic (Declaration)

```
<WidgetDependenciesAttribute(System.Object[])>
Friend Class WijChartNavigator
    Inherits WijChartCore
```

C#

```
[WidgetDependencies(System.Object[])]
internal class WijChartNavigator : WijChartCore
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.WijChartNavigator**

## See Also

### Reference

[WijChartNavigator Members](#)[C1.Web.Wijmo.Controls Namespace](#)

## Overview

## Object Model

**WijChartNavigator**

## Syntax

Visual Basic (Declaration)

```
<WidgetDependenciesAttribute(System.Object[])>  
Friend Class WijChartNavigator  
    Inherits WijChartCore
```

C#

```
[WidgetDependencies(System.Object[])]  
internal class WijChartNavigator : WijChartCore
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.WijChartNavigator**

## See Also


### Reference

[WijChartNavigator Members](#)[C1.Web.Wijmo.Controls Namespace](#)

## Members

The following tables list the members exposed by [WijChartNavigator](#).

## Public Constructors

	Name	Description
	<a href="#">WijChartNavigator Constructor</a>	

[Top](#)

## See Also

### Reference

[WijChartNavigator Class](#)

[C1.Web.Wijmo.Controls Namespace](#)

## WijChartNavigator Constructor

### Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public WijChartNavigator()	

## See Also

### Reference




[WijChartNavigator Class](#)














[WijChartNavigator Members](#)


# C1.Web.Wijmo.Controls.C1EventsCalendar Namespace

## Overview







### Classes

	Class	Description
	<a href="#">C1CalendarStorage</a>	Allows binding to DataSource and mapping DataSource's fields to Calendar's properties.
	<a href="#">C1CalendarStorageMappings</a>	C1CalendarStorage mappings configuration.
	<a href="#">C1EventCollection</a>	Represents C1Event object collection.

 <a href="#">C1EventsCalendar</a>	The C1EventsCalendar control is a fully functional schedule that allows users to add, edit, and manage their appointments.
 <a href="#">C1EventsCalendarSerializer</a>	Serializes the C1EventsCalendar Control.
 <a href="#">C1EventsCalendarStorage</a>	The C1EventsCalendarStorage defines data storage configuration.
 <a href="#">C1EventsCalendarXmlExchanger</a>	Allows import/export events data to/from XML.
 <a href="#">C1EventStorage</a>	Allows binding to DataSource and mapping DataSource's fields to Event's properties.
 <a href="#">C1EventStorageMappings</a>	C1EventStorage mappings configuration.
 <a href="#">C1MappingsBase</a>	Represents a collection of mappings for properties to the appropriate data fields. Names of properties are keys in this collection.
 <a href="#">DatePagerLocalizationOption</a>	Date pager localization strings.
 <a href="#">DayViewHeaderFormatOption</a>	Format of the text for the day header in the day/week or list view.
 <a href="#">Event</a>	The <a href="#">Event</a> object is a meeting, a one-time event, or a recurring event or meeting in the C1EventsCalendar. The <a href="#">Event</a> class includes properties that specify meeting details such as the location and time.
 <a href="#">LocalizationOption</a>	Localization strings.
 <a href="#">MappingInfo</a>	Information about mapping.
 <a href="#">RecurrencePattern</a>	Represents the recurrence attributes of an event. Only master events can have this field (recurrenceState is "master").

 <a href="#">View</a>	The view can be added to C1EventsCalendar's Views.
--	--

## Enumerations

	Enumeration	Description
	<a href="#">C1EventsCalendarCustomViewUnit</a>	Specifies the unit of the calendar custom view.
	<a href="#">C1EventsCalendarViewType</a>	Specifies the type of the calendar view.
	<a href="#">FileFormatEnum</a>	Specifies the format of the exported or imported file.
	<a href="#">RecurrencePattern.RecurrenceTypeEnum</a>	Determines the types of recurrent appointments.
	<a href="#">RecurrencePattern.WeekDaysEnum</a>	Determines the set of days and groups of days for recurrence patterns.
	<a href="#">RecurrencePattern.WeekOfMonthEnum</a>	Determines the week in a month in which the appointment will occur.

## See Also

### Reference

[C1.Web.Wijmo.Controls.4 Assembly](#)

## Classes

### C1CalendarStorage

Allows binding to DataSource and mapping DataSource's fields to Calendar's properties.

## Object Model

**C1CalendarStorage**

## Syntax

Visual Basic (Declaration)

```
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<System.ComponentModel.DefaultPropertyAttribute("DataSourceID")>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
Public Class C1CalendarStorage
```

C#

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[System.ComponentModel.DefaultProperty("DataSourceID")]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
public class C1CalendarStorage
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1CalendarStorage**

## See Also

### Reference

[C1CalendarStorage Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Allows binding to DataSource and mapping DataSource's fields to Calendar's properties.

## Object Model

C1CalendarStorage

## Syntax

Visual Basic (Declaration)

```
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<System.ComponentModel.DefaultPropertyAttribute("DataSourceID")>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
Public Class C1CalendarStorage
```

C#

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[System.ComponentModel.DefaultProperty("DataSourceID")]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
public class C1CalendarStorage
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1CalendarStorage**

## See Also

### Reference


[C1CalendarStorage Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members




The following tables list the members exposed by [C1CalendarStorage](#).

### Public Constructors

	Name	Description
	<a href="#">C1CalendarStorage Constructor</a>	

[Top](#)

### Public Properties

	Name	Description
	<a href="#">DataMember</a>	The specific data member in a multimember data source.
	<a href="#">DataSourceID</a>	The ID of the control from which the events calendar control retrieves its list of data items.
	<a href="#">Mappings</a>	Use the Mappings property to bind the calendar object properties to the appropriate fields in the datasource specified by DataSourceID property.

[Top](#)

## See Also

## Reference

[C1CalendarStorage Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

# C1CalendarStorage Constructor

## Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1CalendarStorage()	

## See Also

### Reference




[C1CalendarStorage Class](#)

[C1CalendarStorage Members](#)

## Properties

For a list of all members of this type, see [C1CalendarStorage members](#).

## Public Properties

	Name	Description
	<a href="#">DataMember</a>	The specific data member in a multimember data source.
	<a href="#">DataSourceID</a>	The ID of the control from which the events calendar control retrieves its list of data items.
	<a href="#">Mappings</a>	Use the Mappings property to bind the calendar object properties to the appropriate fields in the datasource specified by DataSourceID property.

[Top](#)

## See Also

### Reference



[C1CalendarStorage Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## DataMember Property

The specific data member in a multimember data source.

### Syntax

#### Visual Basic (Declaration)

```

<C1CategoryAttribute("Data")>
<System.ComponentModel.DefaultValueAttribute()>
<EditorAttribute(EditorBaseTypeName="System.Drawing.Design.UITypeEditor,
System.Drawing, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a",
EditorTypeName="C1.Web.UI.Design.C1DataMemberEditor")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorage.DataMember",
Description="The specific data member in a multimember data source.")>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
Public Property DataMember As System.String

```

#### C#

```

[C1Category("Data")]
[System.ComponentModel.DefaultValue()]
[Editor(EditorBaseTypeName="System.Drawing.Design.UITypeEditor,
System.Drawing, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a",
EditorTypeName="C1.Web.UI.Design.C1DataMemberEditor")]
[System.ComponentModel.NotifyParentProperty(true)]
[C1Description(Key="C1EventsCalendar.C1CalendarStorage.DataMember",
Description="The specific data member in a multimember data source.")]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
public System.string DataMember {get; set;}

```

### See Also

#### Reference

[C1CalendarStorage Class](#)[C1CalendarStorage Members](#)

## DataSourceID Property

The ID of the control from which the events calendar control retrieves its list of data items.

### Syntax

#### Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorage.DataSourceID",
, Description="The ID of the control from which the events calendar control
retrieves its list of data items.")>
<C1CategoryAttribute("Data")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
<EditorAttribute(EditorBaseTypeName="System.Drawing.Design.UITypeEditor,
System.Drawing, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a",
EditorTypeName="C1.Web.UI.Design.C1DataSourceIDEditor")>
Public Property DataSourceID As System.String
```

#### C#

```
[C1Description(Key="C1EventsCalendar.C1CalendarStorage.DataSourceID",
Description="The ID of the control from which the events calendar control
retrieves its list of data items.")]
[C1Category("Data")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
[Editor(EditorBaseTypeName="System.Drawing.Design.UITypeEditor,
System.Drawing, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a",
EditorTypeName="C1.Web.UI.Design.C1DataSourceIDEditor")]
public System.string DataSourceID {get; set;}
```

### See Also

#### Reference

[C1CalendarStorage Class](#)

[C1CalendarStorage Members](#)

## Mappings Property

Use the Mappings property to bind the calendar object properties to the appropriate fields in the datasource specified by DataSourceID property.

## Syntax

### Visual Basic (Declaration)

```
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<C1CategoryAttribute("Data")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorage.Mappings",
Description="Use the Mappings property to bind the calendar object properties
to the appropriate fields in the datasource specified by DataSourceID
property.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
Public ReadOnly Property Mappings As C1CalendarStorageMappings
```

### C#

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[C1Category("Data")]
[C1Description(Key="C1EventsCalendar.C1CalendarStorage.Mappings",
Description="Use the Mappings property to bind the calendar object properties
to the appropriate fields in the datasource specified by DataSourceID
property.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
public C1CalendarStorageMappings Mappings {get;}
```

## See Also

### Reference

[C1CalendarStorage Class](#)

[C1CalendarStorage Members](#)

# C1CalendarStorageMappings

C1CalendarStorage mappings configuration.

## Object Model

C1CalendarStorageMappings

## Syntax

Visual Basic (Declaration)

```
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>  
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,  
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>  
Public Class C1CalendarStorageMappings  
    Inherits C1MappingsBase
```

C#

```
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]  
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,  
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]  
public class C1CalendarStorageMappings : C1MappingsBase
```

## Inheritance Hierarchy

System.Object

[C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase](#)

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1CalendarStorageMappings**

## See Also

### Reference

[C1CalendarStorageMappings Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

C1CalendarStorage mappings configuration.

## Object Model

C1CalendarStorageMappings

## Syntax

Visual Basic (Declaration)

```
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
Public Class C1CalendarStorageMappings
    Inherits C1MappingsBase
```

C#

```
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
public class C1CalendarStorageMappings : C1MappingsBase
```

## Inheritance Hierarchy

System.Object

[C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase](#)

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1CalendarStorageMappings**

## See Also

### Reference



[C1CalendarStorageMappings Members](#)






[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members

The following tables list the members exposed by [C1CalendarStorageMappings](#).




## Public Properties

	Name	Description
	<a href="#">ColorMapping</a>	Allows the color property to be bound to appropriate field in the data source.
	<a href="#">DescriptionMapping</a>	Allows the description property to be bound to appropriate field in the data source.

	<a href="#">IdMapping</a>	(Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
	<a href="#">LocationMapping</a>	Allows the location property to be bound to appropriate field in the data source.
	<a href="#">NameMapping</a>	Allows the name property to be bound to appropriate field in the data source.
	<a href="#">PropertiesMapping</a>	Allows other calendar properties to be bound to appropriate field in the data source.
	<a href="#">TagMapping</a>	Allows the tag property to be bound to appropriate field in the data source (the tag property can be used to store any user-defined information).

[Top](#)


## Public Methods

	Name	Description
	<a href="#">Dispose</a>	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
	<a href="#">FindByMappingName</a>	Finds the mapping info by mapping name (name of the data field). (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
	<a href="#">ToArray</a>	Creates an array of MappingInfo objects. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )

[Top](#)

## Public Events

	Name	Description
--	------	-------------

	<b>Disposed</b>	Represents the method that handles the Disposed event of a component. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
---	-----------------	---

[Top](#)

## See Also








### Reference

[C1CalendarStorageMappings Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Properties

For a list of all members of this type, see [C1CalendarStorageMappings members](#).

## Public Properties

	Name	Description
	<b>ColorMapping</b>	Allows the color property to be bound to appropriate field in the data source.
	<b>DescriptionMapping</b>	Allows the description property to be bound to appropriate field in the data source.
	<b>IdMapping</b>	(Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
	<b>LocationMapping</b>	Allows the location property to be bound to appropriate field in the data source.
	<b>NameMapping</b>	Allows the name property to be bound to appropriate field in the data source.
	<b>PropertiesMapping</b>	Allows other calendar properties to be bound to appropriate field in the data source.
	<b>TagMapping</b>	Allows the tag property to be bound to appropriate field in the data source (the tag property can be used to store any user-defined

		information).
--	--	---------------

[Top](#)

## See Also

### Reference

[C1CalendarStorageMappings Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## ColorMapping Property

Allows the color property to be bound to appropriate field in the data source.

## Syntax

Visual Basic (Declaration)	
<pre> &lt;C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorageMappings.Color Mapping", Description="Allows the color property to be bound to appropriate field in the data source.")&gt; &lt;EditorBrowsableAttribute()&gt; &lt;System.ComponentModel.BrowsableAttribute(True)&gt; &lt;System.ComponentModel.BindableAttribute(Bindable=False, Direction=BindingDirection.OneWay)&gt; &lt;System.ComponentModel.NotifyParentPropertyAttribute(True)&gt; &lt;System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)&gt; &lt;TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")&gt; &lt;C1CategoryAttribute("Mappings")&gt; Public ReadOnly Property ColorMapping As <a href="#">MappingInfo</a> </pre>	
C#	
<pre> [C1Description(Key="C1EventsCalendar.C1CalendarStorageMappings.ColorMapping", Description="Allows the color property to be bound to appropriate field in the data source.")] [EditorBrowsable()] [System.ComponentModel.Browsable(true)] [System.ComponentModel.Bindable(Bindable=false, Direction=BindingDirection.OneWay)] [System.ComponentModel.NotifyParentProperty(true)] [System.Web.UI.Persistencemode(Persistencemode.InnerProperty)] </pre>	



```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[C1Category("Mappings")]
public MappingInfo ColorMapping {get;}
```

## See Also

### Reference

[C1CalendarStorageMappings Class](#)

[C1CalendarStorageMappings Members](#)

## DescriptionMapping Property

Allows the description property to be bound to appropriate field in the data source.

## Syntax

### Visual Basic (Declaration)

```
<EditorBrowsableAttribute(>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorageMappings.Descr
iptionMapping", Description="Allows the description property to be bound to
appropriate field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property DescriptionMapping As MappingInfo
```

### C#

```
[EditorBrowsable()]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1CalendarStorageMappings.DescriptionMap
ping", Description="Allows the description property to be bound to
appropriate field in the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
```

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[System.ComponentModel.Browsable(true)]
public MappingInfo DescriptionMapping {get;}
```

## See Also

### Reference

[C1CalendarStorageMappings Class](#)

[C1CalendarStorageMappings Members](#)

## LocationMapping Property

Allows the location property to be bound to appropriate field in the data source.

## Syntax

### Visual Basic (Declaration)

```
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<System.ComponentModel.BrowsableAttribute(True)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorageMappings.Locat
ionMapping", Description="Allows the location property to be bound to
appropriate field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
Public ReadOnly Property LocationMapping As MappingInfo
```

### C#

```
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[System.ComponentModel.Browsable(true)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1CalendarStorageMappings.LocationMappin
g", Description="Allows the location property to be bound to appropriate
field in the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
```

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
public MappingInfo LocationMapping {get;}
```

## See Also

### Reference

[C1CalendarStorageMappings Class](#)

[C1CalendarStorageMappings Members](#)

## NameMapping Property

Allows the name property to be bound to appropriate field in the data source.

## Syntax

### Visual Basic (Declaration)

```
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorageMappings.NameM
apping", Description="Allows the name property to be bound to appropriate
field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property NameMapping As MappingInfo
```

### C#

```
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1CalendarStorageMappings.NameMapping",
Description="Allows the name property to be bound to appropriate field in the
data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
```

```
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089"')]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
public MappingInfo NameMapping {get;}
```

## See Also

### Reference

[C1CalendarStorageMappings Class](#)

[C1CalendarStorageMappings Members](#)

## PropertiesMapping Property

Allows other calendar properties to be bound to appropriate field in the data source.

## Syntax

### Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<EditorBrowsableAttribute()>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorageMappings.PropertiesMapping", Description="Allows other calendar properties to be bound to appropriate field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
Public ReadOnly Property PropertiesMapping As MappingInfo
```

### C#

```
[System.ComponentModel.Browsable(true)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[EditorBrowsable()]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1CalendarStorageMappings.PropertiesMapping", Description="Allows other calendar properties to be bound to appropriate field in the data source.")]
```

```
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
public MappingInfo PropertiesMapping {get;}
```

## See Also

### Reference

[C1CalendarStorageMappings Class](#)

[C1CalendarStorageMappings Members](#)

## TagMapping Property

Allows the tag property to be bound to appropriate field in the data source (the tag property can be used to store any user-defined information).

## Syntax

### Visual Basic (Declaration)

```
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1CalendarStorageMappings.TagMa
pping", Description="Allows the tag property to be bound to appropriate field
in the data source. (the tag property can be used to store any user-defined
information)")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public ReadOnly Property TagMapping As MappingInfo
```

### C#

```
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
```

```
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1CalendarStorageMappings.TagMapping",
Description="Allows the tag property to be bound to appropriate field in the
data source. (the tag property can be used to store any user-defined
information)")]
[System.ComponentModel.NotifyParentProperty(true)]
public MappingInfo TagMapping {get;}
```

## See Also

### Reference

[C1CalendarStorageMappings Class](#)

[C1CalendarStorageMappings Members](#)

## C1EventCollection

Represents C1Event object collection.

## Object Model

C1EventCollection

## Syntax

Visual Basic (Declaration)

```
Public Class C1EventCollection
    Inherits System.Collections.Generic.List(Of Event)
```

C#

```
public class C1EventCollection : System.Collections.Generic.List<Event>
```

## Inheritance Hierarchy

System.Object

System.Collections.Generic.List<T>

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventCollection**

## See Also

## Reference

[C1EventCollection Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Represents C1Event object collection.

## Object Model

**C1EventCollection**

## Syntax

Visual Basic (Declaration)

```
Public Class C1EventCollection
    Inherits System.Collections.Generic.List(Of Event)
```

C#

```
public class C1EventCollection : System.Collections.Generic.List<Event>
```

## Inheritance Hierarchy

System.Object

System.Collections.Generic.List<T>

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventCollection**

## See Also

### Reference


[C1EventCollection Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members




The following tables list the members exposed by [C1EventCollection](#).

### Public Constructors

	Name	Description
	<a href="#">C1EventCollection Constructor</a>	Initializes a new instance of the <a href="#">C1EventCollection</a> class.










[Top](#)

## Public Properties

















	Name	Description
	<a href="#">Capacity</a>	(Inherited from System.Collections.Generic.List<Event>)
	<a href="#">Count</a>	(Inherited from System.Collections.Generic.List<Event>)
	<a href="#">Item</a>	(Inherited from System.Collections.Generic.List<Event>)

[Top](#)

## Public Methods

	Name	Description
	<a href="#">AsReadOnly</a>	(Inherited from System.Collections.Generic.List<Event>)
	<a href="#">BinarySearch</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
	<a href="#">Clear</a>	This method is hidden. Please, use DataStorage methods(AddEvent/DeleteEvent/UpdateEvent) in order to modify collection.
	<a href="#">Contains</a>	(Inherited from System.Collections.Generic.List<Event>)
	<a href="#">ConvertAll&lt;TOutput&gt;</a>	(Inherited from System.Collections.Generic.List<Event>)
	<a href="#">CopyTo</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
	<a href="#">Exists</a>	(Inherited from System.Collections.Generic.List<Event>)
	<a href="#">Find</a>	(Inherited from System.Collections.Generic.List<Event>)
	<a href="#">FindAll</a>	(Inherited from System.Collections.Generic.List<Event>)



≡  <a href="#">FindIndex</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">FindLast</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">FindLastIndex</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">ForEach</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">GetRange</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">IndexOf</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">Insert</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">InsertRange</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">LastIndexOf</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">RemoveAll</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">RemoveAt</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">RemoveRange</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">Reverse</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">Sort</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">ToArray</a>	(Inherited from System.Collections.Generic.List<Event>)
≡  <a href="#">TrimExcess</a>	(Inherited from System.Collections.Generic.List<Event>)

	TrueForAll	(Inherited from System.Collections.Generic.List<Event>)
---	------------	---

[Top](#)

## See Also

### Reference

[C1EventCollection Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## C1EventCollection Constructor

Initializes a new instance of the [C1EventCollection](#) class.

## Syntax

Visual Basic (Declaration)	
<pre>Public Function New( _     ByVal storage As C1EventsCalendarStorage _ )</pre>	
C#	
<pre>public C1EventCollection(     C1EventsCalendarStorage storage )</pre>	

### Parameters

*storage*

The events calendar storage object.

## See Also

### Reference












[C1EventCollection Class](#)[C1EventCollection Members](#)

## Methods

For a list of all members of this type, see [C1EventCollection members](#).

## Public Methods

	Name	Description
≡	<a href="#">AsReadOnly</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">BinarySearch</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">Clear</a>	This method is hidden. Please, use DataStorage methods(AddEvent/DeleteEvent/UpdateEvent) in order to modify collection.
≡	<a href="#">Contains</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">ConvertAll&lt;TOutput&gt;</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">CopyTo</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">Exists</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">Find</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">FindAll</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">FindIndex</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">FindLast</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">FindLastIndex</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">ForEach</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">GetRange</a>	(Inherited from System.Collections.Generic.List<Event>)
≡	<a href="#">IndexOf</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)

 <a href="#">Insert</a>	(Inherited from System.Collections.Generic.List<Event>)
 <a href="#">InsertRange</a>	(Inherited from System.Collections.Generic.List<Event>)
 <a href="#">LastIndexOf</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
 <a href="#">RemoveAll</a>	(Inherited from System.Collections.Generic.List<Event>)
 <a href="#">RemoveAt</a>	(Inherited from System.Collections.Generic.List<Event>)
 <a href="#">RemoveRange</a>	(Inherited from System.Collections.Generic.List<Event>)
 <a href="#">Reverse</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
 <a href="#">Sort</a>	Overloaded. (Inherited from System.Collections.Generic.List<Event>)
 <a href="#">ToArray</a>	(Inherited from System.Collections.Generic.List<Event>)
 <a href="#">TrimExcess</a>	(Inherited from System.Collections.Generic.List<Event>)
 <a href="#">TrueForAll</a>	(Inherited from System.Collections.Generic.List<Event>)

[Top](#)

## See Also

### Reference

[C1EventCollection Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Clear Method

This method is hidden. Please, use `DataStorage` methods(`AddEvent/DeleteEvent/UpdateEvent`) in order to modify collection.

## Syntax

Visual Basic (Declaration)	
Public Shadows Sub Clear()	
C#	
public new void Clear()	

## See Also

### Reference

[C1EventCollection Class](#)

[C1EventCollection Members](#)

## C1EventsCalendar

The C1EventsCalendar control is a fully functional schedule that allows users to add, edit, and manage their appointments.

## Object Model

C1EventsCalendar

## Syntax

Visual Basic (Declaration)	
<pre>&lt;System.Web.UI.ToolboxDataAttribute("&lt;{0}:C1EventsCalendar runat="server"&gt;&lt;/{0}:C1EventsCalendar"&gt;")&gt; &lt;System.Drawing.ToolboxBitmapAttribute()&gt; &lt;System.Web.UI.ParseChildrenAttribute(True)&gt; &lt;System.ComponentModel.LicenseProviderAttribute()&gt; &lt;System.ComponentModel.DesignerAttribute(DesignerBaseTypeName="System.Compone ntModel.Design.IDesigner", DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1EventsCalendar.C1EventsCalen darDesigner, C1.Web.Wijmo.Controls.Design.4, Version=4.0.20151.207, Culture=neutral, PublicKeyToken=9b75583953471eea")&gt; &lt;WidgetDependenciesAttribute(System.Object[])&gt; Public Class C1EventsCalendar     Inherits <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a></pre>	
C#	
<pre>[System.Web.UI.ToolboxData("&lt;{0}:C1EventsCalendar</pre>	

```

runat="server"></{0}:C1EventsCalendar>")]]
[System.Drawing.ToolboxBitmap()]
[System.Web.UI.ParseChildren(true)]
[System.ComponentModel.LicenseProvider()]
[System.ComponentModel.Designer(DesignerBaseTypeName="System.ComponentModel.D
esign.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1EventsCalendar.C1EventsCalen
darDesigner, C1.Web.Wijmo.Controls.Design.4, Version=4.0.20151.207,
Culture=neutral, PublicKeyToken=9b75583953471eea")]
[WidgetDependencies(System.Object[])]
public class C1EventsCalendar : C1.Web.Wijmo.Controls.C1TargetControlBase

```

## Inheritance Hierarchy

```

System.Object
  System.Web.UI.Control
    System.Web.UI.WebControls.WebControl
      C1.Web.Wijmo.Controls.C1TargetControlBase
        C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendar

```

## See Also

### Reference

[C1EventsCalendar Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

The C1EventsCalendar control is a fully functional schedule that allows users to add, edit, and manage their appointments.

## Object Model

C1EventsCalendar

## Syntax

Visual Basic (Declaration)

```

<System.Web.UI.ToolboxDataAttribute("<{0}:C1EventsCalendar
runat="server"></{0}:C1EventsCalendar>")>
<System.Drawing.ToolboxBitmapAttribute()>
<System.Web.UI.ParseChildrenAttribute(True)>
<System.ComponentModel.LicenseProviderAttribute()>

```

```
<System.ComponentModel.DesignerAttribute(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1EventsCalendar.C1EventsCalendarDesigner, C1.Web.Wijmo.Controls.Design.4, Version=4.0.20151.207,
Culture=neutral, PublicKeyToken=9b75583953471eea">
<WidgetDependenciesAttribute(System.Object[])>
Public Class C1EventsCalendar
    Inherits C1.Web.Wijmo.Controls.C1TargetControlBase
```

C#

```
[System.Web.UI.ToolboxData("<{0}:C1EventsCalendar
runat='server'></{0}:C1EventsCalendar>")]
[System.Drawing.ToolboxBitmap()]
[System.Web.UI.ParseChildren(true)]
[System.ComponentModel.LicenseProvider()]
[System.ComponentModel.Designer(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1EventsCalendar.C1EventsCalendarDesigner, C1.Web.Wijmo.Controls.Design.4, Version=4.0.20151.207,
Culture=neutral, PublicKeyToken=9b75583953471eea")]
[WidgetDependencies(System.Object[])]
public class C1EventsCalendar : C1.Web.Wijmo.Controls.C1TargetControlBase
```

## Inheritance Hierarchy

```
System.Object
    System.Web.UI.Control
        System.Web.UI.WebControls.WebControl
            C1.Web.Wijmo.Controls.C1TargetControlBase
                C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendar
```

## See Also

### Reference


[C1EventsCalendar Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members











The following tables list the members exposed by [C1EventsCalendar](#).

## Public Constructors












	Name	Description
	<a href="#">C1EventsCalendar Constructor</a>	Initializes a new instance of the <b>C1.Web.Wijmo.Controls.C1ReportViewer</b> class.














[Top](#)








## Public Properties













	Name	Description
	<a href="#">AccessKey</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">AppRelativeTemplateSourceDirectory</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Attributes</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">CDNDependencyPaths</a>	A string array value specifies all css/js references that wijmo control depends on. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">CDNPath</a>	Content Delivery Network path. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">ClientID</a>	(Inherited from System.Web.UI.Control)
	<a href="#">ClientIDMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Colors</a>	The colors property specifies the list of color names which will be shown in the color name drop down list.
	<a href="#">Controls</a>	(Inherited from System.Web.UI.Control)
	<a href="#">ControlStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)

























	<a href="#">ControlStyleCreated</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">CssClass</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">Culture</a>	CultureInfo object. Each culture has different conventions for displaying dates, time, numbers, and other information. Neutrals cultures are not supported.
	<a href="#">CultureCalendar</a>	A value that indicators the culture calendar to format the text. This property must work with Culture property.
	<a href="#">DataStorage</a>	C1EventsCalendar storage configuration.
	<a href="#">DatePagerLocalization</a>	Use the Localization property in order to customize localization strings.
	<a href="#">DayHeaderFormat</a>	Format of the text for the day cell header in the month view. Format argument: {0} = Day date.
	<a href="#">DayViewHeaderFormat</a>	Format of the text for the day header in the day/week or list view. Format argument: {0} = Day date.
	<a href="#">DisplayVisible</a>	Gets or sets a value that indicates whether a server control is rendered as UI on the page. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">EnableCombinedJavaScripts</a>	Enable JavaScripts files combined into one file. In order to combined enabled, you must register the WijmoHttpHandler in web.config. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">Enabled</a>	Gets or sets a value indicating whether the Web server control is enabled. (Inherited from


		<a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">EnableLogs</a>	Enables built-in log console.
	<a href="#">EnableTheming</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">EnableViewState</a>	(Inherited from System.Web.UI.Control)
	<a href="#">EnsureEventDialogOnBody</a>	Indicating the event dialog element will be append to the body or eventcalendar container.
	<a href="#">EventDialogTemplate</a>	Gets or sets the Event Dialog template used to display the content of the Event Dialog.
	<a href="#">EventTemplate</a>	Gets or sets the Event template used to display the content of the Event.
	<a href="#">EventTitleFormat</a>	Format of the title text for the event. Format arguments: {0} = Start, {1} = End, {2} = Subject, {3} = Location, {4} = Icons.
	<a href="#">FirstRowDayHeaderFormat</a>	Format of the text for the first cell header in the first row of the month view. Format argument: {0} = Day date.
	<a href="#">HasAttributes</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">HeaderBarVisible</a>	Indicates whether the header bar will be visible.
	<a href="#">Height</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">ID</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Localization</a>	Use the Localization property in order to customize

		localization strings.
	<a href="#">NamingContainer</a>	(Inherited from System.Web.UI.Control)
	<a href="#">NavigationBarVisible</a>	Indicates whether the bottom navigation bar will be visible.
	<a href="#">OnClientBeforeAddEvent</a>	Occurs before the add event action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new event data that should be added to a data source.
	<a href="#">OnClientBeforeDeleteCalendar</a>	Occurs before the delete calendar action. Return false or call event.preventDefault() in order to cancel event and prevent the delete action. args.data - the calendar data that should be deleted from a data source.
	<a href="#">OnClientBeforeDeleteEvent</a>	Occurs before the delete action. Return false or call event.preventDefault() in order to cancel event and prevent the delete action. args.data - the event object that should be deleted.
	<a href="#">OnClientBeforeEditEventDialogShow</a>	Occurs before the built-in event dialog is shown. Return false or call event.preventDefault() in order to cancel event and prevent the built-in dialog to be shown. args.data - the event data. args.targetCell - target offset DOM element which can be used for popup callout.
	<a href="#">OnClientBeforeSaveCalendar</a>	Occurs before the save calendar action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new calendar data that should be saved to a data source. args.prevData - previous calendar data, this argument is empty for a new calendar.

	<a href="#">OnClientBeforeUpdateEvent</a>	Occurs before the update event action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new event data that should be updated. args.prevData - previous event data, this argument is empty for a new event.
	<a href="#">OnClientCalendarsChanged</a>	Occurs when calendars option has been changed. args.calendars - the new calendars option value.
	<a href="#">OnClientEventCreating</a>	Occurs when event markup is creating. args.data - the event data. args.eventTemplate The template of events displayed on the view.
	<a href="#">OnClientInitialized</a>	Occurs when events calendar is constructed and events data is loaded from an external or local data source.
	<a href="#">OnClientSelectedDatesChanged</a>	Occurs when selectedDates option has been changed. args.selectedDates - the new selectedDates value.
	<a href="#">OnClientViewTypeChanged</a>	Occurs when viewType option has been changed.
	<a href="#">Page</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Parent</a>	(Inherited from System.Web.UI.Control)
	<a href="#">RenderingCompatibility</a>	(Inherited from System.Web.UI.Control)
	<a href="#">SelectedDate</a>	The selected date.
	<a href="#">SelectedDates</a>	Collection of DateTime objects that represent the selected dates on the C1EventsCalendar control for the DayView.
	<a href="#">Site</a>	(Inherited from System.Web.UI.Control)












	<a href="#">SkinID</a>	(Inherited from <code>System.Web.UI.WebControls.WebControl</code> )
	<a href="#">Style</a>	(Inherited from <code>System.Web.UI.WebControls.WebControl</code> )
	<a href="#">SupportsDisabledAttribute</a>	(Inherited from <code>System.Web.UI.WebControls.WebControl</code> )
	<a href="#">TabIndex</a>	(Inherited from <code>System.Web.UI.WebControls.WebControl</code> )
	<a href="#">TemplateControl</a>	(Inherited from <code>System.Web.UI.Control</code> )
	<a href="#">TemplateSourceDirectory</a>	(Inherited from <code>System.Web.UI.Control</code> )
	<a href="#">Theme</a>	Name of the theme that will be used to style the widgets. Available embedded themes include: aristo, midnight, and ui-lightness. Note that only one theme can be used for the whole page at one time. Please make sure that all widget extenders have the same Theme value. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">ThemeSwatch</a>	A value that indicates the theme swatch of the control, this property only works when <code>WijmoControlMode</code> property is Mobile. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">TimeInterval</a>	Time interval for the Day view (in minutes).
	<a href="#">TimeIntervalHeight</a>	The Day view time interval row height in pixels.
	<a href="#">TimeRulerFormat</a>	Time ruler format for the Day view. Format argument: {0} = Current ruler time.
	<a href="#">TimeRulerInterval</a>	Time ruler interval for the Day view (in minutes).

	<a href="#">ToolTip</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">UniqueID</a>	(Inherited from System.Web.UI.Control)
	<a href="#">UseCDN</a>	Determines whether the widget extender must load client resources from CDN (Content Delivery Network) path given by property CDNPath. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">ValidateRequestMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Views</a>	List of the views which need to be shown. Both default views("day", "week", "month", "list") and custom views can be added. If add a default view, only Type property of View should be set. If add a custom view, All the properties of View should be set.
	<a href="#">ViewStateMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">VisibleCalendars</a>	Array of the calendar names which need to be shown.
	<a href="#">WebServiceUrl</a>	The URL to the web service which will be used to store information about events.
	<a href="#">Width</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">WijmoControlMode</a>	A value that indicates mode of the control, whether it is a mobile or web control. Note that only one value can be used for the whole website or project. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )

	<a href="#">WijmoCssAdapter</a>	Indicates the control applies the theme of JQuery UI or Bootstrap. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
---	---------------------------------	--

[Top](#)

## Public Methods







	Name	Description
	<a href="#">ApplyStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">ApplyStyleSheetSkin</a>	(Inherited from System.Web.UI.Control)
	<a href="#">CopyBaseAttributes</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">DataBind</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Dispose</a>	(Inherited from System.Web.UI.Control)
	<a href="#">FindControl</a>	Overridden. Searches the current naming container for a server control with the specified <a href="#">id</a> parameter.
	<a href="#">Focus</a>	(Inherited from System.Web.UI.Control)
	<a href="#">GetPropertyValue&lt;V&gt;</a>	Gets the property value by property name. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">GetRouteUrl</a>	Overloaded. (Inherited from System.Web.UI.Control)
	<a href="#">GetScriptDescriptors</a>	When overridden in a derived class, registers the WidgetDescriptor objects for the control. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">GetScriptReferences</a>	When overridden in a derived class, registers an additional script libraries for the extender control. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )

≡◆	<a href="#">GetUniqueIDRelativeTo</a>	(Inherited from System.Web.UI.Control)
≡◆	<a href="#">HasControls</a>	(Inherited from System.Web.UI.Control)
≡◆	<a href="#">LoadLayout</a>	Overloaded. Loads control layout properties from the file.
≡◆	<a href="#">LoadPostData</a>	When implemented by a class, processes postback data for an ASP.NET server control.
≡◆	<a href="#">MergeStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
≡◆	<a href="#">RaisePostBackEvent</a>	Process an event raised when a form is posted to the server.
≡◆	<a href="#">RaisePostDataChangedEvent</a>	When implemented by a class, signals the server control to notify the ASP.NET application that the state of the control has changed.
≡◆	<a href="#">RegisterDesignTimeStyleSheet</a>	Register design time stylesheet (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
≡◆	<a href="#">RenderBeginTag</a>	(Inherited from System.Web.UI.WebControls.WebControl)
≡◆	<a href="#">RenderControl</a>	(Inherited from System.Web.UI.Control)
≡◆	<a href="#">RenderEndTag</a>	(Inherited from System.Web.UI.WebControls.WebControl)
≡◆	<a href="#">ResolveClientUrl</a>	(Inherited from System.Web.UI.Control)
≡◆	<a href="#">ResolveUrl</a>	(Inherited from System.Web.UI.Control)
≡◆	<a href="#">SaveLayout</a>	Overloaded. Saves the control layout properties to the file.
≡◆	<a href="#">SetPropertyValue&lt;V&gt;</a>	Sets the property value by property name. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
≡◆	<a href="#">SetRenderMethodDelegate</a>	(Inherited from System.Web.UI.Control)

[Top](#)



## Public Events

	Name	Description
	<a href="#">DataBinding</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Disposed</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Init</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Load</a>	(Inherited from System.Web.UI.Control)
	<a href="#">PreRender</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Unload</a>	(Inherited from System.Web.UI.Control)

[Top](#)

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## C1EventsCalendar Constructor

Initializes a new instance of the **C1.Web.Wijmo.Controls.C1ReportViewer** class.

## Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1EventsCalendar()	

## See Also

### Reference

[C1EventsCalendar Class](#)















[C1EventsCalendar Members](#)

## Methods

For a list of all members of this type, see [C1EventsCalendar members](#).

### Public Methods

	Name	Description
≡	<a href="#">ApplyStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
≡	<a href="#">ApplyStyleSheetSkin</a>	(Inherited from System.Web.UI.Control)
≡	<a href="#">CopyBaseAttributes</a>	(Inherited from System.Web.UI.WebControls.WebControl)
≡	<a href="#">DataBind</a>	(Inherited from System.Web.UI.Control)
≡	<a href="#">Dispose</a>	(Inherited from System.Web.UI.Control)
≡	<a href="#">FindControl</a>	Overridden. Searches the current naming container for a server control with the specified <a href="#">id</a> parameter.
≡	<a href="#">Focus</a>	(Inherited from System.Web.UI.Control)
≡	<a href="#">GetPropertyValue&lt;V&gt;</a>	Gets the property value by property name. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
≡	<a href="#">GetRouteUrl</a>	Overloaded. (Inherited from System.Web.UI.Control)
≡	<a href="#">GetScriptDescriptors</a>	When overridden in a derived class, registers the WidgetDescriptor objects for the control. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
≡	<a href="#">GetScriptReferences</a>	When overridden in a derived class, registers an additional script libraries for the extender control. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
≡	<a href="#">GetUniqueIDRelativeTo</a>	(Inherited from System.Web.UI.Control)
≡	<a href="#">HasControls</a>	(Inherited from System.Web.UI.Control)

 <a href="#">LoadLayout</a>	Overloaded. Loads control layout properties from the file.
 <a href="#">LoadPostData</a>	When implemented by a class, processes postback data for an ASP.NET server control.
 <a href="#">MergeStyle</a>	(Inherited from System.Web.UI.WebControls.WebControl)
 <a href="#">RaisePostBackEvent</a>	Process an event raised when a form is posted to the server.
 <a href="#">RaisePostDataChangedEvent</a>	When implemented by a class, signals the server control to notify the ASP.NET application that the state of the control has changed.
 <a href="#">RegisterDesignTimeStyleSheet</a>	Register design time stylesheet (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
 <a href="#">RenderBeginTag</a>	(Inherited from System.Web.UI.WebControls.WebControl)
 <a href="#">RenderControl</a>	(Inherited from System.Web.UI.Control)
 <a href="#">RenderEndTag</a>	(Inherited from System.Web.UI.WebControls.WebControl)
 <a href="#">ResolveClientUrl</a>	(Inherited from System.Web.UI.Control)
 <a href="#">ResolveUrl</a>	(Inherited from System.Web.UI.Control)
 <a href="#">SaveLayout</a>	Overloaded. Saves the control layout properties to the file.
 <a href="#">SetPropertyValue&lt;V&gt;</a>	Sets the property value by property name. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
 <a href="#">SetRenderMethodDelegate</a>	(Inherited from System.Web.UI.Control)

[Top](#)

## See Also

## Reference

[C1EventsCalendar Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## FindControl Method

Searches the current naming container for a server control with the specified [id](#) parameter.

### Syntax

Visual Basic (Declaration)	
<pre>Public Overrides Function FindControl( _     ByVal id As System.String _ ) As System.Web.UI.Control</pre>	
C#	
<pre>public override System.Web.UI.Control FindControl(     System.string id )</pre>	

### Parameters

*id*

The identifier for the control to be found.

### Return Value

The specified control, or null if the specified control does not exist.

## See Also

### Reference

[C1EventsCalendar Class](#)[C1EventsCalendar Members](#)

## LoadLayout Method

Loads control layout properties from the file.

### Overload List

Overload	Description
<a href="#">LoadLayout(String)</a>	Loads control layout properties from the file.

<a href="#">LoadLayout(Stream)</a>	Load control layout properties from the stream.
<a href="#">LoadLayout(String,LayoutType)</a>	Loads control layout properties with specified types from the file.
<a href="#">LoadLayout(Stream,LayoutType)</a>	Loads the control layout properties with specified types from the stream.

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## LoadLayout(String) Method

Loads control layout properties from the file.

## Syntax

Visual Basic (Declaration)

```
Public Overloads Sub LoadLayout( _  
    ByVal filename As System.String _  
)
```

C#

```
public void LoadLayout(  
    System.string filename  
)
```

### Parameters

*filename*

The file where the values of layout properties will be loaded.

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

[Overload List](#)

## LoadLayout(Stream) Method

Load control layout properties from the stream.

### Syntax

Visual Basic (Declaration)

```
Public Overloads Sub LoadLayout( _  
    ByVal stream As System.IO.Stream _  
)
```

C#

```
public void LoadLayout(  
    System.IO.Stream stream  
)
```

### Parameters

*stream*

The stream where the values of layout properties will be loaded.

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

[Overload List](#)

## LoadLayout(String,LayoutType) Method

The file where the values of layout properties will be loaded.

The layout types to load.

Loads control layout properties with specified types from the file.

### Syntax

Visual Basic (Declaration)

```
Public Overloads Sub LoadLayout( _  
    ByVal filename As System.String, _  
    ByVal layoutTypes As LayoutType _  
)
```

C#

```
public void LoadLayout(  
    System.string filename,  
    LayoutType layoutTypes  
)
```

### Parameters

*filename*

The file where the values of layout properties will be loaded.

*layoutTypes*

The layout types to load.

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

[Overload List](#)

## LoadLayout(Stream,LayoutType) Method

The stream where the values of the layout properties will be loaded.

The layout types to load.

Loads the control layout properties with specified types from the stream.

## Syntax

Visual Basic (Declaration)

```
Public Overloads Sub LoadLayout( _  
    ByVal stream As System.IO.Stream, _  
    ByVal layoutTypes As LayoutType _  
)
```

C#

```
public void LoadLayout(  
    System.IO.Stream stream,  
    LayoutType layoutTypes  
)
```

## Parameters

*stream*

The stream where the values of the layout properties will be loaded.

*layoutTypes*

The layout types to load.

## See Also

## Reference

[C1EventsCalendar Class](#)  
[C1EventsCalendar Members](#)  
[Overload List](#)

## LoadPostData Method

The key identifier for the control.

The collection of all incoming name values.

When implemented by a class, processes postback data for an ASP.NET server control.

## Syntax

Visual Basic (Declaration)

```
Public Function LoadPostData( _  
    ByVal postDataKey As System.String, _  
    ByVal postCollection As System.Collections.Specialized.NameValueCollection  
_  
) As System.Boolean
```

C#

```
public System.bool LoadPostData(  
    System.string postDataKey,  
    System.Collections.Specialized.NameValueCollection postCollection  
)
```

## Parameters

*postDataKey*

The key identifier for the control.

*postCollection*



The collection of all incoming name values.

### Return Value

true if the server control's state changes as a result of the postback; otherwise, false.

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## RaisePostBackEvent Method

A System.String that represents an optional event argument to be passed to the event handler.

Process an event raised when a form is posted to the server.

## Syntax

Visual Basic (Declaration)	
<pre>Public Sub RaisePostBackEvent( _     ByVal <i>eventArgument</i> As System.String _ )</pre>	
C#	
<pre>public void RaisePostBackEvent(     System.string <i>eventArgument</i> )</pre>	

### Parameters

*eventArgument*

A System.String that represents an optional event argument to be passed to the event handler.

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## RaisePostDataChangedEvent Method

When implemented by a class, signals the server control to notify the ASP.NET application that the state of the control has changed.

### Syntax

Visual Basic (Declaration)	
<pre>Public Sub RaisePostDataChangedEvent()</pre>	
C#	
<pre>public void RaisePostDataChangedEvent()</pre>	

### See Also

#### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## SaveLayout Method

Saves the control layout properties to the file.

### Overload List

Overload	Description
<a href="#">SaveLayout(String)</a>	Saves the control layout properties to the file.
<a href="#">SaveLayout(Stream)</a>	Saves control layout properties to the stream.

### See Also

#### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## SaveLayout(String) Method

Saves the control layout properties to the file.

### Syntax

## Visual Basic (Declaration)

```
Public Overloads Sub SaveLayout( _  
    ByVal filename As System.String _  
)
```

## C#

```
public void SaveLayout(  
    System.string filename  
)
```

**Parameters**

*filename*

The file where the values of the layout properties will be saved.

## See Also

**Reference**

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

[Overload List](#)

## SaveLayout(Stream) Method

Saves control layout properties to the stream.

## Syntax

## Visual Basic (Declaration)

```
Public Overloads Sub SaveLayout( _  
    ByVal stream As System.IO.Stream _  
)
```

## C#

```
public void SaveLayout(  
    System.IO.Stream stream  
)
```

**Parameters**

*stream*

The stream where the values of layout properties will be saved.

## See Also










### Reference












[C1EventsCalendar Class](#)  
[C1EventsCalendar Members](#)  
[Overload List](#)













## Properties










For a list of all members of this type, see [C1EventsCalendar members](#).












### Public Properties

	Name	Description
	<a href="#">AccessKey</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">AppRelativeTemplateSourceDirectory</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )
	<a href="#">Attributes</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">CDNDependencyPaths</a>	A string array value specifies all css/js references that wijmo control depends on. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">CDNPath</a>	Content Delivery Network path. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">ClientID</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )
	<a href="#">ClientIDMode</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )
	<a href="#">Colors</a>	The colors property specifies the list of color names which will be shown in the color name drop down list.
	<a href="#">Controls</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )












	<a href="#">ControlStyle</a>	(Inherited from <code>System.Web.UI.WebControls.WebControl</code> )
	<a href="#">ControlStyleCreated</a>	(Inherited from <code>System.Web.UI.WebControls.WebControl</code> )
	<a href="#">CssClass</a>	(Inherited from <code>System.Web.UI.WebControls.WebControl</code> )
	<a href="#">Culture</a>	CultureInfo object. Each culture has different conventions for displaying dates, time, numbers, and other information. Neutrals cultures are not supported.
	<a href="#">CultureCalendar</a>	A value that indicators the culture calendar to format the text. This property must work with Culture property.
	<a href="#">DataStorage</a>	C1EventsCalendar storage configuration.
	<a href="#">DatePagerLocalization</a>	Use the Localization property in order to customize localization strings.
	<a href="#">DayHeaderFormat</a>	Format of the text for the day cell header in the month view. Format argument: {0} = Day date.
	<a href="#">DayViewHeaderFormat</a>	Format of the text for the day header in the day/week or list view. Format argument: {0} = Day date.
	<a href="#">DisplayVisible</a>	Gets or sets a value that indicates whether a server control is rendered as UI on the page. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">EnableCombinedJavaScripts</a>	Enable JavaScripts files combined into one file. In order to combined enabled, you must register the WijmoHttpHandler in web.config. (Inherited from













		<a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">Enabled</a>	Gets or sets a value indicating whether the Web server control is enabled. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">EnableLogs</a>	Enables built-in log console.
	<a href="#">EnableTheming</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">EnableViewState</a>	(Inherited from <a href="#">System.Web.UI.Control</a> )
	<a href="#">EnsureEventDialogOnBody</a>	Indicating the event dialog element will be append to the body or eventcalendar container.
	<a href="#">EventDialogTemplate</a>	Gets or sets the Event Dialog template used to display the content of the Event Dialog.
	<a href="#">EventTemplate</a>	Gets or sets the Event template used to display the content of the Event.
	<a href="#">EventTitleFormat</a>	Format of the title text for the event. Format arguments: {0} = Start, {1} = End, {2} = Subject, {3} = Location, {4} = Icons.
	<a href="#">FirstRowDayHeaderFormat</a>	Format of the text for the first cell header in the first row of the month view. Format argument: {0} = Day date.
	<a href="#">HasAttributes</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )
	<a href="#">HeaderBarVisible</a>	Indicates whether the header bar will be visible.
	<a href="#">Height</a>	(Inherited from <a href="#">System.Web.UI.WebControls.WebControl</a> )


	ID	(Inherited from System.Web.UI.Control)
	Localization	Use the Localization property in order to customize localization strings.
	NamingContainer	(Inherited from System.Web.UI.Control)
	NavigationBarVisible	Indicates whether the bottom navigation bar will be visible.
	OnClientBeforeAddEvent	Occurs before the add event action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new event data that should be added to a data source.
	OnClientBeforeDeleteCalendar	Occurs before the delete calendar action. Return false or call event.preventDefault() in order to cancel event and prevent the delete action. args.data - the calendar data that should be deleted from a data source.
	OnClientBeforeDeleteEvent	Occurs before the delete action. Return false or call event.preventDefault() in order to cancel event and prevent the delete action. args.data - the event object that should be deleted.
	OnClientBeforeEditEventDialogShow	Occurs before the built-in event dialog is shown. Return false or call event.preventDefault() in order to cancel event and prevent the built-in dialog to be shown. args.data - the event data. args.targetCell - target offset DOM element which can be used for popup callout.
	OnClientBeforeSaveCalendar	Occurs before the save calendar action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new calendar data that should be saved to a data

		source. args.prevData - previous calendar data, this argument is empty for a new calendar.
	<a href="#">OnClientBeforeUpdateEvent</a>	Occurs before the update event action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new event data that should be updated. args.prevData - previous event data, this argument is empty for a new event.
	<a href="#">OnClientCalendarsChanged</a>	Occurs when calendars option has been changed. args.calendars - the new calendars option value.
	<a href="#">OnClientEventCreating</a>	Occurs when event markup is creating. args.data - the event data. args.eventTemplate The template of events displayed on the view.
	<a href="#">OnClientInitialized</a>	Occurs when events calendar is constructed and events data is loaded from an external or local data source.
	<a href="#">OnClientSelectedDatesChanged</a>	Occurs when selectedDates option has been changed. args.selectedDates - the new selectedDates value.
	<a href="#">OnClientViewTypeChanged</a>	Occurs when viewType option has been changed.
	<a href="#">Page</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Parent</a>	(Inherited from System.Web.UI.Control)
	<a href="#">RenderingCompatibility</a>	(Inherited from System.Web.UI.Control)
	<a href="#">SelectedDate</a>	The selected date.
	<a href="#">SelectedDates</a>	Collection of DateTime objects that represent the selected dates on the C1EventsCalendar control for



		the DayView.
	<a href="#">Site</a>	(Inherited from System.Web.UI.Control)
	<a href="#">SkinID</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">Style</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">SupportsDisabledAttribute</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">TabIndex</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">TemplateControl</a>	(Inherited from System.Web.UI.Control)
	<a href="#">TemplateSourceDirectory</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Theme</a>	Name of the theme that will be used to style the widgets. Available embedded themes include: aristo, midnight, and ui-lightness. Note that only one theme can be used for the whole page at one time. Please make sure that all widget extenders have the same Theme value. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">ThemeSwatch</a>	A value that indicates the theme swatch of the control, this property only works when WijmoControlMode property is Mobile. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">TimeInterval</a>	Time interval for the Day view (in minutes).
	<a href="#">TimeIntervalHeight</a>	The Day view time interval row height in pixels.

	<a href="#">TimeRulerFormat</a>	Time ruler format for the Day view. Format argument: {0} = Current ruler time.
	<a href="#">TimeRulerInterval</a>	Time ruler interval for the Day view (in minutes).
	<a href="#">ToolTip</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">UniqueID</a>	(Inherited from System.Web.UI.Control)
	<a href="#">UseCDN</a>	Determines whether the widget extender must load client resources from CDN (Content Delivery Network) path given by property CDNPath. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">ValidateRequestMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">Views</a>	List of the views which need to be shown. Both default views("day", "week", "month", "list") and custom views can be added. If add a default view, only Type property of View should be set. If add a custom view, All the properties of View should be set.
	<a href="#">ViewStateMode</a>	(Inherited from System.Web.UI.Control)
	<a href="#">VisibleCalendars</a>	Array of the calendar names which need to be shown.
	<a href="#">WebServiceUrl</a>	The URL to the web service which will be used to store information about events.
	<a href="#">Width</a>	(Inherited from System.Web.UI.WebControls.WebControl)
	<a href="#">WijmoControlMode</a>	A value that indicates mode of the control, whether it is a mobile or web control. Note that only one

		value can be used for the whole website or project. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )
	<a href="#">WijmoCssAdapter</a>	Indicates the control applies the theme of JQuery UI or Bootstrap. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1TargetControlBase</a> )

[Top](#)

## See Also

### Reference

[C1EventsCalendar Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Colors Property

The colors property specifies the list of color names which will be shown in the color name drop down list.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.Colors", Description="The
Colors property specifies the list of color name which will be shown in the
color name drop down list in the event dialog.")>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<TypeConverterAttribute("System.Web.UI.WebControls.StringArrayConverter,
System.Web, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property Colors As System.String()
```

C#

```
[C1Description(Key="C1EventsCalendar.Colors", Description="The Colors
property specifies the list of color name which will be shown in the color
name drop down list in the event dialog.")]
[C1Category("Appearance")]
```

```
[System.ComponentModel.DefaultValue()]
[TypeConverter("System.Web.UI.WebControls.StringArrayConverter, System.Web,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string[] Colors {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## Culture Property

CultureInfo object. Each culture has different conventions for displaying dates, time, numbers, and other information. Neutrals cultures are not supported.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.Culture",
Description="CultureInfo object. Each culture has different conventions for
displaying dates, time, numbers, and other information. Neutrals cultures are
not supported.")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<EditorAttribute(EditorBaseTypeName="System.ComponentModel.CultureInfoConvert
er, System, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089",
EditorTypeName="System.Drawing.Design.UITypeEditor, System.Drawing,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a")>
Public Property Culture As System.Globalization.CultureInfo
```

C#

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.Culture", Description="CultureInfo
```

```

object. Each culture has different conventions for displaying dates, time,
numbers, and other information. Neutrals cultures are not supported.")]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[Editor(EditorBaseTypeName="System.ComponentModel.CultureInfoConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089",
EditorTypeName="System.Drawing.Design.UITypeEditor, System.Drawing,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a")]
public System.Globalization.CultureInfo Culture {get; set;}

```

## Remarks

returning value can be null if Culture is not defined

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## CultureCalendar Property

A value that indicators the culture calendar to format the text. This property must work with Culture property.

## Syntax

Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="Culture.CultureCalendar", Description="A value
that indicators the culture calendar to format the text. This property must
work with Culture property.")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
Public Property CultureCalendar As System.String

```

C#

```

[C1Category("Appearance")]
[C1Description(Key="Culture.CultureCalendar", Description="A value that
indicators the culture calendar to format the text. This property must work

```

```
with Culture property."))]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
public System.string CultureCalendar {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## DataStorage Property

C1EventsCalendar storage configuration.

## Syntax

### Visual Basic (Declaration)

```
<EditorBrowsableAttribute(>
<C1CategoryAttribute("Data")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DataStorage",
Description="C1EventsCalendar storage configuration.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<System.ComponentModel.BrowsableAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
Public ReadOnly Property DataStorage As C1EventsCalendarStorage
```

### C#

```
[EditorBrowsable()]
[C1Category("Data")]
[C1Description(Key="C1EventsCalendar.DataStorage",
Description="C1EventsCalendar storage configuration.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[System.ComponentModel.Browsable(true)]
[System.ComponentModel.Bindable(Bindable=false,
```

```
Direction=BindingDirection.OneWay)]
public C1EventsCalendarStorage DataStorage {get;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## DatePagerLocalization Property

Use the Localization property in order to customize localization strings.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalization",
Description="Use the Localization property in order to customize localization
strings.")>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<WidgetOptionAttribute()>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Content)>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
Public ReadOnly Property DatePagerLocalization As DatePagerLocalizationOption
```

C#

```
[C1Description(Key="C1EventsCalendar.DatePagerLocalization", Description="Use
the Localization property in order to customize localization strings.")]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[WidgetOption()]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
```

```
public DatePagerLocalizationOption DatePagerLocalization {get;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## DayHeaderFormat Property

Format of the text for the day cell header in the month view. Format argument: {0} = Day date.

## Syntax

### Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DayHeaderFormat",
Description="Format of the text for the day cell header in the month view.
Format argument: {0} = Day date.")>
<TypeConverterAttribute("C1.Web.Wijmo.Controls.C1EventsCalendar.DayHeaderForm
atConverter, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property DayHeaderFormat As System.String
```

### C#

```
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DayHeaderFormat", Description="Format of
the text for the day cell header in the month view. Format argument: {0} =
Day date.")]
[TypeConverter("C1.Web.Wijmo.Controls.C1EventsCalendar.DayHeaderFormatConvert
er, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[Layout(LayoutType.Appearance)]
public System.string DayHeaderFormat {get; set;}
```

## See Also

### Reference



[C1EventsCalendar Class](#)[C1EventsCalendar Members](#)

## DayViewHeaderFormat Property

Format of the text for the day header in the day/week or list view. Format argument: {0} = Day date.

### Syntax

Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DayViewHeaderFormat",
Description="Format of the text for the day header in the day/week or list
view. Format argument: {0} = Day date.")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Content)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<LayoutAttribute(LayoutType.Appearance)>
Public ReadOnly Property DayViewHeaderFormat As DayViewHeaderFormatOption

```

C#

```

[C1Category("Appearance")]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.DayViewHeaderFormat",
Description="Format of the text for the day header in the day/week or list
view. Format argument: {0} = Day date.")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
[System.ComponentModel.NotifyParentProperty(true)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[Layout(LayoutType.Appearance)]
public DayViewHeaderFormatOption DayViewHeaderFormat {get;}

```

### See Also

### Reference

[C1EventsCalendar Class](#)[C1EventsCalendar Members](#)

## EnableLogs Property

Enables built-in log console.

### Syntax

Visual Basic (Declaration)

```

<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1EventsCalendar.EnableLogs",
Description="Enables built-in log console.")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
Public Property EnableLogs As System.Boolean

```

C#

```

[System.ComponentModel.DefaultValue()]
[C1Category("Behavior")]
[C1Description(Key="C1EventsCalendar.EnableLogs", Description="Enables built-
in log console.")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
public System.bool EnableLogs {get; set;}

```

### See Also

#### Reference

[C1EventsCalendar Class](#)[C1EventsCalendar Members](#)

## EnsureEventDialogOnBody Property

Indicating the event dialog element will be append to the body or eventcalendar container.

### Syntax

Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1EventsCalendar.EnsureEventDialogOnBody",
Description="Indicating the event dialog element will be append to the body

```

```

or eventcalendar container.">
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Behavior")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property EnsureEventDialogOnBody As System.Boolean

```

C#

```

[C1Description(Key="C1EventsCalendar.EnsureEventDialogOnBody",
Description="Indicating the event dialog element will be append to the body
or eventcalendar container.")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Behavior")]
[Layout(LayoutType.Behavior)]
public System.bool EnsureEventDialogOnBody {get; set;}

```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## EventDialogTemplate Property

Gets or sets the Event Dialog template used to display the content of the Event Dialog.

## Syntax

Visual Basic (Declaration)

```

<System.ComponentModel.BrowsableAttribute(False)>
<JsonAttribute(IsSerializableToJson=False, IsSkipIfEmpty=False)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1Dialog.EventDialogTemplate",
Description="C1Dialog.EventDialogTemplate")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Content)>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<System.Web.UI.TemplateInstanceAttribute(TemplateInstance.Single)>
<LayoutAttribute(LayoutType.Behavior)>
Public Property EventDialogTemplate As System.Web.UI.ITemplate

```

C#

```
[System.ComponentModel.Browsable(false)]
[Json(IsSerializableToJson=false, IsSkipIfEmpty=false)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1Dialog.EventDialogTemplate",
Description="C1Dialog.EventDialogTemplate")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[System.Web.UI.TemplateInstance(TemplateInstance.Single)]
[Layout(LayoutType.Behavior)]
public System.Web.UI.ITemplate EventDialogTemplate {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## EventTemplate Property

Gets or sets the Event template used to display the content of the Event.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(False)>
<JsonAttribute(IsSerializableToJson=False, IsSkipIfEmpty=False)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1Dialog.EventTemplate",
Description="C1Dialog.EventTemplate")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Content)>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<System.Web.UI.TemplateInstanceAttribute(TemplateInstance.Single)>
<LayoutAttribute(LayoutType.Behavior)>
Public Property EventTemplate As System.Web.UI.ITemplate
```

C#

```
[System.ComponentModel.Browsable(false)]
[Json(IsSerializableToJson=false, IsSkipIfEmpty=false)]
```

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1Dialog.EventTemplate",
Description="C1Dialog.EventTemplate")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[System.Web.UI.TemplateInstance(TemplateInstance.Single)]
[Layout(LayoutType.Behavior)]
public System.Web.UI.ITemplate EventTemplate {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## EventTitleFormat Property

Format of the title text for the event. Format arguments: {0} = Start, {1} = End, {2} = Subject, {3} = Location, {4} = Icons.

## Syntax

Visual Basic (Declaration)

```
<TypeConverterAttribute("C1.Web.Wijmo.Controls.C1EventsCalendar.EventTitleFor
matConverter, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207,
Culture=neutral, PublicKeyToken=9b75583953471eea")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.EventTitleFormat",
Description="Format of the title text for the event. Format arguments: {0} =
Start, {1} = End, {2} = Subject, {3} = Location, {4} = Icons.")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property EventTitleFormat As System.String
```

C#

```
[TypeConverter("C1.Web.Wijmo.Controls.C1EventsCalendar.EventTitleFormatConver
ter, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
```

```
[C1Description(Key="C1EventsCalendar.EventTitleFormat", Description="Format
of the title text for the event. Format arguments: {0} = Start, {1} = End,
{2} = Subject, {3} = Location, {4} = Icons.")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string EventTitleFormat {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## FirstRowDayHeaderFormat Property

Format of the text for the first cell header in the first row of the month view. Format argument: {0} = Day date.

## Syntax

### Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<C1CategoryAttribute("Appearance")>
<TypeConverterAttribute("C1.Web.Wijmo.Controls.C1EventsCalendar.FirstRowDayHeaderFormatConverter, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral, PublicKeyToken=9b75583953471eea")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.FirstRowDayHeaderFormat",
Description="Format of the text for the first cell header in the first row of
the month view. Format argument: {0} = Day date.")>
Public Property FirstRowDayHeaderFormat As System.String
```

### C#

```
[WidgetOption()]
[Layout(LayoutType.Appearance)]
[C1Category("Appearance")]
[TypeConverter("C1.Web.Wijmo.Controls.C1EventsCalendar.FirstRowDayHeaderFormatConverter, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral, PublicKeyToken=9b75583953471eea")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.FirstRowDayHeaderFormat",
```

```
Description="Format of the text for the first cell header in the first row of
the month view. Format argument: {0} = Day date.")]
public System.string FirstRowDayHeaderFormat {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## HeaderBarVisible Property

Indicates whether the header bar will be visible.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.HeaderBarVisible",
Description="Indicates whether the header bar will be visible.")>
Public Property HeaderBarVisible As System.Boolean
```

C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.HeaderBarVisible",
Description="Indicates whether the header bar will be visible.")]
public System.bool HeaderBarVisible {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## Localization Property

Use the Localization property in order to customize localization strings.

## Syntax

### Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.Localization", Description="Use
the Localization property in order to customize localization strings.")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Content)>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<LayoutAttribute(LayoutType.Appearance)>
Public ReadOnly Property Localization As LocalizationOption
```

### C#

```
[C1Category("Appearance")]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.Localization", Description="Use the
Localization property in order to customize localization strings.")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[Layout(LayoutType.Appearance)]
public LocalizationOption Localization {get;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## NavigationBarVisible Property

Indicates whether the bottom navigation bar will be visible.

## Syntax



## Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.NavigationBarVisible",
Description="Indicates whether the bottom navigation bar will be visible.")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property NavigationBarVisible As System.Boolean
```

## C#

```
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.NavigationBarVisible",
Description="Indicates whether the bottom navigation bar will be visible.")]
[Layout(LayoutType.Appearance)]
public System.bool NavigationBarVisible {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)[C1EventsCalendar Members](#)

## OnClientBeforeAddEvent Property

Occurs before the add event action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new event data that should be added to a data source.

## Syntax

## Visual Basic (Declaration)

```
<C1CategoryAttribute("Client-Side Events")>
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientBeforeAddEvent",
Description="Occurs before the add event action. Return false or call
event.preventDefault() in order to cancel event and prevent the save action.
args.data - the new event data that should be added to a data source.")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetEventAttribute("e, args")>
<WidgetOptionNameAttribute("beforeAddEvent")>
```

```
Public Property OnClientBeforeAddEvent As System.String
```

```
C#
```

```
[C1Category("Client-Side Events")]
[C1Description(Key="C1EventsCalendar.OnClientBeforeAddEvent",
Description="Occurs before the add event action. Return false or call
event.preventDefault() in order to cancel event and prevent the save action.
args.data - the new event data that should be added to a data source.")]
[System.ComponentModel.DefaultValue()]
[WidgetEvent("e, args")]
[WidgetOptionName("beforeAddEvent")]
public System.string OnClientBeforeAddEvent {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## OnClientBeforeDeleteCalendar Property

Occurs before the delete calendar action. Return false or call event.preventDefault() in order to cancel event and prevent the delete action. args.data - the calendar data that should be deleted from a data source.

## Syntax

```
Visual Basic (Declaration)
```

```
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientBeforeDeleteCalendar",
Description="Occurs before the delete calendar action. Return false or call
event.preventDefault() in order to cancel event and prevent the delete
action. args.data - the calendar data that should be deleted from a data
source.")>
<WidgetOptionNameAttribute("beforeDeleteCalendar")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, args")>
Public Property OnClientBeforeDeleteCalendar As System.String
```

```
C#
```

```
[C1Description(Key="C1EventsCalendar.OnClientBeforeDeleteCalendar",
Description="Occurs before the delete calendar action. Return false or call
```

```

event.preventDefault() in order to cancel event and prevent the delete
action. args.data - the calendar data that should be deleted from a data
source.")]
[WidgetOptionName("beforeDeleteCalendar")]
[System.ComponentModel.DefaultValue()]
[C1Category("Client-Side Events")]
[WidgetEvent("e, args")]
public System.string OnClientBeforeDeleteCalendar {get; set;}

```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## OnClientBeforeDeleteEvent Property

Occurs before the delete action. Return false or call event.preventDefault() in order to cancel event and prevent the delete action. args.data - the event object that should be deleted.

## Syntax

Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientBeforeDeleteEvent",
Description="Occurs before the delete action. Return false or call
event.preventDefault() in order to cancel event and prevent the delete
action. args.data - the new event data that should be saved to a data source.
args.data - the event object that should be deleted.")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, args")>
<WidgetOptionNameAttribute("beforeDeleteEvent")>
Public Property OnClientBeforeDeleteEvent As System.String

```

C#

```

[C1Description(Key="C1EventsCalendar.OnClientBeforeDeleteEvent",
Description="Occurs before the delete action. Return false or call
event.preventDefault() in order to cancel event and prevent the delete
action. args.data - the new event data that should be saved to a data source.
args.data - the event object that should be deleted.")]
[System.ComponentModel.DefaultValue()]
[C1Category("Client-Side Events")]

```

```
[WidgetEvent("e, args")]
[WidgetOptionName("beforeDeleteEvent")]
public System.string OnClientBeforeDeleteEvent {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## OnClientBeforeEditEventDialogShow Property

Occurs before the built-in event dialog is shown. Return false or call event.preventDefault() in order to cancel event and prevent the built-in dialog to be shown. args.data - the event data. args.targetCell - target offset DOM element which can be used for popup callout.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Client-Side Events")>
<WidgetOptionNameAttribute("beforeEditEventDialogShow")>
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientBeforeEditEventDialogShow", Description="Occurs before the built-in event dialog is shown. Return false or call event.preventDefault() in order to cancel event and prevent the built-in dialog to be shown. args.data - the event data. args.targetCell - target offset DOM element which can be used for popup callout.")>
<WidgetEventAttribute("e, args")>
Public Property OnClientBeforeEditEventDialogShow As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Client-Side Events")]
[WidgetOptionName("beforeEditEventDialogShow")]
[C1Description(Key="C1EventsCalendar.OnClientBeforeEditEventDialogShow", Description="Occurs before the built-in event dialog is shown. Return false or call event.preventDefault() in order to cancel event and prevent the built-in dialog to be shown. args.data - the event data. args.targetCell - target offset DOM element which can be used for popup callout.")]
[WidgetEvent("e, args")]
public System.string OnClientBeforeEditEventDialogShow {get; set;}
```

## See Also

## Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## OnClientBeforeSaveCalendar Property

Occurs before the save calendar action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new calendar data that should be saved to a data source. args.prevData - previous calendar data, this argument is empty for a new calendar.

## Syntax

### Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("beforeSaveCalendar")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientBeforeSaveCalendar",
Description="Occurs before the save calendar action. Return false or call
event.preventDefault() in order to cancel event and prevent the save action.
args.data - the new calendar data that should be saved to a data source.
args.prevData - previous calendar data, this argument is empty for a new
calendar.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, args")>
Public Property OnClientBeforeSaveCalendar As System.String
```

### C#

```
[WidgetOptionName("beforeSaveCalendar")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.OnClientBeforeSaveCalendar",
Description="Occurs before the save calendar action. Return false or call
event.preventDefault() in order to cancel event and prevent the save action.
args.data - the new calendar data that should be saved to a data source.
args.prevData - previous calendar data, this argument is empty for a new
calendar.")]
[C1Category("Client-Side Events")]
[WidgetEvent("e, args")]
public System.string OnClientBeforeSaveCalendar {get; set;}
```

## See Also

## Reference

[C1EventsCalendar Class](#)[C1EventsCalendar Members](#)

## OnClientBeforeUpdateEvent Property

Occurs before the update event action. Return false or call event.preventDefault() in order to cancel event and prevent the save action. args.data - the new event data that should be updated. args.prevData - previous event data, this argument is empty for a new event.

### Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientBeforeUpdateEvent",
Description="Occurs before the save event action. Return false or call
event.preventDefault() in order to cancel event and prevent the save action.
args.data - the new event data that should be updated. args.prevData -
previous event data, this argument is empty for a new event.")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, args")>
<WidgetOptionNameAttribute("beforeUpdateEvent")>
Public Property OnClientBeforeUpdateEvent As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.OnClientBeforeUpdateEvent",
Description="Occurs before the save event action. Return false or call
event.preventDefault() in order to cancel event and prevent the save action.
args.data - the new event data that should be updated. args.prevData -
previous event data, this argument is empty for a new event.")]
[System.ComponentModel.DefaultValue()]
[C1Category("Client-Side Events")]
[WidgetEvent("e, args")]
[WidgetOptionName("beforeUpdateEvent")]
public System.string OnClientBeforeUpdateEvent {get; set;}
```

### See Also

#### Reference

[C1EventsCalendar Class](#)[C1EventsCalendar Members](#)

## OnClientCalendarsChanged Property

Occurs when calendars option has been changed. args.calendars - the new calendars option value.

### Syntax

Visual Basic (Declaration)

```
<WidgetEventAttribute("e, args")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientCalendarsChanged",
Description="Occurs when calendars option has been changed. args.calendars -
the new calendars option value.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetOptionNameAttribute("calendarsChanged")>
Public Property OnClientCalendarsChanged As System.String
```

C#

```
[WidgetEvent("e, args")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.OnClientCalendarsChanged",
Description="Occurs when calendars option has been changed. args.calendars -
the new calendars option value.")]
[C1Category("Client-Side Events")]
[WidgetOptionName("calendarsChanged")]
public System.string OnClientCalendarsChanged {get; set;}
```

### See Also

#### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## OnClientEventCreating Property

Occurs when event markup is creating. args.data - the event data. args.eventTemplate The template of events displayed on the view.

### Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("eventCreating")>
```

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientEventCreating",
Description="Occurs when event markup is creating. args.data - the event
data. args.eventTemplate The template of events displayed on the view.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, args")>
Public Property OnClientEventCreating As System.String
```

C#

```
[WidgetOptionName("eventCreating")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.OnClientEventCreating",
Description="Occurs when event markup is creating. args.data - the event
data. args.eventTemplate The template of events displayed on the view.")]
[C1Category("Client-Side Events")]
[WidgetEvent("e, args")]
public System.string OnClientEventCreating {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## OnClientInitialized Property

Occurs when events calendar is constructed and events data is loaded from an external or local data source.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientInitialized",
Description="Occurs when events calendar is constructed and events data is
loaded from an external or local data source.")>
<WidgetOptionNameAttribute("initialized")>
<WidgetEventAttribute("e")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Client-Side Events")>
Public Property OnClientInitialized As System.String
```

C#



```
[C1Description(Key="C1EventsCalendar.OnClientInitialized",
Description="Occurs when events calendar is constructed and events data is
loaded from an external or local data source.")]
[WidgetOptionName("initialized")]
[WidgetEvent("e")]
[System.ComponentModel.DefaultValue()]
[C1Category("Client-Side Events")]
public System.string OnClientInitialized {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## OnClientSelectedDatesChanged Property

Occurs when selectedDates option has been changed. args.selectedDates - the new selectedDates value.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("selectedDatesChanged")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientSelectedDatesChanged",
Description="Occurs when selectedDates option has been changed.
args.selectedDates - the new selectedDates value.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, args")>
Public Property OnClientSelectedDatesChanged As System.String
```

C#

```
[WidgetOptionName("selectedDatesChanged")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.OnClientSelectedDatesChanged",
Description="Occurs when selectedDates option has been changed.
args.selectedDates - the new selectedDates value.")]
[C1Category("Client-Side Events")]
[WidgetEvent("e, args")]
public System.string OnClientSelectedDatesChanged {get; set;}
```

## See Also

## Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## OnClientViewTypeChanged Property

Occurs when viewType option has been changed.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("viewTypeChanged")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.OnClientViewTypeChanged",
Description="Occurs when viewType option has been changed.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, viewType")>
Public Property OnClientViewTypeChanged As System.String
```

C#

```
[WidgetOptionName("viewTypeChanged")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.OnClientViewTypeChanged",
Description="Occurs when viewType option has been changed.")]
[C1Category("Client-Side Events")]
[WidgetEvent("e, viewType")]
public System.string OnClientViewTypeChanged {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## SelectedDate Property

The selected date.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.SelectedDate", Description="The
```

```

selected date.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.ComponentModel.DefaultValueAttribute(<>)>
<System.ComponentModel.BindableAttribute(Bindable=True,
Direction=BindingDirection.TwoWay)>
<WidgetOptionAttribute(<>)>
Public Property SelectedDate As System.Date

```

C#

```

[C1Description(Key="C1EventsCalendar.SelectedDate", Description="The selected
date.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.ComponentModel.DefaultValue(<>)]
[System.ComponentModel.Bindable(Bindable=true,
Direction=BindingDirection.TwoWay)]
[WidgetOption(<>)]
public System.DateTime SelectedDate {get; set;}

```

### Property Value

A **System.DateTime** that represents the selected date.

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## SelectedDates Property

Collection of DateTime objects that represent the selected dates on the C1EventsCalendar control for the DayView.

## Syntax

Visual Basic (Declaration)

```

<System.ComponentModel.BrowsableAttribute(False)>
<C1DescriptionAttribute(Key="C1EventsCalendar.SelectedDates",
Description="Collection of DateTime objects that represent the selected dates
on the C1EventsCalendar control for the DayView.")>
<LayoutAttribute(LayoutType.Appearance)>

```

```
<WidgetOptionAttribute()>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Hidden)>
Public ReadOnly Property SelectedDates As
System.Web.UI.WebControls.SelectedDatesCollection
```

C#

```
[System.ComponentModel.Browsable(false)]
[C1Description(Key="C1EventsCalendar.SelectedDates", Description="Collection
of DateTime objects that represent the selected dates on the C1EventsCalendar
control for the DayView.")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
public System.Web.UI.WebControls.SelectedDatesCollection SelectedDates {get;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## TimeInterval Property

Time interval for the Day view (in minutes).

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.TimeInterval",
Description="Time interval for the Day view (in minutes).")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Property TimeInterval As System.Integer
```

C#

```
[C1Description(Key="C1EventsCalendar.TimeInterval", Description="Time
interval for the Day view (in minutes).")]
```

```
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[WidgetOption()]
[Layout(LayoutType.Appearance)]
public System.int TimeInterval {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## TimeIntervalHeight Property

The Day view time interval row height in pixels.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.TimeIntervalHeight",
Description="The Day view time interval row height in pixels.")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property TimeIntervalHeight As System.Integer
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.TimeIntervalHeight", Description="The
Day view time interval row height in pixels.")]
[Layout(LayoutType.Appearance)]
public System.int TimeIntervalHeight {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## TimeRulerFormat Property

Time ruler format for the Day view. Format argument: {0} = Current ruler time.

### Syntax

#### Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.TimeRulerFormat",
Description="Time ruler format for the Day view. Format argument: {0} =
Current ruler time.")>
<TypeConverterAttribute("C1.Web.Wijmo.Controls.C1EventsCalendar.TimeRulerForm
atConverter, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property TimeRulerFormat As System.String
```

#### C#

```
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.TimeRulerFormat", Description="Time
ruler format for the Day view. Format argument: {0} = Current ruler time.")]
[TypeConverter("C1.Web.Wijmo.Controls.C1EventsCalendar.TimeRulerFormatConvert
er, C1.Web.Wijmo.Controls.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[Layout(LayoutType.Appearance)]
public System.string TimeRulerFormat {get; set;}
```

### See Also

#### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## TimeRulerInterval Property

Time ruler interval for the Day view (in minutes).

### Syntax

## Visual Basic (Declaration)

```

<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1EventsCalendar.TimeRulerInterval",
Description="Time ruler interval for the Day view (in minutes).")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
Public Property TimeRulerInterval As System.Integer

```

## C#

```

[WidgetOption()]
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1EventsCalendar.TimeRulerInterval", Description="Time
ruler interval for the Day view (in minutes).")]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
public System.int TimeRulerInterval {get; set;}

```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## Views Property

List of the views which need to be shown. Both default views("day", "week", "month", "list") and custom views can be added. If add a default view, only Type property of View should be set. If add a custom view, All the properties of View should be set.

## Syntax

## Visual Basic (Declaration)

```

<WidgetOptionAttribute()>
<CollectionItemTypeAttribute(C1.Web.Wijmo.Controls.C1EventsCalendar.View)>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.Views", Description="List of
the view types which need to be shown.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<System.ComponentModel.MergablePropertyAttribute(False)>

```

```
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Content)>
<LayoutAttribute(LayoutType.Appearance)>
Public ReadOnly Property Views As System.Collections.Generic.List(Of View)
```

C#

```
[WidgetOption()]
[CollectionItemType(C1.Web.Wijmo.Controls.C1EventsCalendar.View)]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.Views", Description="List of the view
types which need to be shown.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[System.ComponentModel.MergableProperty(false)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[Layout(LayoutType.Appearance)]
public System.Collections.Generic.List<View> Views {get;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## VisibleCalendars Property

Array of the calendar names which need to be shown.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.VisibleCalendars",
Description="Array of the calendar names which need to be shown.")>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<TypeConverterAttribute("System.Web.UI.WebControls.StringArrayConverter,
System.Web, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property VisibleCalendars As System.String()
```



C#

```
[C1Description(Key="C1EventsCalendar.VisibleCalendars", Description="Array of
the calendar names which need to be shown.")]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[TypeConverter("System.Web.UI.WebControls.StringArrayConverter, System.Web,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string[] VisibleCalendars {get; set;}
```

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

## WebServiceUrl Property

The URL to the web service which will be used to store information about events.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Behavior")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.WebServiceUrl",
Description="The URL to the web service which will be used to store
information about events.")>
<System.Web.UI.UrlPropertyAttribute()>
Public Property WebServiceUrl As System.String
```

C#

```
[C1Category("Behavior")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.WebServiceUrl", Description="The URL to
the web service which will be used to store information about events.")]
[System.Web.UI.UrlProperty()]
public System.string WebServiceUrl {get; set;}
```

## Property Value

The HTTP handler URL.

## See Also

### Reference

[C1EventsCalendar Class](#)

[C1EventsCalendar Members](#)

# C1EventsCalendarSerializer

Serializes the C1EventsCalendar Control.

## Object Model

C1EventsCalendarSerializer

## Syntax

Visual Basic (Declaration)	
<pre>Public Class C1EventsCalendarSerializer     Inherits C1.Web.Wijmo.Controls.Base.C1BaseSerializer(Of C1EventsCalendar,Object,C1EventsCalendar)</pre>	
C#	
<pre>public class C1EventsCalendarSerializer : C1.Web.Wijmo.Controls.Base.C1BaseSerializer&lt;C1EventsCalendar,object,C1EventsC alendar&gt;</pre>	

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendarSerializer**

## See Also

### Reference

[C1EventsCalendarSerializer Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Serializes the C1EventsCalendar Control.

## Object Model

C1EventsCalendarSerializer

## Syntax

Visual Basic (Declaration)

```
Public Class C1EventsCalendarSerializer
    Inherits C1.Web.Wijmo.Controls.Base.C1BaseSerializer(Of
C1EventsCalendar, Object, C1EventsCalendar)
```

C#

```
public class C1EventsCalendarSerializer :
C1.Web.Wijmo.Controls.Base.C1BaseSerializer<C1EventsCalendar, object, C1EventsC
alendar>
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendarSerializer**

## See Also

### Reference


[C1EventsCalendarSerializer Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members

The following tables list the members exposed by [C1EventsCalendarSerializer](#).

### Public Constructors

	Name	Description
	<a href="#">C1EventsCalendarSerializer Constructor</a>	Initializes a new instance of the C1EventsCalendarSerializer class.

[Top](#)

## See Also

### Reference

[C1EventsCalendarSerializer Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## C1EventsCalendarSerializer Constructor

Initializes a new instance of the C1EventsCalendarSerializer class.

### Syntax

Visual Basic (Declaration)

```
Public Function New( _  
    ByVal obj As System.Object _  
)
```

C#

```
public C1EventsCalendarSerializer(  
    System.object obj  
)
```

### Parameters

*obj*

### See Also

### Reference

[C1EventsCalendarSerializer Class](#)[C1EventsCalendarSerializer Members](#)

## C1EventsCalendarStorage

The C1EventsCalendarStorage defines data storage configuration.

### Object Model

C1EventsCalendarStorage

### Syntax

Visual Basic (Declaration)

```
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>  
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,  
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
```

```
Public Class C1EventsCalendarStorage
```

```
C#
```

```
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]  
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,  
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]  
public class C1EventsCalendarStorage
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendarStorage**

## See Also

### Reference

[C1EventsCalendarStorage Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

The C1EventsCalendarStorage defines data storage configuration.

## Object Model

C1EventsCalendarStorage

## Syntax

Visual Basic (Declaration)

```
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>  
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,  
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>  
Public Class C1EventsCalendarStorage
```

```
C#
```

```
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]  
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,  
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]  
public class C1EventsCalendarStorage
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendarStorage**

## See Also

### Reference






[C1EventsCalendarStorage Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members




The following tables list the members exposed by [C1EventsCalendarStorage](#).













### Public Properties

	Name	Description
	<a href="#">Calendars</a>	Gets the calendars collection.
	<a href="#">CalendarStorage</a>	Gets the <b>WebStatusStorage</b> object.
	<a href="#">DataFile</a>	The name of the xml file which will be used for unbound mode.
	<a href="#">Events</a>	Gets the events collection.
	<a href="#">EventStorage</a>	The EventStorage property defines data storage configuration for events..

[Top](#)

### Public Methods

	Name	Description
	<a href="#">AddCalendar</a>	
	<a href="#">AddEvent</a>	Overloaded. Adds the event.
	<a href="#">BeginUpdate</a>	Prevents the events calendar storage from being updated until the EndUpdate method is called.

 <a href="#">DeleteCalendar</a>	
 <a href="#">DeleteEvent</a>	Overloaded. Deletes the event.
 <a href="#">EndUpdate</a>	Unlocks the events calendar storage after a call to the BeginUpdate method call and causes an immediate update.
 <a href="#">Export</a>	Overloaded. Exports the events calendar's data to a file in the specified format.
 <a href="#">FindEventByID</a>	Finds the event by its ID.
 <a href="#">Import</a>	Overloaded. Imports data into the events calendar from a file of the specified format.
 <a href="#">Initialize</a>	Initialize data storage.
 <a href="#">LoadData</a>	Load events calendar data from data source or from xml file.
 <a href="#">SaveData</a>	Save events data to data source or xml file depending on data storage settings.
 <a href="#">ToString</a>	ToString
 <a href="#">UpdateCalendar</a>	
 <a href="#">UpdateEvent</a>	Overloaded. Updates the event.

[Top](#)

## See Also

### Reference

[C1EventsCalendarStorage Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Methods

For a list of all members of this type, see [C1EventsCalendarStorage members](#).

## Public Methods

	Name	Description
≡	<a href="#">AddCalendar</a>	
≡	<a href="#">AddEvent</a>	Overloaded. Adds the event.
≡	<a href="#">BeginUpdate</a>	Prevents the events calendar storage from being updated until the EndUpdate method is called.
≡	<a href="#">DeleteCalendar</a>	
≡	<a href="#">DeleteEvent</a>	Overloaded. Deletes the event.
≡	<a href="#">EndUpdate</a>	Unlocks the events calendar storage after a call to the BeginUpdate method call and causes an immediate update.
≡	<a href="#">Export</a>	Overloaded. Exports the events calendar's data to a file in the specified format.
≡	<a href="#">FindEventByID</a>	Finds the event by its ID.
≡	<a href="#">Import</a>	Overloaded. Imports data into the events calendar from a file of the specified format.
≡	<a href="#">Initialize</a>	Initialize data storage.
≡	<a href="#">LoadData</a>	Load events calendar data from data source or from xml file.
≡	<a href="#">SaveData</a>	Save events data to data source or xml file depending on data storage settings.
≡	<a href="#">ToString</a>	ToString
≡	<a href="#">UpdateCalendar</a>	
≡	<a href="#">UpdateEvent</a>	Overloaded. Updates the event.

[Top](#)



## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## AddCalendar Method

### Syntax

Visual Basic (Declaration)	
<pre>Public Function AddCalendar( _     ByVal calData As System.Collections.Hashtable _ ) As System.Collections.Hashtable</pre>	
C#	
<pre>public System.Collections.Hashtable AddCalendar(     System.Collections.Hashtable calData )</pre>	

### Parameters

*calData*

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## AddEvent Method

Adds the event.

### Overload List

Overload	Description
<a href="#">AddEvent(Event)</a>	Adds the event.
<a href="#">AddEvent(Hashtable)</a>	Adds the event.

## See Also

## Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## AddEvent(Event) Method

Adds the event.

## Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Function AddEvent( _     ByVal ev As <a href="#">Event</a> _ ) As <a href="#">Event</a></pre>	
C#	
<pre>public <a href="#">Event</a> AddEvent(     <a href="#">Event</a> ev )</pre>	

## Parameters

*ev*

The event.

## Return Value

Event object which will be added to data source or data file.

## See Also

## Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

[Overload List](#)

## AddEvent(Hashtable) Method

Adds the event.

## Syntax

Visual Basic (Declaration)	
----------------------------	--

```
Public Overloads Function AddEvent( _  
    ByVal eventData As System.Collections.Hashtable _  
) As Event
```

C#

```
public Event AddEvent(  
    System.Collections.Hashtable eventData  
)
```

### Parameters

*eventData*

The raw event data.

### Return Value

Event object which will be added to data source or data file.

## See Also

### Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)  
[Overload List](#)

## BeginUpdate Method

Prevents the events calendar storage from being updated until the EndUpdate method is called.

## Syntax

Visual Basic (Declaration)

```
Public Sub BeginUpdate()
```

C#

```
public void BeginUpdate()
```

## See Also

### Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)

## DeleteCalendar Method

### Syntax

Visual Basic (Declaration)

```
Public Sub DeleteCalendar( _  
    ByVal calData As System.Collections.Hashtable _  
)
```

C#

```
public void DeleteCalendar(  
    System.Collections.Hashtable calData  
)
```

### Parameters

*calData*

### See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## DeleteEvent Method

Deletes the event.

### Overload List

Overload	Description
<a href="#">DeleteEvent(Event)</a>	Deletes the event.
<a href="#">DeleteEvent(Hashtable)</a>	Deletes the event.

### See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## DeleteEvent(Event) Method

Deletes the event.

### Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Sub DeleteEvent( _     ByVal ev As Event _ )</pre>	
C#	
<pre>public void DeleteEvent(     Event ev )</pre>	

### Parameters

*ev*

The event.

### See Also

### Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)  
[Overload List](#)

## DeleteEvent(Hashtable) Method

Deletes the event.

### Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Sub DeleteEvent( _     ByVal eventData As System.Collections.Hashtable _ )</pre>	
C#	
<pre>public void DeleteEvent(     System.Collections.Hashtable eventData</pre>	

```
)
```

## Parameters

*eventData*

The raw event data.

## See Also

## Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

[Overload List](#)

## EndUpdate Method

Unlocks the events calendar storage after a call to the BeginUpdate method call and causes an immediate update.

## Syntax

Visual Basic (Declaration)	
<pre>Public Sub EndUpdate()</pre>	
C#	
<pre>public void EndUpdate()</pre>	

## See Also

## Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## Export Method

Exports the events calendar's data to a file in the specified format.

## Overload List

Overload	Description
<a href="#">Export(String,FileFormatEnum)</a>	Exports the events calendar's data to a file in the specified format.

<a href="#">Export(Stream,FileFormatEnum)</a>	Exports the events calendar's data to a stream in the specified format.
---	---

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## Export(String,FileFormatEnum) Method

A **System.String** containing the full path (including the file name and extension) to export the events calendar's data to.

The [FileFormatEnum](#) value.

Exports the events calendar's data to a file in the specified format.

## Syntax

Visual Basic (Declaration)

```
Public Overloads Sub Export( _  
    ByVal path As System.String, _  
    ByVal format As FileFormatEnum _  
)
```

C#

```
public void Export(  
    System.string path,  
    FileFormatEnum format  
)
```

### Parameters

*path*

A **System.String** containing the full path (including the file name and extension) to export the events calendar's data to.

*format*

The [FileFormatEnum](#) value.

## See Also

## Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)  
[Overload List](#)

## Export(Stream,FileFormatEnum) Method

A **System.IO.Stream** object which specifies the stream into which the events calendar's data will be exported.

The [FileFormatEnum](#) value.

Exports the events calendar's data to a stream in the specified format.

## Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Sub Export( _     ByVal stream As System.IO.Stream, _     ByVal format As FileFormatEnum _ )</pre>	
C#	
<pre>public void Export(     System.IO.Stream stream,     FileFormatEnum format )</pre>	

## Parameters

*stream*

A **System.IO.Stream** object which specifies the stream into which the events calendar's data will be exported.

*format*

The [FileFormatEnum](#) value.

## See Also

## Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)  
[Overload List](#)



## FindEventByID Method

The event id.

Finds the event by its ID.

### Syntax

Visual Basic (Declaration)

```
Public Function FindEventByID( _  
    ByVal id As System.String _  
) As Event
```

C#

```
public Event FindEventByID(  
    System.string id  
)
```

### Parameters

*id*

The event id.

### See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## Import Method

Imports data into the events calendar from a file of the specified format.

### Overload List

Overload	Description
<a href="#">Import(String,FileFormatEnum)</a>	Imports data into the events calendar from a file of the specified format.
<a href="#">Import(Stream,FileFormatEnum)</a>	Imports the events calendar's data from a stream whose data is in the specified format.

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## Import(String,FileFormatEnum) Method

A **System.String** value containing the full path (including the file name and extension) to a file which contains the data to be imported into the events calendar.

The [FileFormatEnum](#) value.

Imports data into the events calendar from a file of the specified format.

## Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Sub Import( _     ByVal <i>path</i> As System.String, _     ByVal <i>format</i> As <a href="#">FileFormatEnum</a> _ )</pre>	
C#	
<pre>public void Import(     System.string <i>path</i>,     <a href="#">FileFormatEnum</a> <i>format</i> )</pre>	

### Parameters

*path*

A **System.String** value containing the full path (including the file name and extension) to a file which contains the data to be imported into the events calendar.

*format*

The [FileFormatEnum](#) value.

## See Also

### Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)  
[Overload List](#)

## Import(Stream,FileFormatEnum) Method

A **System.IO.Stream** object which specifies the stream that contains the data to import to the events calendar.

The [FileFormatEnum](#) value.

Imports the events calendar's data from a stream whose data is in the specified format.

## Syntax

Visual Basic (Declaration)
<pre>Public Overloads Sub Import( _     ByVal stream As System.IO.Stream, _     ByVal format As FileFormatEnum _ )</pre>
C#
<pre>public void Import(     System.IO.Stream stream,     FileFormatEnum format )</pre>

## Parameters

*stream*

A **System.IO.Stream** object which specifies the stream that contains the data to import to the events calendar.

*format*

The [FileFormatEnum](#) value.

## See Also

## Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)  
[Overload List](#)

## Initialize Method

Initialize data storage.

### Syntax

Visual Basic (Declaration)	
Public Sub Initialize()	
C#	
public void Initialize()	

### See Also

#### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## LoadData Method

Load events calendar data from data source or from xml file.

### Syntax

Visual Basic (Declaration)	
Public Sub LoadData()	
C#	
public void LoadData()	

### See Also

#### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## SaveData Method

Save events data to data source or xml file depending on data storage settings.

### Syntax

Visual Basic (Declaration)	
Public Sub SaveData()	
C#	
public void SaveData()	

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## ToString Method

ToString

### Syntax

Visual Basic (Declaration)	
Public Overrides Function ToString() As System.String	
C#	
public override System.string ToString()	

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## UpdateCalendar Method

### Syntax

Visual Basic (Declaration)	
Public Function UpdateCalendar( _ ByVal calData As System.Collections.Hashtable _ ) As System.Collections.Hashtable	
C#	

```
public System.Collections.Hashtable UpdateCalendar(  
    System.Collections.Hashtable calData  
)
```

### Parameters

*calData*

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## UpdateEvent Method

Updates the event.

## Overload List

Overload	Description
<a href="#">UpdateEvent(Event)</a>	Updates the event.
<a href="#">UpdateEvent(Hashtable)</a>	Updates the event.

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## UpdateEvent(Event) Method

Updates the event.

## Syntax

Visual Basic (Declaration)

```
Public Overloads Function UpdateEvent( _  
    ByVal ev As Event _  
) As Event
```

C#

```
public Event UpdateEvent(  
    Event ev  
)
```

### Parameters

*ev*

The event.

## See Also

### Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)  
[Overload List](#)

## UpdateEvent(Hashtable) Method

Updates the event.

## Syntax

Visual Basic (Declaration)	
Public Overloads Function UpdateEvent( _ ByVal <i>eventData</i> As System.Collections.Hashtable _ ) As <a href="#">Event</a>	
C#	
public <a href="#">Event</a> UpdateEvent( System.Collections.Hashtable <i>eventData</i> )	

### Parameters

*eventData*

The raw event data.

## See Also






### Reference

[C1EventsCalendarStorage Class](#)  
[C1EventsCalendarStorage Members](#)  
[Overload List](#)

## Properties

For a list of all members of this type, see [C1EventsCalendarStorage members](#).

### Public Properties

	Name	Description
	<a href="#">Calendars</a>	Gets the calendars collection.
	<a href="#">CalendarStorage</a>	Gets the <b>WebStatusStorage</b> object.
	<a href="#">DataFile</a>	The name of the xml file which will be used for unbound mode.
	<a href="#">Events</a>	Gets the events collection.
	<a href="#">EventStorage</a>	The EventStorage property defines data storage configuration for events..

[Top](#)

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Calendars Property

Gets the calendars collection.

### Syntax

Visual Basic (Declaration)	
<pre>&lt;System.ComponentModel.BrowsableAttribute(False)&gt; &lt;System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Hidden)&gt; Public ReadOnly Property Calendars As System.Collections.Generic.IList(Of Object)</pre>	
C#	



```
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
public System.Collections.Generic.IList<object> Calendars {get;}
```

### Property Value

The calendars.

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## CalendarStorage Property

Gets the **WebStatusStorage** object.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Storages")>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<C1DescriptionAttribute(Key="C1EventsCalendarStorage.CalendarStorage",
Description="The CalendarStorage property defines data storage configuration
for calendars.")>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property CalendarStorage As C1CalendarStorage
```

C#

```
[C1Category("Storages")]
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[C1Description(Key="C1EventsCalendarStorage.CalendarStorage",
Description="The CalendarStorage property defines data storage configuration
```

```
for calendars."))
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[System.ComponentModel.NotifyParentProperty(true)]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
public C1CalendarStorage CalendarStorage {get;}
```

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## DataFile Property

The name of the xml file which will be used for unbound mode.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendarStorage.DataFile",
Description="The name of the xml file which will be used for unbound mode.")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property DataFile As System.String
```

C#

```
[C1Description(Key="C1EventsCalendarStorage.DataFile", Description="The name
of the xml file which will be used for unbound mode.")]
[System.ComponentModel.DefaultValue()]
public System.string DataFile {get; set;}
```

### Property Value

The data file.

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## Events Property

Gets the events collection.

### Syntax

Visual Basic (Declaration)	
<pre>&lt;System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Hidden)&gt; &lt;System.ComponentModel.BrowsableAttribute(False)&gt; Public ReadOnly Property Events As <a href="#">C1EventCollection</a></pre>	
C#	
<pre>[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Hidden)] [System.ComponentModel.Browsable(false)] public <a href="#">C1EventCollection</a> Events {get;}</pre>	

### Property Value

The events.

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## EventStorage Property

The EventStorage property defines data storage configuration for events..

### Syntax

Visual Basic (Declaration)	
<pre>&lt;System.ComponentModel.NotifyParentPropertyAttribute(True)&gt; &lt;C1CategoryAttribute("Storages")&gt; &lt;C1DescriptionAttribute(Key="C1EventsCalendarStorage.EventStorage", Description="The EventStorage property defines data storage configuration for events.")&gt; &lt;System.ComponentModel.BindableAttribute(Bindable=False, Direction=BindingDirection.OneWay)&gt; &lt;System.ComponentModel.BrowsableAttribute(True)&gt;</pre>	

```
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
Public ReadOnly Property EventStorage As C1EventStorage
```

C#

```
[System.ComponentModel.NotifyParentProperty(true)]
[C1Category("Storages")]
[C1Description(Key="C1EventsCalendarStorage.EventStorage", Description="The
EventStorage property defines data storage configuration for events.")]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[System.ComponentModel.Browsable(true)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
public C1EventStorage EventStorage {get;}
```

## See Also

### Reference

[C1EventsCalendarStorage Class](#)

[C1EventsCalendarStorage Members](#)

## C1EventsCalendarXmlExchanger

Allows import/export events data to/from XML.

## Object Model

[C1EventsCalendarXmlExchanger](#)

## Syntax

Visual Basic (Declaration)

```
Public Class C1EventsCalendarXmlExchanger
```

C#

```
public class C1EventsCalendarXmlExchanger
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendarXmlExchanger**

## See Also

### Reference

[C1EventsCalendarXmlExchanger Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Allows import/export events data to/from XML.

## Object Model

C1EventsCalendarXmlExchanger

## Syntax

Visual Basic (Declaration)

```
Public Class C1EventsCalendarXmlExchanger
```

C#

```
public class C1EventsCalendarXmlExchanger
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendarXmlExchanger**

## See Also

### Reference

[C1EventsCalendarXmlExchanger Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members

The following tables list the members exposed by [C1EventsCalendarXmlExchanger](#).

## Public Constructors

	Name	Description
	<a href="#">C1EventsCalendarXmlExchanger Constructor</a>	




[Top](#)

## Public Properties

	Name	Description
	<a href="#">Calendars</a>	
	<a href="#">Events</a>	

[Top](#)

## Public Methods

	Name	Description
	<a href="#">Export</a>	Overloaded. Exports the scheduler's data to a file.
	<a href="#">Import</a>	Overloaded. Imports data into the scheduler from a file.
 <b>S</b>	<a href="#">ReadStringValue</a>	

[Top](#)

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## C1EventsCalendarXmlExchanger Constructor

### Syntax

Visual Basic (Declaration)	
<b>Public Function New()</b>	

C#

```
public C1EventsCalendarXmlExchanger()
```

## See Also




### Reference

[C1EventsCalendarXmlExchanger Class](#)[C1EventsCalendarXmlExchanger Members](#)

## Methods

For a list of all members of this type, see [C1EventsCalendarXmlExchanger members](#).

## Public Methods

	Name	Description
	<a href="#">Export</a>	Overloaded. Exports the scheduler's data to a file.
	<a href="#">Import</a>	Overloaded. Imports data into the scheduler from a file.
	<a href="#">ReadStringValue</a>	

[Top](#)

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Export Method

Exports the scheduler's data to a file.

## Overload List

Overload	Description
<a href="#">Export(String)</a>	Exports the scheduler's data to a file.
<a href="#">Export(Stream)</a>	Exports the scheduler's data to a stream.

<a href="#">Export(String,IList&lt;Event&gt;)</a>	Exports the appointments's data to a file.
<a href="#">Export(Stream,IList&lt;Event&gt;)</a>	Exports the events's data to a stream.

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

## Export(String) Method

A **System.String** containing the full path (including the file name and extension) to export the scheduler's data to.

Exports the scheduler's data to a file.

## Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Sub Export( _     ByVal <i>path</i> As System.String _ )</pre>	
C#	
<pre>public void Export(     System.string <i>path</i> )</pre>	

### Parameters

*path*

A **System.String** containing the full path (including the file name and extension) to export the scheduler's data to.

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

[Overload List](#)



## Export(Stream) Method

A **System.IO.Stream** object which specifies the stream into which the scheduler's data will be exported.

Exports the scheduler's data to a stream.

## Syntax

Visual Basic (Declaration)

```
Public Overloads Sub Export( _  
    ByVal stream As System.IO.Stream _  
)
```

C#

```
public void Export(  
    System.IO.Stream stream  
)
```

## Parameters

*stream*

A **System.IO.Stream** object which specifies the stream into which the scheduler's data will be exported.

## See Also

## Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

[Overload List](#)

## Export(String,IList<Event>) Method

A **System.String** containing the full path (including the file name and extension) to export.

A **System.Collections.Generic.IList<T>** object.

Exports the appointments's data to a file.

## Syntax

Visual Basic (Declaration)

```
Public Overloads Sub Export( _
```

```

ByVal path As System.String, _
    ByVal appointments As System.Collections.Generic.IList(Of Event) _
)

```

C#

```

public void Export(
    System.string path,
    System.Collections.Generic.IList<Event> appointments
)

```

### Parameters

*path*

A **System.String** containing the full path (including the file name and extension) to export.

*appointments*

A **System.Collections.Generic.IList`1** object.

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

[Overload List](#)

## Export(Stream,IList<Event>) Method

A **System.IO.Stream** object which specifies the stream into which the appointments's data will be exported.

A **System.Collections.Generic.IList`1** object.

Exports the events's data to a stream.

## Syntax

Visual Basic (Declaration)

```

Public Overloads Sub Export( _
    ByVal stream As System.IO.Stream, _
    ByVal appointments As System.Collections.Generic.IList(Of Event) _
)

```

C#

```
public void Export(  
    System.IO.Stream stream,  
    System.Collections.Generic.IList<Event> appointments  
)
```

### Parameters

*stream*

A **System.IO.Stream** object which specifies the stream into which the appointments's data will be exported.

*appointments*

A **System.Collections.Generic.IList`1** object.

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

[Overload List](#)

## Import Method

Imports data into the scheduler from a file.

## Overload List

Overload	Description
<a href="#">Import(String)</a>	Imports data into the scheduler from a file.
<a href="#">Import(Stream)</a>	Imports the scheduler's data from a stream.

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

## Import(String) Method

A **System.String** value containing the full path (including the file name and extension) to a file which contains the data to be imported into the scheduler.

Imports data into the scheduler from a file.

### Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Sub Import( _     ByVal <i>path</i> As System.String _ )</pre>	
C#	
<pre>public void Import(     System.string <i>path</i> )</pre>	

### Parameters

*path*

A **System.String** value containing the full path (including the file name and extension) to a file which contains the data to be imported into the scheduler.

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)  
[C1EventsCalendarXmlExchanger Members](#)  
[Overload List](#)

## Import(Stream) Method

A **System.IO.Stream** object which specifies the stream that contains the data to import to the scheduler.

Imports the scheduler's data from a stream.

### Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Sub Import( _     ByVal <i>stream</i> As System.IO.Stream _</pre>	

```
)  
C#  
public void Import(  
    System.IO.Stream stream  
)
```

### Parameters

*stream*

A **System.IO.Stream** object which specifies the stream that contains the data to import to the scheduler.

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

[Overload List](#)

## ReadStringValue Method

### Syntax

Visual Basic (Declaration)

```
Public Shared Function ReadStringValue( _  
    ByVal reader As System.Xml.XmlReader _  
) As System.String
```

C#

```
public static System.string ReadStringValue(  
    System.Xml.XmlReader reader  
)
```

### Parameters

*reader*

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

## Properties

For a list of all members of this type, see [C1EventsCalendarXmlExchanger members](#).

### Public Properties

	Name	Description
	<a href="#">Calendars</a>	
	<a href="#">Events</a>	

[Top](#)

### See Also

#### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Calendars Property

### Syntax

Visual Basic (Declaration)	
Public Property Calendars As System.Collections.Generic.IList(Of Object)	
C#	
public System.Collections.Generic.IList<object> Calendars {get; set;}	

### See Also

#### Reference

[C1EventsCalendarXmlExchanger Class](#)

[C1EventsCalendarXmlExchanger Members](#)

## Events Property

### Syntax

Visual Basic (Declaration)	
Public Property Events As System.Collections.Generic.IList(Of Event)	

C#

```
public System.Collections.Generic.IList<Event> Events {get; set;}
```

## See Also

### Reference

[C1EventsCalendarXmlExchanger Class](#)
[C1EventsCalendarXmlExchanger Members](#)

## C1EventStorage

Allows binding to DataSource and mapping DataSource's fields to Event's properties.

## Object Model

C1EventStorage

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultPropertyAttribute("DataSourceID")>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
Public Class C1EventStorage
```

C#

```
[System.ComponentModel.DefaultProperty("DataSourceID")]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
public class C1EventStorage
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventStorage**

## See Also

### Reference

[C1EventStorage Members](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Allows binding to DataSource and mapping DataSource's fields to Event's properties.

## Object Model

**C1EventStorage**

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultPropertyAttribute("DataSourceID")>  
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>  
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,  
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>  
Public Class C1EventStorage
```

C#

```
[System.ComponentModel.DefaultProperty("DataSourceID")]  
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]  
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,  
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]  
public class C1EventStorage
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventStorage**

## See Also

### Reference


[C1EventStorage Members](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members

The following tables list the members exposed by [C1EventStorage](#).




## Public Constructors



	Name	Description
	<a href="#">C1EventStorage Constructor</a>	

[Top](#)

## Public Properties

	Name	Description
	<a href="#">DataMember</a>	The specific data member in a multimember data source.
	<a href="#">DataSourceID</a>	The ID of the control from which the events calendar control retrieves its list of data items.
	<a href="#">Mappings</a>	Use the Mappings property to bind the event object properties to the appropriate fields in the datasource specified by DataSourceID property.

[Top](#)

## See Also

### Reference

[C1EventStorage Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## C1EventStorage Constructor

### Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1EventStorage()	

## See Also




### Reference

[C1EventStorage Class](#)[C1EventStorage Members](#)

## Properties

For a list of all members of this type, see [C1EventStorage members](#).

### Public Properties

	Name	Description
	<a href="#">DataMember</a>	The specific data member in a multimember data source.
	<a href="#">DataSourceID</a>	The ID of the control from which the events calendar control retrieves its list of data items.
	<a href="#">Mappings</a>	Use the Mappings property to bind the event object properties to the appropriate fields in the datasource specified by DataSourceID property.

[Top](#)

### See Also

#### Reference

[C1EventStorage Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## DataMember Property

The specific data member in a multimember data source.

### Syntax

Visual Basic (Declaration)	
<pre> &lt;System.ComponentModel.DefaultValueAttribute()&gt; &lt;C1DescriptionAttribute(Key="C1EventsCalendar.DataStorage.C1EventStorage.Data Member", Description="The specific data member in a multimember data source.")&gt; &lt;System.ComponentModel.NotifyParentPropertyAttribute(True)&gt; &lt;C1CategoryAttribute("Data")&gt; &lt;System.ComponentModel.BindableAttribute(Bindable=False, Direction=BindingDirection.OneWay)&gt; &lt;EditorAttribute(EditorBaseTypeName="System.Drawing.Design.UITypeEditor, System.Drawing, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a", </pre>	

```
EditorTypeName="C1.Web.UI.Design.C1DataMemberEditor">
Public Property DataMember As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DataStorage.C1EventStorage.DataMember",
Description="The specific data member in a multimember data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[C1Category("Data")]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[Editor(EditorBaseTypeName="System.Drawing.Design.UITypeEditor,
System.Drawing, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a",
EditorTypeName="C1.Web.UI.Design.C1DataMemberEditor")]
public System.string DataMember {get; set;}
```

## See Also

### Reference

[C1EventStorage Class](#)

[C1EventStorage Members](#)

## DataSourceID Property

The ID of the control from which the events calendar control retrieves its list of data items.

## Syntax

Visual Basic (Declaration)

```
<EditorAttribute(EditorBaseTypeName="System.Drawing.Design.UITypeEditor,
System.Drawing, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a",
EditorTypeName="C1.Web.UI.Design.C1DataSourceIDEditor">
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1CategoryAttribute("Data")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DataStorage.C1EventStorage.Data
SourceID", Description="The ID of the control from which the events calendar
control retrieves its list of data items.")>
Public Property DataSourceID As System.String
```

C#

```
[Editor(EditorBaseTypeName="System.Drawing.Design.UITypeEditor,
System.Drawing, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a",
EditorTypeName="C1.Web.UI.Design.C1DataSourceIDEditor")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[C1Category("Data")]
[C1Description(Key="C1EventsCalendar.DataStorage.C1EventStorage.DataSourceID"
, Description="The ID of the control from which the events calendar control
retrieves its list of data items.")]
public System.string DataSourceID {get; set;}
```

## See Also

### Reference

[C1EventStorage Class](#)

[C1EventStorage Members](#)

## Mappings Property

Use the Mappings property to bind the event object properties to the appropriate fields in the datasource specified by DataSourceID property.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.DataStorage.C1EventStorage.Mapp
ings", Description="Use the Mappings property to bind the event object
properties to the appropriate fields in the datasource specified by
DataSourceID property.")>
<C1CategoryAttribute("Data")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Content)>
```

```
<EditorBrowsableAttribute(>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property Mappings As C1EventStorageMappings
```

C#

```
[C1Description(Key="C1EventsCalendar.DataStorage.C1EventStorage.Mappings",
Description="Use the Mappings property to bind the event object properties to
the appropriate fields in the datasource specified by DataSourceID
property.")]
[C1Category("Data")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
public C1EventStorageMappings Mappings {get;}
```

## See Also

### Reference

[C1EventStorage Class](#)

[C1EventStorage Members](#)

## C1EventStorageMappings

C1EventStorage mappings configuration.

## Object Model

[C1EventStorageMappings](#)

## Syntax

Visual Basic (Declaration)

```
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
```

```
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
Public Class C1EventStorageMappings
    Inherits C1MappingsBase
```

C#

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
public class C1EventStorageMappings : C1MappingsBase
```

## Inheritance Hierarchy

System.Object

[C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase](#)

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventStorageMappings**

## See Also

### Reference

[C1EventStorageMappings Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

C1EventStorage mappings configuration.

## Object Model

**C1EventStorageMappings**

## Syntax

Visual Basic (Declaration)

```
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
Public Class C1EventStorageMappings
    Inherits C1MappingsBase
```

C#

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
```

```
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]  
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]  
public class C1EventStorageMappings : C1MappingsBase
```

## Inheritance Hierarchy

System.Object

[C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase](#)

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventStorageMappings**

## See Also

### Reference


[C1EventStorageMappings Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members





The following tables list the members exposed by [C1EventStorageMappings](#).







### Public Constructors

	Name	Description
	<a href="#">C1EventStorageMappings Constructor</a>	

[Top](#)




### Public Properties

	Name	Description
	<a href="#">CalendarMapping</a>	Allows the calendar property to be bound to appropriate field in the data source.
	<a href="#">ColorMapping</a>	Allows the color property to be bound to appropriate field in the data source.
	<a href="#">DescriptionMapping</a>	Allows the description property to be bound to appropriate field in the data source.
	<a href="#">EndMapping</a>	Allows the end property to be bound to appropriate field in the data

		source.
	<a href="#">IdMapping</a>	(Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
	<a href="#">LocationMapping</a>	Allows the location property to be bound to appropriate field in the data source.
	<a href="#">PropertiesMapping</a>	Allows other event properties to be bound to appropriate field in the data source.
	<a href="#">StartMapping</a>	Allows the start property to be bound to appropriate field in the data source.
	<a href="#">SubjectMapping</a>	Allows the subject property to be bound to appropriate field in the data source.
	<a href="#">TagMapping</a>	Allows the tag property to be bound to appropriate field in the data source (the tag property can be used to store any user-defined information).

[Top](#)


## Public Methods

	Name	Description
	<a href="#">Dispose</a>	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
	<a href="#">FindByMappingName</a>	Finds the mapping info by mapping name (name of the data field). (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
	<a href="#">ToArray</a>	Creates an array of MappingInfo objects. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )

[Top](#)



## Public Events

	Name	Description
	<a href="#">Disposed</a>	Represents the method that handles the Disposed event of a component. (Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )

[Top](#)

## See Also

### Reference

[C1EventStorageMappings Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## C1EventStorageMappings Constructor

### Syntax

Visual Basic (Declaration)	
<pre>Public Function New( _     ByVal c1EventStorage As <a href="#">C1EventStorage</a> _ )</pre>	
C#	
<pre>public C1EventStorageMappings(     <a href="#">C1EventStorage</a> c1EventStorage )</pre>	

### Parameters

*c1EventStorage*

## See Also

### Reference











[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## Properties

For a list of all members of this type, see [C1EventStorageMappings members](#).

## Public Properties

	Name	Description
	<a href="#">CalendarMapping</a>	Allows the calendar property to be bound to appropriate field in the data source.
	<a href="#">ColorMapping</a>	Allows the color property to be bound to appropriate field in the data source.
	<a href="#">DescriptionMapping</a>	Allows the description property to be bound to appropriate field in the data source.
	<a href="#">EndMapping</a>	Allows the end property to be bound to appropriate field in the data source.
	<a href="#">IdMapping</a>	(Inherited from <a href="#">C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase</a> )
	<a href="#">LocationMapping</a>	Allows the location property to be bound to appropriate field in the data source.
	<a href="#">PropertiesMapping</a>	Allows other event properties to be bound to appropriate field in the data source.
	<a href="#">StartMapping</a>	Allows the start property to be bound to appropriate field in the data source.
	<a href="#">SubjectMapping</a>	Allows the subject property to be bound to appropriate field in the data source.
	<a href="#">TagMapping</a>	Allows the tag property to be bound to appropriate field in the data source (the tag property can be used to store any user-defined information).

[Top](#)

## See Also

### Reference

[C1EventStorageMappings Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## CalendarMapping Property

Allows the calendar property to be bound to appropriate field in the data source.

### Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.Calendar
Mapping", Description="Allows the calendar property to be bound to
appropriate field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property CalendarMapping As MappingInfo
```

C#

```
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.CalendarMapping",
Description="Allows the calendar property to be bound to appropriate field in
the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
public MappingInfo CalendarMapping {get;}
```

### See Also

## Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## ColorMapping Property

Allows the color property to be bound to appropriate field in the data source.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.ColorMapping", Description="Allows the color property to be bound to appropriate field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property ColorMapping As MappingInfo
```

C#

```
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.ColorMapping",
Description="Allows the color property to be bound to appropriate field in the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
public MappingInfo ColorMapping {get;}
```

## See Also

## Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## DescriptionMapping Property

Allows the description property to be bound to appropriate field in the data source.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.Descript
ionMapping", Description="Allows the description property to be bound to
appropriate field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property DescriptionMapping As MappingInfo
```

C#

```
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.DescriptionMappin
g", Description="Allows the description property to be bound to appropriate
field in the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
public MappingInfo DescriptionMapping {get;}
```

## See Also

## Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## EndMapping Property

Allows the end property to be bound to appropriate field in the data source.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.EndMapping",
Description="Allows the end property to be bound to appropriate field in
the data source.")>
<C1CategoryAttribute("Mappings")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
Public ReadOnly Property EndMapping As MappingInfo
```

C#

```
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.EndMapping",
Description="Allows the end property to be bound to appropriate field in the
data source.")]
[C1Category("Mappings")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
public MappingInfo EndMapping {get;}
```

## See Also

## Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## LocationMapping Property

Allows the location property to be bound to appropriate field in the data source.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.Location
Mapping", Description="Allows the location property to be bound to
appropriate field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property LocationMapping As MappingInfo
```

C#

```
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.LocationMapping",
Description="Allows the location property to be bound to appropriate field in
the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
public MappingInfo LocationMapping {get;}
```

## See Also

### Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## PropertiesMapping Property

Allows other event properties to be bound to appropriate field in the data source.

## Syntax

### Visual Basic (Declaration)

```
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.PropertiesMapping", Description="Allows other event properties to be bound to appropriate field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencyModeAttribute(PersistencyMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False, Direction=BindingDirection.OneWay)>
Public ReadOnly Property PropertiesMapping As MappingInfo
```

### C#

```
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.PropertiesMapping", Description="Allows other event properties to be bound to appropriate field in the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.PersistencyMode(PersistencyMode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
[System.ComponentModel.Bindable(Bindable=false,
```



```
Direction=BindingDirection.OneWay)]
public MappingInfo PropertiesMapping {get;}
```

## See Also

### Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## StartMapping Property

Allows the start property to be bound to appropriate field in the data source.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.StartMap
ping", Description="Allows the start property to be bound to appropriate
field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property StartMapping As MappingInfo
```

C#

```
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.StartMapping",
Description="Allows the start property to be bound to appropriate field in
the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
```

```
[System.ComponentModel.Browsable(true)]
public MappingInfo StartMapping {get;}
```

## See Also

### Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## SubjectMapping Property

Allows the subject property to be bound to appropriate field in the data source.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.SubjectM
apping", Description="Allows the subject property to be bound to appropriate
field in the data source.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
Public ReadOnly Property SubjectMapping As MappingInfo
```

C#

```
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.SubjectMapping",
Description="Allows the subject property to be bound to appropriate field in
the data source.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[EditorBrowsable()]
[System.ComponentModel.Browsable(true)]
[System.ComponentModel.Bindable(Bindable=false,
```

```
Direction=BindingDirection.OneWay)]
public MappingInfo SubjectMapping {get;}
```

## See Also

### Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## TagMapping Property

Allows the tag property to be bound to appropriate field in the data source (the tag property can be used to store any user-defined information).

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<C1CategoryAttribute("Mappings")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1EventStorageMappings.TagMapping",
Description="Allows the tag property to be bound to appropriate field in
the data source. (the tag property can be used to store any user-defined
information)")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<EditorBrowsableAttribute()>
<System.ComponentModel.BrowsableAttribute(True)>
Public ReadOnly Property TagMapping As MappingInfo
```

C#

```
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[C1Category("Mappings")]
[C1Description(Key="C1EventsCalendar.C1EventStorageMappings.TagMapping",
Description="Allows the tag property to be bound to appropriate field in the
data source. (the tag property can be used to store any user-defined
information)")]
[System.ComponentModel.NotifyParentProperty(true)]
```

```
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]  
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]  
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,  
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]  
[EditorBrowsable()]  
[System.ComponentModel.Browsable(true)]  
public MappingInfo TagMapping {get;}
```

## See Also

### Reference

[C1EventStorageMappings Class](#)

[C1EventStorageMappings Members](#)

## C1MappingsBase

Represents a collection of mappings for properties to the appropriate data fields. Names of properties are keys in this collection.

## Object Model

C1MappingsBase

## Syntax

Visual Basic (Declaration)

```
Public Class C1MappingsBase
```

C#

```
public class C1MappingsBase
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase**

[C1.Web.Wijmo.Controls.C1EventsCalendar.C1CalendarStorageMappings](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventStorageMappings](#)

## See Also

### Reference

[C1MappingsBase Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Represents a collection of mappings for properties to the appropriate data fields. Names of properties are keys in this collection.

## Object Model

C1MappingsBase

## Syntax

Visual Basic (Declaration)	
Public Class C1MappingsBase	
C#	
public class C1MappingsBase	

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1MappingsBase**

[C1.Web.Wijmo.Controls.C1EventsCalendar.C1CalendarStorageMappings](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventStorageMappings](#)

## See Also

### Reference

[C1MappingsBase Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members

The following tables list the members exposed by [C1MappingsBase](#).

### Public Constructors

	Name	Description
≡	<a href="#">C1MappingsBase Constructor</a>	Overloaded.




[Top](#)

### Public Properties

	Name	Description
	<a href="#">IdMapping</a>	


[Top](#)

## Public Methods

	Name	Description
	<a href="#">Dispose</a>	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	<a href="#">FindByMappingName</a>	Finds the mapping info by mapping name (name of the data field).
	<a href="#">ToArray</a>	Creates an array of MappingInfo objects.

[Top](#)

## Public Events

	Name	Description
	<a href="#">Disposed</a>	Represents the method that handles the Disposed event of a component.

[Top](#)

## See Also

### Reference

[C1MappingsBase Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## C1MappingsBase Constructor

### Overload List

Overload	Description
<a href="#">C1MappingsBase Constructor()</a>	Initializes a new instance of the <a href="#">C1MappingsBase</a> class.

<a href="#">C1MappingsBase Constructor(StateBag)</a>	Initializes a new instance of the <b>MappingCollectionBase&lt;T&gt;</b> class.

## See Also

### Reference

[C1MappingsBase Class](#)

[C1MappingsBase Members](#)

## C1MappingsBase Constructor()

Initializes a new instance of the [C1MappingsBase](#) class.

## Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1MappingsBase()	

## See Also

### Reference

[C1MappingsBase Class](#)

[C1MappingsBase Members](#)

[Overload List](#)

## C1MappingsBase Constructor(StateBag)

Initializes a new instance of the **MappingCollectionBase<T>** class.

## Syntax

Visual Basic (Declaration)	
Public Function New( _ ByVal <i>parentStateBag</i> As System.Web.UI.StateBag _ )	

C#

```
public C1MappingsBase(  
    System.Web.UI.StateBag parentStateBag  
)
```

### Parameters

*parentStateBag*

## See Also

### Reference

[C1MappingsBase Class](#)




[C1MappingsBase Members](#)

[Overload List](#)

## Methods

For a list of all members of this type, see [C1MappingsBase members](#).

## Public Methods

	Name	Description
	<a href="#">Dispose</a>	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	<a href="#">FindByMappingName</a>	Finds the mapping info by mapping name (name of the data field).
	<a href="#">ToArray</a>	Creates an array of MappingInfo objects.

[Top](#)

## See Also

### Reference

[C1MappingsBase Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Dispose Method

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.



## Syntax

Visual Basic (Declaration)	
Public Overridable Sub Dispose()	
C#	
public virtual void Dispose()	

## See Also

### Reference

[C1MappingsBase Class](#)

[C1MappingsBase Members](#)

## FindByMappingName Method

Finds the mapping info by mapping name (name of the data field).

## Syntax

Visual Basic (Declaration)	
Public Function FindByMappingName( _ ByVal <i>mappingName</i> As System.String _ ) As <a href="#">MappingInfo</a>	
C#	
public <a href="#">MappingInfo</a> FindByMappingName( System.string <i>mappingName</i> )	

### Parameters

*mappingName*

Name of the mapping.

## See Also

### Reference

[C1MappingsBase Class](#)

[C1MappingsBase Members](#)

## ToArray Method

Creates an array of MappingInfo objects.

### Syntax

Visual Basic (Declaration)	
Public Function ToArray() As MappingInfo()	
C#	
public MappingInfo[] ToArray()	

### See Also

#### Reference

[C1MappingsBase Class](#)

[C1MappingsBase Members](#)

## Properties

For a list of all members of this type, see [C1MappingsBase members](#).

### Public Properties

	Name	Description
	<a href="#">IdMapping</a>	

[Top](#)

### See Also

#### Reference

[C1MappingsBase Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## IdMapping Property

### Syntax

Visual Basic (Declaration)	
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>	

```
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Content)>
<C1CategoryAttribute("Mapping")>
<C1DescriptionAttribute(Key="C1EventsCalendar.C1MappingsBase.Id",
Description="Allows the id property to be bound to appropriate field in the
data source.")>
Public ReadOnly Property IdMapping As MappingInfo
```

C#

```
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Content)]
[C1Category("Mapping")]
[C1Description(Key="C1EventsCalendar.C1MappingsBase.Id", Description="Allows
the id property to be bound to appropriate field in the data source.")]
public MappingInfo IdMapping {get;}
```

## See Also

### Reference


[C1MappingsBase Class](#)

[C1MappingsBase Members](#)

## Events

For a list of all members of this type, see [C1MappingsBase members](#).

## Public Events

	Name	Description
	<a href="#">Disposed</a>	Represents the method that handles the Disposed event of a component.

[Top](#)

## See Also

### Reference

[C1MappingsBase Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Disposed Event

Represents the method that handles the Disposed event of a component.

### Syntax

Visual Basic (Declaration)	
Public Event Disposed As System.EventHandler	
C#	
public event System.EventHandler Disposed	

### See Also

#### Reference

[C1MappingsBase Class](#)

[C1MappingsBase Members](#)

## DatePagerLocalizationOption

Date pager localization strings.

### Object Model

DatePagerLocalizationOption

### Syntax

Visual Basic (Declaration)	
Public Class DatePagerLocalizationOption Inherits C1.Web.Wijmo.Controls.Settings	
C#	
public class DatePagerLocalizationOption : C1.Web.Wijmo.Controls.Settings	

### Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.DatePagerLocalizationOption**

### See Also

#### Reference

[DatePagerLocalizationOption Members](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Date pager localization strings.

## Object Model

**DatePagerLocalizationOption**

## Syntax

Visual Basic (Declaration)

```
Public Class DatePagerLocalizationOption
    Inherits C1.Web.Wijmo.Controls.Settings
```

C#

```
public class DatePagerLocalizationOption : C1.Web.Wijmo.Controls.Settings
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.DatePagerLocalizationOption**

## See Also


### Reference

[DatePagerLocalizationOption Members](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members

The following tables list the members exposed by [DatePagerLocalizationOption](#).

### Public Constructors

	Name	Description
	<a href="#">DatePagerLocalizationOption Constructor</a>	Initializes a new instance of the <a href="#">LocalizationOption</a> class.

[Top](#)


## Public Properties

	Name	Description
	<a href="#">customViewDayTooltipFormat</a>	customViewDayTooltipFormat string.
	<a href="#">customViewDayUnitTooltip2MonthesFormat</a>	customViewDayUnitTooltip2MonthesFormat string.
	<a href="#">customViewMonthTooltip2YearsFormat</a>	customViewMonthTooltip2YearsFormat string.
	<a href="#">customViewMonthTooltipFormat</a>	customViewMonthTooltipFormat string.
	<a href="#">customViewSingleDayTooltipFormat</a>	customViewSingleDayTooltipFormat string.
	<a href="#">customViewSingleMonthTooltipFormat</a>	customViewSingleMonthTooltipFormat string.
	<a href="#">customViewSingleYearTooltipFormat</a>	customViewSingleYearTooltipFormat string.
	<a href="#">customViewUnitDayLabelFormat</a>	customViewUnitDayLabelFormat string.
	<a href="#">customViewUnitDayLabelFormat2Months</a>	customViewUnitDayLabelFormat2Months string.
	<a href="#">customViewUnitMonthLabelFormat</a>	customViewUnitMonthLabelFormat string.
	<a href="#">customViewUnitMonthLabelFormat2Years</a>	customViewUnitMonthLabelFormat2Years string.
	<a href="#">customViewUnitSingleDayLabelFormat</a>	customViewUnitSingleDayLabelFormat string.
	<a href="#">customViewUnitSingleMonthLabelFormat</a>	customViewUnitSingleMonthLabelFormat string.
	<a href="#">customViewUnitSingleYearLabelFormat</a>	customViewUnitSingleYearLabelFormat

		string.
	<a href="#">customViewUnitYearLabelFormat</a>	customViewUnitYearLabelFormat string.
	<a href="#">customViewYearTooltipFormat</a>	customViewYearTooltipFormat string.
	<a href="#">dayViewLabelFormat</a>	dayViewLabelFormat string.
	<a href="#">dayViewTooltipFormat</a>	dayViewTooltipFormat string.
	<a href="#">monthViewLabelFormat</a>	monthViewLabelFormat string.
	<a href="#">monthViewTooltipFormat</a>	monthViewTooltipFormat string.
	<a href="#">weekViewLabelFormat</a>	weekViewLabelFormat string.
	<a href="#">weekViewTooltip2MonthesFormat</a>	weekViewTooltip2MonthesFormat string.
	<a href="#">weekViewTooltipFormat</a>	weekViewTooltipFormat string.

[Top](#)

## Public Methods

	Name	Description
	<a href="#">ToString</a>	Returns a <b>System.String</b> that represents this instance.

[Top](#)

## See Also

### Reference

[DatePagerLocalizationOption Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## DatePagerLocalizationOption Constructor

Initializes a new instance of the [LocalizationOption](#) class.

## Syntax

## Visual Basic (Declaration)

```
Public Function New( _  
    ByVal evCal As C1EventsCalendar _  
)
```

## C#

```
public DatePagerLocalizationOption(  
    C1EventsCalendar evCal  
)
```

## Parameters

*evCal*

## See Also

## Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## Methods

For a list of all members of this type, see [DatePagerLocalizationOption members](#).

## Public Methods

	Name	Description
	<a href="#">ToString</a>	Returns a <b>System.String</b> that represents this instance.

[Top](#)

## See Also

## Reference

[DatePagerLocalizationOption Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## ToString Method

Returns a **System.String** that represents this instance.

## Syntax



Visual Basic (Declaration)	
Public Overrides Function ToString() As System.String	
C#	
public override System.string ToString()	

### Return Value

A **System.String** that represents this instance.

## See Also

### Reference








[DatePagerLocalizationOption Class](#)
















[DatePagerLocalizationOption Members](#)

## Properties

For a list of all members of this type, see [DatePagerLocalizationOption members](#).

## Public Properties

	Name	Description
	<a href="#">customViewDayTooltipFormat</a>	customViewDayTooltipFormat string.
	<a href="#">customViewDayUnitTooltip2MonthesFormat</a>	customViewDayUnitTooltip2MonthesFormat string.
	<a href="#">customViewMonthTooltip2YearsFormat</a>	customViewMonthTooltip2YearsFormat string.
	<a href="#">customViewMonthTooltipFormat</a>	customViewMonthTooltipFormat string.
	<a href="#">customViewSingleDayTooltipFormat</a>	customViewSingleDayTooltipFormat string.
	<a href="#">customViewSingleMonthTooltipFormat</a>	customViewSingleMonthTooltipFormat string.
	<a href="#">customViewSingleYearTooltipFormat</a>	customViewSingleYearTooltipFormat string.

	<a href="#">customViewUnitDayLabelFormat</a>	customViewUnitDayLabelFormat string.
	<a href="#">customViewUnitDayLabelFormat2Months</a>	customViewUnitDayLabelFormat2Months string.
	<a href="#">customViewUnitMonthLabelFormat</a>	customViewUnitMonthLabelFormat string.
	<a href="#">customViewUnitMonthLabelFormat2Years</a>	customViewUnitMonthLabelFormat2Years string.
	<a href="#">customViewUnitSingleDayLabelFormat</a>	customViewUnitSingleDayLabelFormat string.
	<a href="#">customViewUnitSingleMonthLabelFormat</a>	customViewUnitSingleMonthLabelFormat string.
	<a href="#">customViewUnitSingleYearLabelFormat</a>	customViewUnitSingleYearLabelFormat string.
	<a href="#">customViewUnitYearLabelFormat</a>	customViewUnitYearLabelFormat string.
	<a href="#">customViewYearTooltipFormat</a>	customViewYearTooltipFormat string.
	<a href="#">dayViewLabelFormat</a>	dayViewLabelFormat string.
	<a href="#">dayViewTooltipFormat</a>	dayViewTooltipFormat string.
	<a href="#">monthViewLabelFormat</a>	monthViewLabelFormat string.
	<a href="#">monthViewTooltipFormat</a>	monthViewTooltipFormat string.
	<a href="#">weekViewLabelFormat</a>	weekViewLabelFormat string.
	<a href="#">weekViewTooltip2MonthesFormat</a>	weekViewTooltip2MonthesFormat string.
	<a href="#">weekViewTooltipFormat</a>	weekViewTooltipFormat string.

[Top](#)

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## customViewDayTooltipFormat Property

customViewDayTooltipFormat string.

## Syntax

### Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewDayTooltipFormat", Description="Date format for custom multiple day view tooltip.")>
<C1CategoryAttribute("Appearance")>
Public Property customViewDayTooltipFormat As System.String
```

### C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.NotifyParentProperty(true)]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewDayTooltipFormat", Description="Date format for custom multiple day view tooltip.")]
[C1Category("Appearance")]
public System.string customViewDayTooltipFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewDayUnitTooltip2MonthesFormat Property

customViewDayUnitTooltip2MonthesFormat string.

## Syntax

### Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewDayUnitTooltip2MonthesFormat", Description="Date format for custom day view tooltip between 2 months.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property customViewDayUnitTooltip2MonthesFormat As System.String
```

### C#

```
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
[C1Category("Appearance")]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewDayUnitTooltip2MonthesFormat", Description="Date format for custom day view tooltip between 2 months.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string customViewDayUnitTooltip2MonthesFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewMonthTooltip2YearsFormat Property

customViewMonthTooltip2YearsFormat string.

## Syntax

### Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
```

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewMonthTooltip2YearsFormat", Description="Date format for custom month view tooltip between 2 years.")>
Public Property customViewMonthTooltip2YearsFormat As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[C1Category("Appearance")]
[WidgetOption()]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewMonthTooltip2YearsFormat", Description="Date format for custom month view tooltip between 2 years.")]
public System.string customViewMonthTooltip2YearsFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewMonthTooltipFormat Property

customViewMonthTooltipFormat string.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewMonthTooltipFormat", Description="Date format for custom multiple month view tooltip.")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property customViewMonthTooltipFormat As System.String
```

C#

```
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewMonthTooltipFormat", Description="Date format for custom multiple month view tooltip.")]
[Layout(LayoutType.Appearance)]
public System.string customViewMonthTooltipFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewSingleDayTooltipFormat Property

customViewSingleDayTooltipFormat string.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewSingleDayTooltipFormat", Description="Date format for custom single day view tooltip.")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
Public Property customViewSingleDayTooltipFormat As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewSingleDayTooltipFormat", Description="Date format for custom single day view tooltip.")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
```

```
public System.string customViewSingleDayTooltipFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewSingleMonthTooltipFormat Property

customViewSingleMonthTooltipFormat string.

## Syntax

### Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewSingleMonthTooltipFormat", Description="Date format for custom single month view tooltip.")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property customViewSingleMonthTooltipFormat As System.String
```

### C#

```
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewSingleMonthTooltipFormat", Description="Date format for custom single month view tooltip.")]
[WidgetOption()]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string customViewSingleMonthTooltipFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewSingleYearTooltipFormat Property

customViewSingleYearTooltipFormat string.

### Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewSingleYearTooltipFormat", Description="Date format for custom single year view tooltip.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Property customViewSingleYearTooltipFormat As System.String
```

C#

```
[C1Category("Appearance")]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewSingleYearTooltipFormat", Description="Date format for custom single year view tooltip.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
public System.string customViewSingleYearTooltipFormat {get; set;}
```

### See Also

#### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewUnitDayLabelFormat Property

customViewUnitDayLabelFormat string.

### Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
```



```

<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitDayLabelFormat", Description="Date format for custom multiple day view label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property customViewUnitDayLabelFormat As System.String

```

C#

```

[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitDayLabelFormat", Description="Date format for custom multiple day view label.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string customViewUnitDayLabelFormat {get; set;}

```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewUnitDayLabelFormat2Months Property

customViewUnitDayLabelFormat2Months string.

## Syntax

Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitDayLabelFormat2Months", Description="Date format for custom day view label between 2 months.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property customViewUnitDayLabelFormat2Months As System.String

```

C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitDayLabelFormat2Months", Description="Date format for custom day view label between 2 months.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string customViewUnitDayLabelFormat2Months {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewUnitMonthLabelFormat Property

customViewUnitMonthLabelFormat string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitMonthLabelFormat", Description="Date format for custom multiple months view label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property customViewUnitMonthLabelFormat As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitMonthLabelFormat", Description="Date format for custom multiple months view label.")]
```

```
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string customViewUnitMonthLabelFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewUnitMonthLabelFormat2Years Property

customViewUnitMonthLabelFormat2Years string.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitMonthLabelFormat2Years", Description="Date format for custom month view label between 2 years.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property customViewUnitMonthLabelFormat2Years As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitMonthLabelFormat2Years", Description="Date format for custom month view label between 2 years.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string customViewUnitMonthLabelFormat2Years {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)[DatePagerLocalizationOption Members](#)

## customViewUnitSingleDayLabelFormat Property

customViewUnitSingleDayLabelFormat string.

### Syntax

Visual Basic (Declaration)

```

<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitSingleDayLabelFormat", Description="Date format for custom single day view label.")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property customViewUnitSingleDayLabelFormat As System.String

```

C#

```

[System.ComponentModel.NotifyParentProperty(true)]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitSingleDayLabelFormat", Description="Date format for custom single day view label.")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string customViewUnitSingleDayLabelFormat {get; set;}

```

### See Also

#### Reference

[DatePagerLocalizationOption Class](#)[DatePagerLocalizationOption Members](#)

## customViewUnitSingleMonthLabelFormat Property

customViewUnitSingleMonthLabelFormat string.

### Syntax

## Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitSingleMonthLabelFormat", Description="Date format for custom single month view label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property customViewUnitSingleMonthLabelFormat As System.String
```

## C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitSingleMonthLabelFormat", Description="Date format for custom single month view label.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string customViewUnitSingleMonthLabelFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewUnitSingleYearLabelFormat Property

customViewUnitSingleYearLabelFormat string.

## Syntax

## Visual Basic (Declaration)

```
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitSingleYearLabelFormat", Description="Date format for custom single year view label.")>
```

```
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property customViewUnitSingleYearLabelFormat As System.String
```

C#

```
[System.ComponentModel.NotifyParentProperty(true)]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitSingleYearLabelFormat", Description="Date format for custom single year view label.")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string customViewUnitSingleYearLabelFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewUnitYearLabelFormat Property

customViewUnitYearLabelFormat string.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitYearLabelFormat", Description="Date format for custom multiple year view label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<WidgetOptionAttribute()>
Public Property customViewUnitYearLabelFormat As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[C1Category("Appearance")]
```

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewUnitYearLabelFormat", Description="Date format for custom multiple year view label.")]
[System.ComponentModel.NotifyParentProperty(true)]
[WidgetOption()]
public System.string customViewUnitYearLabelFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## customViewYearTooltipFormat Property

customViewYearTooltipFormat string.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewYearTooltipFormat", Description="Date format for custom multiple year view tooltip.")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property customViewYearTooltipFormat As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.customViewYearTooltipFormat", Description="Date format for custom multiple year view tooltip.")]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string customViewYearTooltipFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## dayViewLabelFormat Property

dayViewLabelFormat string.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.day
ViewLabelFormat", Description="Date format for day view label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property dayViewLabelFormat As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.dayViewLabel
Format", Description="Date format for day view label.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string dayViewLabelFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## dayViewTooltipFormat Property

dayViewTooltipFormat string.

## Syntax



## Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.day
ViewTooltipFormat", Description="Date format for day view tooltip.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
Public Property dayViewTooltipFormat As System.String

```

## C#

```

[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.dayViewToolt
ipFormat", Description="Date format for day view tooltip.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
public System.string dayViewTooltipFormat {get; set;}

```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## monthViewLabelFormat Property

monthViewLabelFormat string.

## Syntax

## Visual Basic (Declaration)

```

<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.mon
thViewLabelFormat", Description="Date format for month view label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property monthViewLabelFormat As System.String

```

C#

```
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.monthViewLabelFormat", Description="Date format for month view label.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string monthViewLabelFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## monthViewTooltipFormat Property

monthViewTooltipFormat string.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.monthViewTooltipFormat", Description="Date format for month view tooltip.")>
Public Property monthViewTooltipFormat As System.String
```

C#

```
[C1Category("Appearance")]
[WidgetOption()]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.monthViewTooltipFormat", Description="Date format for month view tooltip.")]
public System.string monthViewTooltipFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## weekViewLabelFormat Property

weekViewLabelFormat string.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.weekViewLabelFormat", Description="Date format for week view label.")>
Public Property weekViewLabelFormat As System.String
```

C#

```
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.weekViewLabelFormat", Description="Date format for week view label.")]
public System.string weekViewLabelFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## weekViewTooltip2MonthesFormat Property

weekViewTooltip2MonthesFormat string.

## Syntax

## Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.weekViewTooltip2MonthesFormat", Description="Date format for week view tooltip between 2 months.")>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property weekViewTooltip2MonthesFormat As System.String
```

## C#

```
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.weekViewTooltip2MonthesFormat", Description="Date format for week view tooltip between 2 months.")]
[C1Category("Appearance")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string weekViewTooltip2MonthesFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)[DatePagerLocalizationOption Members](#)

## weekViewTooltipFormat Property

weekViewTooltipFormat string.

## Syntax

## Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DatePagerLocalizationOption.weekViewTooltipFormat", Description="Date format for week view tooltip.")>
```

```
<LayoutAttribute(LayoutType.Appearance)>
Public Property weekViewTooltipFormat As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DatePagerLocalizationOption.weekViewTool
tipFormat", Description="Date format for week view tooltip.")]
[Layout(LayoutType.Appearance)]
public System.string weekViewTooltipFormat {get; set;}
```

## See Also

### Reference

[DatePagerLocalizationOption Class](#)

[DatePagerLocalizationOption Members](#)

## DayViewHeaderFormatOption

Format of the text for the day header in the day/week or list view.

## Object Model

DayViewHeaderFormatOption

## Syntax

Visual Basic (Declaration)

```
Public Class DayViewHeaderFormatOption
    Inherits C1.Web.Wijmo.Controls.Settings
```

C#

```
public class DayViewHeaderFormatOption : C1.Web.Wijmo.Controls.Settings
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.DayViewHeaderFormatOption**

## See Also

## Reference

[DayViewHeaderFormatOption Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Format of the text for the day header in the day/week or list view.

## Object Model

DayViewHeaderFormatOption

## Syntax

Visual Basic (Declaration)

```
Public Class DayViewHeaderFormatOption
    Inherits C1.Web.Wijmo.Controls.Settings
```

C#

```
public class DayViewHeaderFormatOption : C1.Web.Wijmo.Controls.Settings
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.DayViewHeaderFormatOption**

## See Also

### Reference


[DayViewHeaderFormatOption Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members





The following tables list the members exposed by [DayViewHeaderFormatOption](#).

### Public Constructors

	Name	Description
	<a href="#">DayViewHeaderFormatOption Constructor</a>	Initializes a new instance of the <a href="#">DayViewHeaderFormatOption</a> class.


[Top](#)

## Public Properties

	Name	Description
	<a href="#">Custom</a>	Format of the text for the custom header in the custom view.
	<a href="#">Day</a>	Format of the text for the day header in the day view.
	<a href="#">List</a>	Format of the text for the day header in the list view.
	<a href="#">Week</a>	Format of the text for the day header in the week view.

[Top](#)

## Public Methods

	Name	Description
	<a href="#">ToString</a>	Returns a <b>System.String</b> that represents this instance.

[Top](#)

## See Also

### Reference

[DayViewHeaderFormatOption Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## DayViewHeaderFormatOption Constructor

Initializes a new instance of the [DayViewHeaderFormatOption](#) class.

## Syntax

Visual Basic (Declaration)	
<pre>Public Function New( _     ByVal evCal As C1EventsCalendar _ )</pre>	
C#	

```
public DayViewHeaderFormatOption(  
    C1EventsCalendar evCal  
)
```

### Parameters

*evCal*

## See Also

### Reference

[DayViewHeaderFormatOption Class](#)

[DayViewHeaderFormatOption Members](#)

## Methods

For a list of all members of this type, see [DayViewHeaderFormatOption members](#).

## Public Methods

	Name	Description
	<a href="#">ToString</a>	Returns a <b>System.String</b> that represents this instance.

[Top](#)

## See Also

### Reference

[DayViewHeaderFormatOption Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## ToString Method

Returns a **System.String** that represents this instance.

## Syntax

Visual Basic (Declaration)	
Public Overrides Function ToString() As System.String	
C#	
public override System.string ToString()	

### Return Value



A **System.String** that represents this instance.

## See Also

### Reference





[DayViewHeaderFormatOption Class](#)

[DayViewHeaderFormatOption Members](#)

## Properties

For a list of all members of this type, see [DayViewHeaderFormatOption members](#).

## Public Properties

	Name	Description
	<a href="#">Custom</a>	Format of the text for the custom header in the custom view.
	<a href="#">Day</a>	Format of the text for the day header in the day view.
	<a href="#">List</a>	Format of the text for the day header in the list view.
	<a href="#">Week</a>	Format of the text for the day header in the week view.

[Top](#)

## See Also

### Reference

[DayViewHeaderFormatOption Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Custom Property

Format of the text for the custom header in the custom view.

## Syntax

Visual Basic (Declaration)	
<pre>&lt;WidgetOptionAttribute()&gt; &lt;System.ComponentModel.DefaultValueAttribute()&gt; &lt;C1CategoryAttribute("Appearance")&gt;</pre>	

```
<C1DescriptionAttribute(Key="C1EventsCalendar.DayViewHeaderFormatOption.Custo
m", Description="Format of the text for the custom header in the custom
view.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Custom As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DayViewHeaderFormatOption.Custom",
Description="Format of the text for the custom header in the custom view.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string Custom {get; set;}
```

## See Also

### Reference

[DayViewHeaderFormatOption Class](#)

[DayViewHeaderFormatOption Members](#)

## Day Property

Format of the text for the day header in the day view.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DayViewHeaderFormatOption.Day",
Description="Format of the text for the day header in the day view.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Day As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
```

```
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DayViewHeaderFormatOption.Day",
Description="Format of the text for the day header in the day view.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string Day {get; set;}
```

## See Also

### Reference

[DayViewHeaderFormatOption Class](#)

[DayViewHeaderFormatOption Members](#)

## List Property

Format of the text for the day header in the list view.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.DayViewHeaderFormatOption.List"
, Description="Format of the text for the day header in the list view.")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property List As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.DayViewHeaderFormatOption.List",
Description="Format of the text for the day header in the list view.")]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[WidgetOption()]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string List {get; set;}
```

## See Also

### Reference

[DayViewHeaderFormatOption Class](#)[DayViewHeaderFormatOption Members](#)

## Week Property

Format of the text for the day header in the week view.

### Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.DayViewHeaderFormatOption.Week"
, Description="Format of the text for the day header in the week view.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Week As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.DayViewHeaderFormatOption.Week",
Description="Format of the text for the day header in the week view.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string Week {get; set;}
```

### See Also

#### Reference

[DayViewHeaderFormatOption Class](#)[DayViewHeaderFormatOption Members](#)

## Event

The [Event](#) object is a meeting, a one-time event, or a recurring event or meeting in the C1EventsCalendar. The [Event](#) class includes properties that specify meeting details such as the location and time.

### Object Model

## Event

## Syntax

Visual Basic (Declaration)	
Public Class Event	
C#	
public class Event	

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.Event**

## See Also

### Reference

[Event Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

The [Event](#) object is a meeting, a one-time event, or a recurring event or meeting in the C1EventsCalendar. The [Event](#) class includes properties that specify meeting details such as the location and time.

## Object Model

## Event

## Syntax

Visual Basic (Declaration)	
Public Class Event	
C#	
public class Event	

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.Event**

## See Also

### Reference

[Event Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members

The following tables list the members exposed by [Event](#).

### Public Constructors

	Name	Description
	<a href="#">Event Constructor</a>	Overloaded.

[Top](#)














### Public Fields


	Name	Description
	<a href="#">_allDayEvent</a>	
	<a href="#">_color</a>	
	<a href="#">_description</a>	
	<a href="#">_end</a>	
	<a href="#">_start</a>	
	<a href="#">_subject</a>	

[Top](#)

### Public Properties





	Name	Description
--	------	-------------

	Allday	indicates all day event.
	Calendar	Calendar id to which the event belongs.
	Color	Event color.
	Data	Gets or sets the raw event data.
	Description	Event description.
	End	End date/time.
	Id	The unique event id (this field generated automatically).
	Location	Event location.
	ParentRecurrenceId	id of the event object that defines the recurrence criteria for this event object. If an event is recurring (see isRecurring) it represents an occurrence in the series that is started and defined by a specific pattern event. Use the getPattern method in order to obtain the pattern of the current event. A pattern event can be recognized by its recurrenceState field set to the "master" value. The recurrence information defined by the pattern event can be accessed via the recurrencePattern field of the event. If this event is a not member of a recurrence, or is itself a root event, this field will be null.
	RecurrencePattern	Represents the recurrence attributes of an event. Only master events can have this field (recurrenceState is "master")
	RecurrenceState	Indicates the recurrence state of the event. (possible values are "notRecurring"(or null), "master", "occurrence", "exception", "removed")
	Start	Start date/time.
	Subject	Event title.

	<a href="#">Tag</a>	The tag property can be used to store any user-defined information
---	---------------------	--

[Top](#)

## Public Methods

	Name	Description
	<a href="#">CopyFrom</a>	Copies properties from event given by parameter ev.
	<a href="#">FromXml</a>	Reconstructs event from an <b>System.Xml.XmlReader</b> .
	<a href="#">PropsToXml</a>	Creates an XML encoding of the event properties and returns it as string.
	<a href="#">ToXml</a>	Creates an XML encoding of the event.

[Top](#)

## See Also

### Reference

[Event Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Event Constructor

### Overload List

Overload	Description
<a href="#">Event Constructor()</a>	Initializes a new instance of the <a href="#">Event</a> class.
<a href="#">Event Constructor(Hashtable)</a>	Initializes a new instance of the <a href="#">Event</a> class.

## See Also

### Reference

[Event Class](#)[Event Members](#)

## Event Constructor()

Initializes a new instance of the [Event](#) class.



## Syntax

Visual Basic (Declaration)	
<code>Public Function New()</code>	
C#	
<code>public Event()</code>	

## See Also

### Reference

[Event Class](#)[Event Members](#)[Overload List](#)

## Event Constructor(Hashtable)

Initializes a new instance of the [Event](#) class.

## Syntax

Visual Basic (Declaration)	
<code>Public Function New( _     ByVal ht As System.Collections.Hashtable _ )</code>	
C#	
<code>public Event(     System.Collections.Hashtable ht )</code>	

### Parameters

*ht*

## See Also





### Reference

[Event Class](#)[Event Members](#)[Overload List](#)

## Methods

For a list of all members of this type, see [Event members](#).

### Public Methods

	Name	Description
	<a href="#">CopyFrom</a>	Copies properties from event given by parameter ev.
	<a href="#">FromXml</a>	Reconstructs event from an <b>System.Xml.XmlReader</b> .
	<a href="#">PropsToXml</a>	Creates an XML encoding of the event properties and returns it as string.
	<a href="#">ToXml</a>	Creates an XML encoding of the event.

[Top](#)

### See Also

#### Reference

[Event Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## CopyFrom Method

Copies properties from event given by parameter ev.

### Syntax

Visual Basic (Declaration)

```
Public Sub CopyFrom( _  
    ByVal ev As Event _  
)
```

C#

```
public void CopyFrom(  
    Event ev  
)
```

#### Parameters

ev

The event.

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## FromXml Method

Reconstructs event from an **System.Xml.XmlReader**.

## Syntax

Visual Basic (Declaration)	
Public Overridable Function FromXml( _ ByVal <i>reader</i> As System.Xml.XmlReader _ ) As System.Boolean	
C#	
public virtual System.bool FromXml( System.Xml.XmlReader <i>reader</i> )	

### Parameters

*reader*

An **System.Xml.XmlReader** which contains the **Appointment** data.

### Return Value

True if event has been changed.

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## PropsToXml Method

Creates an XML encoding of the event properties and returns it as string.

## Syntax

Visual Basic (Declaration)	
Public Function PropsToXml() As System.String	
C#	
public System.string PropsToXml()	

## See Also

### Reference

[Event Class](#)[Event Members](#)

## ToXml Method

Creates an XML encoding of the event.

## Syntax

Visual Basic (Declaration)	
Public Overridable Sub ToXml( _ ByVal <i>writer</i> As System.Xml.XmlWriter _ )	
C#	
public virtual void ToXml( System.Xml.XmlWriter <i>writer</i> )	

### Parameters

*writer*

The **System.Xml.XmlWriter** that will receive the object data.

## See Also













### Reference



[Event Class](#)[Event Members](#)

## Properties

For a list of all members of this type, see [Event members](#).

## Public Properties

	Name	Description
	<a href="#">Allday</a>	indicates all day event.
	<a href="#">Calendar</a>	Calendar id to which the event belongs.
	<a href="#">Color</a>	Event color.
	<a href="#">Data</a>	Gets or sets the raw event data.
	<a href="#">Description</a>	Event description.
	<a href="#">End</a>	End date/time.
	<a href="#">Id</a>	The unique event id (this field generated automatically).
	<a href="#">Location</a>	Event location.
	<a href="#">ParentRecurrenceId</a>	id of the event object that defines the recurrence criteria for this event object. If an event is recurring (see isRecurring) it represents an occurrence in the series that is started and defined by a specific pattern event. Use the getPattern method in order to obtain the pattern of the current event. A pattern event can be recognized by its recurrenceState field set to the "master" value. The recurrence information defined by the pattern event can be accessed via the recurrencePattern field of the event. If this event is a not member of a recurrence, or is itself a root event, this field will be null.
	<a href="#">RecurrencePattern</a>	Represents the recurrence attributes of an event. Only master events can have this field (recurrenceState is "master")
	<a href="#">RecurrenceState</a>	Indicates the recurrence state of the event. (possible values are "notRecurring"(or null), "master", "occurrence", "exception", "removed")
	<a href="#">Start</a>	Start date/time.

	<b>Subject</b>	Event title.
	<b>Tag</b>	The tag property can be used to store any user-defined information

[Top](#)

## See Also

### Reference

[Event Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Allday Property

indicates all day event.

## Syntax

Visual Basic (Declaration)	
<pre>&lt;WidgetOptionAttribute()&gt; &lt;WidgetOptionNameAttribute("allday")&gt; Public Property Allday As System.Boolean</pre>	
C#	
<pre>[WidgetOption()] [WidgetOptionName("allday")] public System.bool Allday {get; set;}</pre>	

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## Calendar Property

Calendar id to which the event belongs.

## Syntax

Visual Basic (Declaration)	
<pre>&lt;WidgetOptionAttribute()&gt;</pre>	

```
<WidgetOptionNameAttribute("calendar")>  
Public Property Calendar As System.String
```

C#

```
[WidgetOption()]  
[WidgetOptionName("calendar")]  
public System.string Calendar {get; set;}
```

### Property Value

The calendar.

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## Color Property

Event color.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("color")>  
<WidgetOptionAttribute()>  
Public Property Color As System.String
```

C#

```
[WidgetOptionName("color")]  
[WidgetOption()]  
public System.string Color {get; set;}
```

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## Data Property

Gets or sets the raw event data.

## Syntax

Visual Basic (Declaration)	
Public Property Data As System.Collections.Hashtable	
C#	
public System.Collections.Hashtable Data {get; set;}	

## See Also

### Reference

[Event Class](#)[Event Members](#)

## Description Property

Event description.

## Syntax

Visual Basic (Declaration)	
<WidgetOptionNameAttribute("description")> <WidgetOptionAttribute()> Public Property Description As System.String	
C#	
[WidgetOptionName("description")] [WidgetOption()] public System.string Description {get; set;}	

## See Also

### Reference

[Event Class](#)[Event Members](#)

## End Property

End date/time.

## Syntax



## Visual Basic (Declaration)

```
<WidgetOptionAttribute()>  
<WidgetOptionNameAttribute("end")>  
Public Property End As System.Date
```

## C#

```
[WidgetOption()]  
[WidgetOptionName("end")]  
public System.DateTime End {get; set;}
```

## See Also

### Reference

[Event Class](#)[Event Members](#)

## Id Property

The unique event id (this field generated automatically).

## Syntax

## Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("id")>  
<WidgetOptionAttribute()>  
Public Property Id As System.String
```

## C#

```
[WidgetOptionName("id")]  
[WidgetOption()]  
public System.string Id {get; set;}
```

### Property Value

The calendar.

## See Also

### Reference

[Event Class](#)[Event Members](#)

## Location Property

Event location.

### Syntax

Visual Basic (Declaration)	
<pre>&lt;WidgetOptionAttribute()&gt; &lt;WidgetOptionNameAttribute("location")&gt; Public Property Location As System.String</pre>	
C#	
<pre>[WidgetOption()] [WidgetOptionName("location")] public System.string Location {get; set;}</pre>	

### See Also

#### Reference

[Event Class](#)

[Event Members](#)

## ParentRecurrenceId Property

id of the event object that defines the recurrence criteria for this event object. If an event is recurring (see `isRecurring`) it represents an occurrence in the series that is started and defined by a specific pattern event. Use the `getPattern` method in order to obtain the pattern of the current event. A pattern event can be recognized by its `recurrenceState` field set to the "master" value. The recurrence information defined by the pattern event can be accessed via the `recurrencePattern` field of the event. If this event is a not member of a recurrence, or is itself a root event, this field will be null.

### Syntax

Visual Basic (Declaration)	
<pre>&lt;WidgetOptionNameAttribute("parentRecurrenceId")&gt; &lt;WidgetOptionAttribute()&gt; Public Property ParentRecurrenceId As System.String</pre>	
C#	
<pre>[WidgetOptionName("parentRecurrenceId")] [WidgetOption()]</pre>	

```
public System.string ParentRecurrenceId {get; set;}
```

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## RecurrencePattern Property

Represents the recurrence attributes of an event. Only master events can have this field (recurrenceState is "master")

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("recurrencePattern")>  
<WidgetOptionAttribute()>  
Public Property RecurrencePattern As RecurrencePattern
```

C#

```
[WidgetOptionName("recurrencePattern")]  
[WidgetOption()]  
public RecurrencePattern RecurrencePattern {get; set;}
```

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## RecurrenceState Property

Indicates the recurrence state of the event. (possible values are "notRecurring"(or null), "master", "occurrence", "exception", "removed")

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("recurrenceState")>  
<WidgetOptionAttribute()>  
Public Property RecurrenceState As System.String
```

C#	
----	--

<pre>[WidgetOptionName("recurrenceState")] [WidgetOption()] public System.string RecurrenceState {get; set;}</pre>	
--	--

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## Start Property

Start date/time.

## Syntax

Visual Basic (Declaration)	
----------------------------	--

<pre>&lt;WidgetOptionNameAttribute("start")&gt; &lt;WidgetOptionAttribute()&gt; Public Property Start As System.Date</pre>	
--	--

C#	
----	--

<pre>[WidgetOptionName("start")] [WidgetOption()] public System.DateTime Start {get; set;}</pre>	
--	--

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## Subject Property

Event title.

## Syntax

Visual Basic (Declaration)	
----------------------------	--

<pre>&lt;WidgetOptionAttribute()&gt;</pre>	
--	--

```
<WidgetOptionNameAttribute("subject")>  
Public Property Subject As System.String
```

C#

```
[WidgetOption()]  
[WidgetOptionName("subject")]  
public System.string Subject {get; set;}
```

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## Tag Property

The tag property can be used to store any user-defined information

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("tag")>  
<WidgetOptionAttribute()>  
Public Property Tag As System.String
```

C#

```
[WidgetOptionName("tag")]  
[WidgetOption()]  
public System.string Tag {get; set;}
```

## See Also

### Reference







[Event Class](#)

[Event Members](#)

## Fields

For a list of all members of this type, see [Event members](#).

## Public Fields

	Name	Description
	<code>_allDayEvent</code>	
	<code>_color</code>	
	<code>_description</code>	
	<code>_end</code>	
	<code>_start</code>	
	<code>_subject</code>	

[Top](#)

## See Also

### Reference

[Event Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## \_allDayEvent Field

### Syntax

Visual Basic (Declaration)

```
Public _allDayEvent As System.Boolean
```

C#

```
public System.bool _allDayEvent
```

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## \_color Field

### Syntax

Visual Basic (Declaration)	
----------------------------	--

Public _color As System.String	
--------------------------------	--

C#	
----	--

public System.string _color	
-----------------------------	--

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## \_description Field

### Syntax

Visual Basic (Declaration)	
----------------------------	--

Public _description As System.String	
--------------------------------------	--

C#	
----	--

public System.string _description	
-----------------------------------	--

## See Also

### Reference

[Event Class](#)

[Event Members](#)

## \_end Field

### Syntax

Visual Basic (Declaration)	
----------------------------	--

Public _end As System.Date	
----------------------------	--

C#	
----	--

public System.DateTime _end	
-----------------------------	--

## See Also

### Reference

[Event Class](#)[Event Members](#)

## \_start Field

### Syntax

Visual Basic (Declaration)	
Public _start As System.Date	
C#	
public System.DateTime _start	

### See Also

#### Reference

[Event Class](#)[Event Members](#)

## \_subject Field

### Syntax

Visual Basic (Declaration)	
Public _subject As System.String	
C#	
public System.string _subject	

### See Also

#### Reference

[Event Class](#)[Event Members](#)

## LocalizationOption

Localization strings.

### Object Model

LocalizationOption



## Syntax

Visual Basic (Declaration)

```
Public Class LocalizationOption
    Inherits C1.Web.Wijmo.Controls.Settings
```

C#

```
public class LocalizationOption : C1.Web.Wijmo.Controls.Settings
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.LocalizationOption**

## See Also

### Reference

[LocalizationOption Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Localization strings.

## Object Model

LocalizationOption

## Syntax

Visual Basic (Declaration)

```
Public Class LocalizationOption
    Inherits C1.Web.Wijmo.Controls.Settings
```

C#

```
public class LocalizationOption : C1.Web.Wijmo.Controls.Settings
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.LocalizationOption**

## See Also

### Reference


[LocalizationOption Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members










The following tables list the members exposed by [LocalizationOption](#).




















### Public Constructors




















	Name	Description
	<a href="#">LocalizationOption Constructor</a>	Initializes a new instance of the <a href="#">LocalizationOption</a> class.



















[Top](#)

### Public Properties

	Name	Description
	<a href="#">activityCreatingCalendar</a>	activityCreatingCalendar string.
	<a href="#">activityCreatingEvent</a>	activityCreatingEvent string.
	<a href="#">activityDeletingCalendar</a>	activityDeletingCalendar string.
	<a href="#">activityDeletingEvent</a>	activityDeletingEvent string.
	<a href="#">activityLoading</a>	activityLoading string.
	<a href="#">activityUpdating</a>	activityUpdating string.
	<a href="#">activityUpdatingCalendar</a>	activityUpdatingCalendar string.
	<a href="#">activityUpdatingEvent</a>	activityUpdatingEvent string.
	<a href="#">agendaHeaderFullDateFormat</a>	agendaHeaderFullDateFormat string.


	<a href="#">agendaLoadingMoreEvents</a>	agendaLoadingMoreEvents string.
	<a href="#">agendaMoreEventsFormat</a>	agendaMoreEventsFormat string.
	<a href="#">agendaTimeFormat</a>	agendaTimeFormat string.
	<a href="#">buttonCancel</a>	buttonCancel string.
	<a href="#">buttonClearAll</a>	buttonClearAll string.
	<a href="#">buttonClose</a>	buttonClose string.
	<a href="#">buttonDayView</a>	buttonDayView string.
	<a href="#">buttonDelete</a>	buttonDelete string.
	<a href="#">buttonListView</a>	buttonListView string.
	<a href="#">buttonMonthView</a>	buttonMonthView string.
	<a href="#">buttonOK</a>	buttonOK string.
	<a href="#">buttonSave</a>	buttonSave string.
	<a href="#">buttonToday</a>	buttonToday string.
	<a href="#">buttonWeekView</a>	buttonWeekView string.
	<a href="#">calendarNextTooltip</a>	calendarNextTooltip string.
	<a href="#">calendarPrevTooltip</a>	calendarPrevTooltip string.
	<a href="#">calendarTitleFormat</a>	calendarTitleFormat string.
	<a href="#">calendarToolTipFormat</a>	calendarToolTipFormat string.
	<a href="#">customMonthViewHeaderFormat</a>	customMonthViewHeaderFormat string.

	<a href="#">customMonthViewSecondTitleFormat</a>	customMonthViewSecondTitleFormat string.
	<a href="#">customSingleMonthViewHeaderFormat</a>	customSingleMonthViewHeaderFormat string.
	<a href="#">customSingleWeekViewHeaderFormat</a>	customSingleWeekViewHeaderFormat string.
	<a href="#">customSingleYearViewHeaderFormat</a>	customSingleYearViewHeaderFormat string.
	<a href="#">customWeekViewHeaderFormat</a>	customWeekViewHeaderFormat string.
	<a href="#">customYearViewHeaderFormat</a>	customYearViewHeaderFormat string.
	<a href="#">customYearViewMonthFormat</a>	customYearViewMonthFormat string.
	<a href="#">customYearViewSecondTitleFormat</a>	customYearViewSecondTitleFormat string.
	<a href="#">dayDetailsLabelFulldateFormat</a>	dayDetailsLabelFulldateFormat string.
	<a href="#">dayDetailsLabelYearFormat</a>	dayDetailsLabelYearFormat string.
	<a href="#">labelAllDay</a>	labelAllDay string.
	<a href="#">labelCalendar</a>	labelCalendar string.
	<a href="#">labelCalendarName</a>	labelCalendarName string.
	<a href="#">labelColor</a>	labelColor string.
	<a href="#">labelDescription</a>	labelDescription string.
	<a href="#">labelEnds</a>	labelEnds string.
	<a href="#">labelLocation</a>	labelLocation string.
	<a href="#">labelName</a>	labelName string.
	<a href="#">labelRepeat</a>	labelRepeat string.

	<a href="#">labelStarts</a>	labelStarts string.
	<a href="#">labelToday</a>	labelToday string.
	<a href="#">logCouldntOpenLocalStorage</a>	logCouldntOpenLocalStorage string.
	<a href="#">messageEndOccursBeforeStart</a>	messageEndOccursBeforeStart string.
	<a href="#">monthCellMoreEventsFormat</a>	monthCellMoreEventsFormat string.
	<a href="#">monthViewHeaderFormat</a>	monthViewHeaderFormat string.
	<a href="#">navigatorBarNextTooltip</a>	navigatorBarNextTooltip string.
	<a href="#">navigatorBarPrevTooltip</a>	navigatorBarPrevTooltip string.
	<a href="#">promptOpenOccurrenceFormat</a>	promptOpenOccurrenceFormat string.
	<a href="#">repeatDaily</a>	repeatDaily string.
	<a href="#">repeatMonthly</a>	repeatMonthly string.
	<a href="#">repeatNone</a>	repeatNone string.
	<a href="#">repeatWeekly</a>	repeatWeekly string.
	<a href="#">repeatWorkDays</a>	repeatWorkDays string.
	<a href="#">repeatYearly</a>	repeatYearly string.
	<a href="#">textNewEvent</a>	textNewEvent string.
	<a href="#">titleEditCalendar</a>	titleEditCalendar string.
	<a href="#">weekViewHeaderFormat</a>	weekViewHeaderFormat string.
	<a href="#">weekViewHeaderFormat2Months</a>	weekViewHeaderFormat2Months string.

[Top](#)

## Public Methods

	Name	Description
	<a href="#">ToString</a>	Returns a <b>System.String</b> that represents this instance.

[Top](#)

## See Also

### Reference

[LocalizationOption Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## LocalizationOption Constructor

Initializes a new instance of the [LocalizationOption](#) class.

## Syntax

Visual Basic (Declaration)	
<pre>Public Function New( _     ByVal evCal As <a href="#">C1EventsCalendar</a> _ )</pre>	
C#	
<pre>public LocalizationOption(     <a href="#">C1EventsCalendar</a> evCal )</pre>	

### Parameters

*evCal*

## See Also

### Reference


[LocalizationOption Class](#)

[LocalizationOption Members](#)

## Methods

For a list of all members of this type, see [LocalizationOption members](#).

## Public Methods

	Name	Description
	<a href="#">ToString</a>	Returns a <b>System.String</b> that represents this instance.

[Top](#)

## See Also

### Reference

[LocalizationOption Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## ToString Method

Returns a **System.String** that represents this instance.

## Syntax

Visual Basic (Declaration)	
Public Overrides Function ToString() As System.String	
C#	
public override System.string ToString()	

### Return Value

A **System.String** that represents this instance.

## See Also

### Reference



















[LocalizationOption Class](#)[LocalizationOption Members](#)

## Properties


For a list of all members of this type, see [LocalizationOption members](#).




















## Public Properties









Name	Description
------	-------------

	<a href="#">activityCreatingCalendar</a>	activityCreatingCalendar string.
	<a href="#">activityCreatingEvent</a>	activityCreatingEvent string.
	<a href="#">activityDeletingCalendar</a>	activityDeletingCalendar string.
	<a href="#">activityDeletingEvent</a>	activityDeletingEvent string.
	<a href="#">activityLoading</a>	activityLoading string.
	<a href="#">activityUpdating</a>	activityUpdating string.
	<a href="#">activityUpdatingCalendar</a>	activityUpdatingCalendar string.
	<a href="#">activityUpdatingEvent</a>	activityUpdatingEvent string.
	<a href="#">agendaHeaderFullDateFormat</a>	agendaHeaderFullDateFormat string.
	<a href="#">agendaLoadingMoreEvents</a>	agendaLoadingMoreEvents string.
	<a href="#">agendaMoreEventsFormat</a>	agendaMoreEventsFormat string.
	<a href="#">agendaTimeFormat</a>	agendaTimeFormat string.
	<a href="#">buttonCancel</a>	buttonCancel string.
	<a href="#">buttonClearAll</a>	buttonClearAll string.
	<a href="#">buttonClose</a>	buttonClose string.
	<a href="#">buttonDayView</a>	buttonDayView string.
	<a href="#">buttonDelete</a>	buttonDelete string.
	<a href="#">buttonListView</a>	buttonListView string.
	<a href="#">buttonMonthView</a>	buttonMonthView string.



	<a href="#">buttonOK</a>	buttonOK string.
	<a href="#">buttonSave</a>	buttonSave string.
	<a href="#">buttonToday</a>	buttonToday string.
	<a href="#">buttonWeekView</a>	buttonWeekView string.
	<a href="#">calendarNextTooltip</a>	calendarNextTooltip string.
	<a href="#">calendarPrevTooltip</a>	calendarPrevTooltip string.
	<a href="#">calendarTitleFormat</a>	calendarTitleFormat string.
	<a href="#">calendarToolTipFormat</a>	calendarToolTipFormat string.
	<a href="#">customMonthViewHeaderFormat</a>	customMonthViewHeaderFormat string.
	<a href="#">customMonthViewSecondTitleFormat</a>	customMonthViewSecondTitleFormat string.
	<a href="#">customSingleMonthViewHeaderFormat</a>	customSingleMonthViewHeaderFormat string.
	<a href="#">customSingleWeekViewHeaderFormat</a>	customSingleWeekViewHeaderFormat string.
	<a href="#">customSingleYearViewHeaderFormat</a>	customSingleYearViewHeaderFormat string.
	<a href="#">customWeekViewHeaderFormat</a>	customWeekViewHeaderFormat string.
	<a href="#">customYearViewHeaderFormat</a>	customYearViewHeaderFormat string.
	<a href="#">customYearViewMonthFormat</a>	customYearViewMonthFormat string.
	<a href="#">customYearViewSecondTitleFormat</a>	customYearViewSecondTitleFormat string.
	<a href="#">dayDetailsLabelFulldateFormat</a>	dayDetailsLabelFulldateFormat string.
	<a href="#">dayDetailsLabelYearFormat</a>	dayDetailsLabelYearFormat string.

	labelAllDay	labelAllDay string.
	labelCalendar	labelCalendar string.
	labelCalendarName	labelCalendarName string.
	labelColor	labelColor string.
	labelDescription	labelDescription string.
	labelEnds	labelEnds string.
	labelLocation	labelLocation string.
	labelName	labelName string.
	labelRepeat	labelRepeat string.
	labelStarts	labelStarts string.
	labelToday	labelToday string.
	logCouldntOpenLocalStorage	logCouldntOpenLocalStorage string.
	messageEndOccursBeforeStart	messageEndOccursBeforeStart string.
	monthCellMoreEventsFormat	monthCellMoreEventsFormat string.
	monthViewHeaderFormat	monthViewHeaderFormat string.
	navigatorBarNextTooltip	navigatorBarNextTooltip string.
	navigatorBarPrevTooltip	navigatorBarPrevTooltip string.
	promptOpenOccurrenceFormat	promptOpenOccurrenceFormat string.
	repeatDaily	repeatDaily string.

	<a href="#">repeatMonthly</a>	repeatMonthly string.
	<a href="#">repeatNone</a>	repeatNone string.
	<a href="#">repeatWeekly</a>	repeatWeekly string.
	<a href="#">repeatWorkDays</a>	repeatWorkDays string.
	<a href="#">repeatYearly</a>	repeatYearly string.
	<a href="#">textNewEvent</a>	textNewEvent string.
	<a href="#">titleEditCalendar</a>	titleEditCalendar string.
	<a href="#">weekViewHeaderFormat</a>	weekViewHeaderFormat string.
	<a href="#">weekViewHeaderFormat2Months</a>	weekViewHeaderFormat2Months string.

[Top](#)

## See Also

### Reference

[LocalizationOption Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## activityCreatingCalendar Property

activityCreatingCalendar string.

## Syntax

Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.activityCrea
tingCalendar", Description="Prompt string for creating calendar.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
Public Property activityCreatingCalendar As System.String

```

C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.activityCreatingCalendar", Description="Prompt string for creating calendar.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
public System.string activityCreatingCalendar {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## activityCreatingEvent Property

activityCreatingEvent string.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.activityCreatingEvent", Description="Prompt string for creating event.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
Public Property activityCreatingEvent As System.String
```

C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.activityCreatingEvent", Description="Prompt string for creating event.")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
public System.string activityCreatingEvent {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## activityDeletingCalendar Property

activityDeletingCalendar string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.activityDele
tingCalendar", Description="Prompt string for deleting calendar.")>
Public Property activityDeletingCalendar As System.String
```

C#

```
[System.ComponentModel.NotifyParentProperty(true)]
[WidgetOption()]
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.activityDeletingCalen
dar", Description="Prompt string for deleting calendar.")]
public System.string activityDeletingCalendar {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## activityDeletingEvent Property

activityDeletingEvent string.

## Syntax

## Visual Basic (Declaration)

```

<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.activityDele
tingEvent", Description="Prompt string for deleting event.")>
Public Property activityDeletingEvent As System.String

```

## C#

```

[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.activityDeletingEvent
", Description="Prompt string for deleting event.")]
public System.string activityDeletingEvent {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## activityLoading Property

activityLoading string.

## Syntax

## Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.activityLoad
ing", Description="Prompt string for loading.")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property activityLoading As System.String

```

C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.activityLoading",
Description="Prompt string for loading.")]
[WidgetOption()]
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string activityLoading {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## activityUpdating Property

activityUpdating string.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.activityUpda
ting", Description="Prompt string for updating.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property activityUpdating As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.activityUpdating",
Description="Prompt string for updating.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string activityUpdating {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## activityUpdatingCalendar Property

activityUpdatingCalendar string.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>  
<C1CategoryAttribute("Appearance")>  
<System.ComponentModel.DefaultValueAttribute()>  
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.activityUpda  
tingCalendar", Description="Prompt string for updating calendar.")>  
<System.ComponentModel.NotifyParentPropertyAttribute(True)>  
<WidgetOptionAttribute()>  
Public Property activityUpdatingCalendar As System.String
```

C#

```
[Layout(LayoutType.Appearance)]  
[C1Category("Appearance")]  
[System.ComponentModel.DefaultValue()]  
[C1Description(Key="C1EventsCalendar.LocalizationOption.activityUpdatingCalen  
dar", Description="Prompt string for updating calendar.")]  
[System.ComponentModel.NotifyParentProperty(true)]  
[WidgetOption()]  
public System.string activityUpdatingCalendar {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## activityUpdatingEvent Property

activityUpdatingEvent string.

## Syntax



## Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.activityUpda
tingEvent", Description="Prompt string for updating event.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property activityUpdatingEvent As System.String

```

## C#

```

[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.activityUpdatingEvent
", Description="Prompt string for updating event.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string activityUpdatingEvent {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## agendaHeaderFullDateFormat Property

agendaHeaderFullDateFormat string.

## Syntax

## Visual Basic (Declaration)

```

<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.agendaHeader
FullDateFormat", Description="Date format for the agenda header.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property agendaHeaderFullDateFormat As System.String

```

C#

```
[WidgetOption()]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.agendaHeaderFullDateF
ormat", Description="Date format for the agenda header.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string agendaHeaderFullDateFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## agendaLoadingMoreEvents Property

agendaLoadingMoreEvents string.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute(>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.agendaLoadin
gMoreEvents", Description="Prompt string for agenda loading for more
events.")>
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute(>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property agendaLoadingMoreEvents As System.String
```

C#

```
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.agendaLoadingMoreEven
ts", Description="Prompt string for agenda loading for more events.")]
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
```

```
public System.string agendaLoadingMoreEvents {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## agendaMoreEventsFormat Property

agendaMoreEventsFormat string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.agendaMoreEv
entsFormat", Description="Text for more events of agenda.")>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
Public Property agendaMoreEventsFormat As System.String
```

C#

```
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1EventsCalendar.LocalizationOption.agendaMoreEventsForma
t", Description="Text for more events of agenda.")]
[C1Category("Appearance")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
public System.string agendaMoreEventsFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## agendaTimeFormat Property

agendaTimeFormat string.

## Syntax

### Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.agendaTimeFo
rmat", Description="Date format of agenda time.")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<WidgetOptionAttribute()>
Public Property agendaTimeFormat As System.String
```

### C#

```
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.agendaTimeFormat",
Description="Date format of agenda time.")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.NotifyParentProperty(true)]
[WidgetOption()]
public System.string agendaTimeFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonCancel Property

buttonCancel string.

## Syntax

### Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonCancel
", Description="Text for cancel button.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
```

```
<LayoutAttribute(LayoutType.Appearance)>
Public Property buttonCancel As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonCancel",
Description="Text for cancel button.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string buttonCancel {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonClearAll Property

buttonClearAll string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonClearA
ll", Description="Text for clear all button.")>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property buttonClearAll As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonClearAll",
Description="Text for clear all button.")]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
```

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string buttonClearAll {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonClose Property

buttonClose string.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonClose"
, Description="Text for close button.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property buttonClose As System.String
```

C#

```
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonClose",
Description="Text for close button.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string buttonClose {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonDayView Property

buttonDayView string.

### Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonDayView", Description="Text for day view button.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property buttonDayView As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonDayView",
Description="Text for day view button.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string buttonDayView {get; set;}
```

### See Also

#### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonDelete Property

buttonDelete string.

### Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
```

```
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonDelete", Description="Text for delete button.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property buttonDelete As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonDelete",
Description="Text for delete button.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string buttonDelete {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonListView Property

buttonListView string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonListVi
ew", Description="Text for list view button.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property buttonListView As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
```



```
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonListView",
Description="Text for list view button.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string buttonListView {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonMonthView Property

buttonMonthView string.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonMonthV
iew", Description="Text for month view button.")>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property buttonMonthView As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonMonthView",
Description="Text for month view button.")]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string buttonMonthView {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)[LocalizationOption Members](#)

## buttonOK Property

buttonOK string.

### Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute(>>
<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonOK",
Description="Text for ok button.")>
<WidgetOptionAttribute(>>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property buttonOK As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonOK",
Description="Text for ok button.")]
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string buttonOK {get; set;}
```

### See Also

#### Reference

[LocalizationOption Class](#)[LocalizationOption Members](#)

## buttonSave Property

buttonSave string.

### Syntax

Visual Basic (Declaration)

```

<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonSave",
Description="Text for save button.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property buttonSave As System.String

```

C#

```

[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonSave",
Description="Text for save button.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string buttonSave {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonToday Property

buttonToday string.

## Syntax

Visual Basic (Declaration)

```

<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonToday"
, Description="Text for today button.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property buttonToday As System.String

```

C#

```
[WidgetOption()]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonToday",
Description="Text for today button.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string buttonToday {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## buttonWeekView Property

buttonWeekView string.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.buttonWeekVi
ew", Description="Text for week view button.")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
Public Property buttonWeekView As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.buttonWeekView",
Description="Text for week view button.")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Category("Appearance")]
public System.string buttonWeekView {get; set;}
```

## See Also

## Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## calendarNextTooltip Property

calendarNextTooltip string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute(>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.calendarNext
Tooltip", Description="Text for calendar next button's tooltip.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<WidgetOptionAttribute(>
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property calendarNextTooltip As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.calendarNextTooltip",
Description="Text for calendar next button's tooltip.")]
[System.ComponentModel.NotifyParentProperty(true)]
[WidgetOption()]
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
public System.string calendarNextTooltip {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## calendarPrevTooltip Property

calendarPrevTooltip string.

## Syntax

## Visual Basic (Declaration)

```

<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.calendarPrev
Tooltip", Description="Text for calendar previous button's tooltip.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property calendarPrevTooltip As System.String

```

## C#

```

[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.calendarPrevTooltip",
Description="Text for calendar previous button's tooltip.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string calendarPrevTooltip {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## calendarTitleFormat Property

calendarTitleFormat string.

## Syntax

## Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.calendarTitl
eFormat", Description="Date format for the calendar title.")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property calendarTitleFormat As System.String

```

C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.calendarTitleFormat",
Description="Date format for the calendar title.")]
[WidgetOption()]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string calendarTitleFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## calendarToolTipFormat Property

calendarToolTipFormat string.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.calendarTool
TipFormat", Description="Date format for the calendar tootip.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
Public Property calendarToolTipFormat As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.calendarToolTipFormat
", Description="Date format for the calendar tootip.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
public System.string calendarToolTipFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customMonthViewHeaderFormat Property

customMonthViewHeaderFormat string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>  
<WidgetOptionAttribute()>  
<C1CategoryAttribute("Appearance")>  
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customMonthViewHeaderFormat", Description="Date format for the header of custom month view with multiple months.")>  
<System.ComponentModel.NotifyParentPropertyAttribute(True)>  
<LayoutAttribute(LayoutType.Appearance)>  
Public Property customMonthViewHeaderFormat As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]  
[WidgetOption()]  
[C1Category("Appearance")]  
[C1Description(Key="C1EventsCalendar.LocalizationOption.customMonthViewHeaderFormat", Description="Date format for the header of custom month view with multiple months.")]  
[System.ComponentModel.NotifyParentProperty(true)]  
[Layout(LayoutType.Appearance)]  
public System.string customMonthViewHeaderFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customMonthViewSecondTitleFormat Property

customMonthViewSecondTitleFormat string.



## Syntax

### Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customMonthViewSecondTitleFormat", Description="Date format for the secondary header of custom month view.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property customMonthViewSecondTitleFormat As System.String
```

### C#

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.customMonthViewSecondTitleFormat", Description="Date format for the secondary header of custom month view.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string customMonthViewSecondTitleFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customSingleMonthViewHeaderFormat Property

customSingleMonthViewHeaderFormat string.

## Syntax

### Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
```

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customSingle
MonthViewHeaderFormat", Description="Date format for the header of custom
month view with single month.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property customSingleMonthViewHeaderFormat As System.String
```

C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.customSingleMonthView
HeaderFormat", Description="Date format for the header of custom month view
with single month.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string customSingleMonthViewHeaderFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customSingleWeekViewHeaderFormat Property

customSingleWeekViewHeaderFormat string.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customSingle
WeekViewHeaderFormat", Description="Date format for the header of custom week
view with single week.")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property customSingleWeekViewHeaderFormat As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.customSingleWeekViewHeaderFormat", Description="Date format for the header of custom week view with single week.")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string customSingleWeekViewHeaderFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customSingleYearViewHeaderFormat Property

customSingleYearViewHeaderFormat string.

## Syntax

Visual Basic (Declaration)	
<pre>&lt;System.ComponentModel.NotifyParentPropertyAttribute(True)&gt; &lt;WidgetOptionAttribute()&gt; &lt;System.ComponentModel.DefaultValueAttribute()&gt; &lt;C1CategoryAttribute("Appearance")&gt; &lt;C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customSingleYearViewHeaderFormat", Description="Date format for the header of custom year view with single year.")&gt; &lt;LayoutAttribute(LayoutType.Appearance)&gt; Public Property customSingleYearViewHeaderFormat As System.String</pre>	
C#	
<pre>[System.ComponentModel.NotifyParentProperty(true)] [WidgetOption()] [System.ComponentModel.DefaultValue()] [C1Category("Appearance")] [C1Description(Key="C1EventsCalendar.LocalizationOption.customSingleYearViewHeaderFormat", Description="Date format for the header of custom year view with single year.")] [Layout(LayoutType.Appearance)]</pre>	

```
public System.string customSingleYearViewHeaderFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customWeekViewHeaderFormat Property

customWeekViewHeaderFormat string.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customWeekViewHeaderFormat", Description="Date format for the header of custom week view with multiple weeks.")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property customWeekViewHeaderFormat As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[C1Description(Key="C1EventsCalendar.LocalizationOption.customWeekViewHeaderFormat", Description="Date format for the header of custom week view with multiple weeks.")]
[Layout(LayoutType.Appearance)]
public System.string customWeekViewHeaderFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customYearViewHeaderFormat Property

customYearViewHeaderFormat string.

### Syntax

#### Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customYearViewHeaderFormat", Description="Date format for the header of custom year view with multiple year.")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property customYearViewHeaderFormat As System.String
```

#### C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.customYearViewHeaderFormat", Description="Date format for the header of custom year view with multiple year.")]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[WidgetOption()]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string customYearViewHeaderFormat {get; set;}
```

### See Also

#### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customYearViewMonthFormat Property

customYearViewMonthFormat string.

### Syntax

#### Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
```

```

<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customYearVi
ewMonthFormat", Description="Date format for the month cell of custom year
view.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property customYearViewMonthFormat As System.String

```

C#

```

[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.customYearViewMonthFo
rmat", Description="Date format for the month cell of custom year view.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string customYearViewMonthFormat {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## customYearViewSecondTitleFormat Property

customYearViewSecondTitleFormat string.

## Syntax

Visual Basic (Declaration)

```

<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.customYearVi
ewSecondTitleFormat", Description="Date format for the secondary header of
custom year view.")>
Public Property customYearViewSecondTitleFormat As System.String

```

C#

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[C1Description(Key="C1EventsCalendar.LocalizationOption.customYearViewSecondTitleFormat", Description="Date format for the secondary header of custom year view.")]
public System.string customYearViewSecondTitleFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## dayDetailsLabelFulldateFormat Property

dayDetailsLabelFulldateFormat string.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.dayDetailsLabelFulldateFormat", Description="Date format for the day details label of full date.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property dayDetailsLabelFulldateFormat As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.dayDetailsLabelFulldateFormat", Description="Date format for the day details label of full date.")]
```

```
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string dayDetailsLabelFullDateFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## dayDetailsLabelYearFormat Property

dayDetailsLabelYearFormat string.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.dayDetailsLa
belYearFormat", Description="Date format for the day details label of
year.")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property dayDetailsLabelYearFormat As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.dayDetailsLabelYearFo
rmat", Description="Date format for the day details label of year.")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string dayDetailsLabelYearFormat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)



## labelAllDay Property

labelAllDay string.

### Syntax

Visual Basic (Declaration)	
<pre> &lt;LayoutAttribute(LayoutType.Appearance)&gt; &lt;WidgetOptionAttribute()&gt; &lt;System.ComponentModel.DefaultValueAttribute()&gt; &lt;C1CategoryAttribute("Appearance")&gt; &lt;C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelAllDay" , Description="Text for all day label.")&gt; &lt;System.ComponentModel.NotifyParentPropertyAttribute(True)&gt; Public Property labelAllDay As System.String </pre>	
C#	
<pre> [Layout(LayoutType.Appearance)] [WidgetOption()] [System.ComponentModel.DefaultValue()] [C1Category("Appearance")] [C1Description(Key="C1EventsCalendar.LocalizationOption.labelAllDay", Description="Text for all day label.")] [System.ComponentModel.NotifyParentProperty(true)] public System.string labelAllDay {get; set;} </pre>	

### See Also

#### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## labelCalendar Property

labelCalendar string.

### Syntax

Visual Basic (Declaration)	
<pre> &lt;System.ComponentModel.DefaultValueAttribute()&gt; &lt;C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelCalenda r", Description="Text for calendar label.")&gt; </pre>	

```
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property labelCalendar As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelCalendar",
Description="Text for calendar label.")]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string labelCalendar {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## labelCalendarName Property

labelCalendarName string.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelCalendarName", Description="Text for calendar name.")>
Public Property labelCalendarName As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
```

```
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelCalendarName",
Description="Text for calendar name.")]
public System.string labelCalendarName {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## labelColor Property

labelColor string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelColor",
Description="Text for color button.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property labelColor As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelColor",
Description="Text for color button.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string labelColor {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)[LocalizationOption Members](#)

## labelDescription Property

labelDescription string.

### Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelDescription", Description="Text for description label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property labelDescription As System.String
```

C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelDescription",
Description="Text for description label.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string labelDescription {get; set;}
```

### See Also

#### Reference

[LocalizationOption Class](#)[LocalizationOption Members](#)

## labelEnds Property

labelEnds string.

### Syntax

Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelEnds",
Description="Text for end label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
Public Property labelEnds As System.String

```

C#

```

[C1Description(Key="C1EventsCalendar.LocalizationOption.labelEnds",
Description="Text for end label.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
public System.string labelEnds {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## labelLocation Property

labelLocation string.

## Syntax

Visual Basic (Declaration)

```

<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelLocatio
n", Description="Text for location label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property labelLocation As System.String

```

C#

```
[WidgetOption()]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelLocation",
Description="Text for location label.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string labelLocation {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## labelName Property

labelName string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelName",
Description="Text for name label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property labelName As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelName",
Description="Text for name label.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string labelName {get; set;}
```

## See Also

## Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## labelRepeat Property

labelRepeat string.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelRepeat",
, Description="Text for repeat label.")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property labelRepeat As System.String
```

C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelRepeat",
Description="Text for repeat label.")]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string labelRepeat {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## labelStarts Property

labelStarts string.

## Syntax

## Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelStarts"
, Description="Text for start label.")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property labelStarts As System.String

```

## C#

```

[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.NotifyParentProperty(true)]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelStarts",
Description="Text for start label.")]
[System.ComponentModel.DefaultValue()]
public System.string labelStarts {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## labelToday Property

labelToday string.

## Syntax

## Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.labelToday",
Description="Text for today label.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<WidgetOptionAttribute()>
Public Property labelToday As System.String

```



C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.labelToday",
Description="Text for today label.")]
[System.ComponentModel.NotifyParentProperty(true)]
[WidgetOption()]
public System.string labelToday {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## logCouldntOpenLocalStorage Property

logCouldntOpenLocalStorage string.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.logCouldntOp
enLocalStorage", Description="Text for couldn't open built-in local data
storage.")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
Public Property logCouldntOpenLocalStorage As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[System.ComponentModel.NotifyParentProperty(true)]
[C1Description(Key="C1EventsCalendar.LocalizationOption.logCouldntOpenLocalSt
orage", Description="Text for couldn't open built-in local data storage.")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
```

```
public System.string logCouldntOpenLocalStorage {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## messageEndOccursBeforeStart Property

messageEndOccursBeforeStart string.

## Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.messageEndOc
cursBeforeStart", Description="The end date you entered occurs before the
start date.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property messageEndOccursBeforeStart As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.messageEndOccursBefor
eStart", Description="The end date you entered occurs before the start
date.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string messageEndOccursBeforeStart {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## monthCellMoreEventsFormat Property

monthCellMoreEventsFormat string.

### Syntax

#### Visual Basic (Declaration)

```
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.monthCellMoreEventsFormat", Description="Text for more events of month cell.")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property monthCellMoreEventsFormat As System.String
```

#### C#

```
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.monthCellMoreEventsFormat", Description="Text for more events of month cell.")]
[WidgetOption()]
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
public System.string monthCellMoreEventsFormat {get; set;}
```

### See Also

#### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## monthViewHeaderFormat Property

monthViewHeaderFormat string.

### Syntax

#### Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.monthViewHeaderFormat", Description="Text for month view header format.")>
```

```

derFormat", Description="Date format for the header of month view.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute(<
Public Property monthViewHeaderFormat As System.String

```

C#

```

[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.monthViewHeaderFormat",
Description="Date format for the header of month view.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string monthViewHeaderFormat {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## navigatorBarNextTooltip Property

navigatorBarNextTooltip string.

## Syntax

Visual Basic (Declaration)

```

<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute(<
<System.ComponentModel.DefaultValueAttribute(<
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.navigatorBar
NextTooltip", Description="Text for calendar next button's tooltip of
navigator bar.")>
Public Property navigatorBarNextTooltip As System.String

```

C#

```

[System.ComponentModel.NotifyParentProperty(true)]

```

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.navigatorBarNextToolt
ip", Description="Text for calendar next button's tooltip of navigator
bar.")]
public System.string navigatorBarNextTooltip {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## navigatorBarPrevTooltip Property

navigatorBarPrevTooltip string.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.navigatorBar
PrevTooltip", Description="Text for calendar previous button's tooltip of
navigator bar.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property navigatorBarPrevTooltip As System.String
```

C#

```
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.navigatorBarPrevToolt
ip", Description="Text for calendar previous button's tooltip of navigator
bar.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.string navigatorBarPrevTooltip {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## promptOpenOccurrenceFormat Property

promptOpenOccurrenceFormat string.

## Syntax

Visual Basic (Declaration)	
<pre> &lt;System.ComponentModel.DefaultValueAttribute()&gt; &lt;C1CategoryAttribute("Appearance")&gt; &lt;C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.promptOpenOc currenceFormat", Description="Prompt text for opening occurrence.")&gt; &lt;System.ComponentModel.NotifyParentPropertyAttribute(True)&gt; &lt;LayoutAttribute(LayoutType.Appearance)&gt; &lt;WidgetOptionAttribute()&gt; Public Property promptOpenOccurrenceFormat As System.String </pre>	
C#	
<pre> [System.ComponentModel.DefaultValue()] [C1Category("Appearance")] [C1Description(Key="C1EventsCalendar.LocalizationOption.promptOpenOccurrenceF ormat", Description="Prompt text for opening occurrence.")] [System.ComponentModel.NotifyParentProperty(true)] [Layout(LayoutType.Appearance)] [WidgetOption()] public System.string promptOpenOccurrenceFormat {get; set;} </pre>	

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## repeatDaily Property

repeatDaily string.

## Syntax

## Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.repeatDaily",
, Description="Text for repeat daily.")>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DefaultValueAttribute()>
Public Property repeatDaily As System.String
```

## C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.repeatDaily",
Description="Text for repeat daily.")]
[C1Category("Appearance")]
[WidgetOption()]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DefaultValue()]
public System.string repeatDaily {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## repeatMonthly Property

repeatMonthly string.

## Syntax

## Visual Basic (Declaration)

```
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.repeatMonthl
y", Description="Text for repeat monthly.")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property repeatMonthly As System.String
```

C#

```
[System.ComponentModel.NotifyParentProperty(true)]
[C1Description(Key="C1EventsCalendar.LocalizationOption.repeatMonthly",
Description="Text for repeat monthly.")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
public System.string repeatMonthly {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## repeatNone Property

repeatNone string.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.repeatNone",
Description="Text for not repeat.")>
<LayoutAttribute(LayoutType.Appearance)>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property repeatNone As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.repeatNone",
Description="Text for not repeat.")]
[Layout(LayoutType.Appearance)]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string repeatNone {get; set;}
```



## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## repeatWeekly Property

repeatWeekly string.

## Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.repeatWeekly",
Description="Text for repeat weekly.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property repeatWeekly As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.repeatWeekly",
Description="Text for repeat weekly.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string repeatWeekly {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## repeatWorkDays Property

repeatWorkDays string.

## Syntax

## Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.repeatWorkDays", Description="Text for repeat work days.")>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
Public Property repeatWorkDays As System.String
```

## C#

```
[C1Description(Key="C1EventsCalendar.LocalizationOption.repeatWorkDays",
Description="Text for repeat work days.")]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
public System.string repeatWorkDays {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## repeatYearly Property

repeatYearly string.

## Syntax

## Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.repeatYearly", Description="Text for repeat yearly.")>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property repeatYearly As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Description(Key="C1EventsCalendar.LocalizationOption.repeatYearly",
Description="Text for repeat yearly.")]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.string repeatYearly {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## textNewEvent Property

textNewEvent string.

## Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.textNewEvent",
Description="Text for new event.")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property textNewEvent As System.String
```

C#

```
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.LocalizationOption.textNewEvent",
Description="Text for new event.")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string textNewEvent {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## titleEditCalendar Property

titleEditCalendar string.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.titleEditCalendar", Description="Text for Edit calendar.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property titleEditCalendar As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1EventsCalendar.LocalizationOption.titleEditCalendar",
Description="Text for Edit calendar.")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.String titleEditCalendar {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## weekViewHeaderFormat Property

weekViewHeaderFormat string.

## Syntax

## Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.weekViewHeaderFormat", Description="Date format for the header of week view.")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
Public Property weekViewHeaderFormat As System.String

```

## C#

```

[C1Description(Key="C1EventsCalendar.LocalizationOption.weekViewHeaderFormat", Description="Date format for the header of week view.")]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
public System.string weekViewHeaderFormat {get; set;}

```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## weekViewHeaderFormat2Months Property

weekViewHeaderFormat2Months string.

## Syntax

## Visual Basic (Declaration)

```

<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.LocalizationOption.weekViewHeaderFormat2Months", Description="Date format for the header of week view between 2 months.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>

```

```
Public Property weekViewHeaderFormat2Months As System.String
```

```
C#
```

```
[WidgetOption()]  
[C1Category("Appearance")]  
[System.ComponentModel.DefaultValue()]  
[C1Description(Key="C1EventsCalendar.LocalizationOption.weekViewHeaderFormat2  
Months", Description="Date format for the header of week view between 2  
months.")]  
[System.ComponentModel.NotifyParentProperty(true)]  
[Layout(LayoutType.Appearance)]  
public System.string weekViewHeaderFormat2Months {get; set;}
```

## See Also

### Reference

[LocalizationOption Class](#)

[LocalizationOption Members](#)

## MappingInfo

Information about mapping.

## Object Model

MappingInfo

## Syntax

```
Visual Basic (Declaration)
```

```
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,  
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>  
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>  
Public Class MappingInfo
```

```
C#
```

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,  
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]  
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]  
public class MappingInfo
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.MappingInfo**

## See Also

### Reference

[MappingInfo Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Information about mapping.

## Object Model

MappingInfo

## Syntax

Visual Basic (Declaration)

```
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,  
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>  
<System.Web.UI.PersistencemodeAttribute(PersistenceMode.InnerProperty)>  
Public Class MappingInfo
```

C#

```
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,  
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]  
[System.Web.UI.Persistencemode(PersistenceMode.InnerProperty)]  
public class MappingInfo
```

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.MappingInfo**

## See Also

### Reference


[MappingInfo Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members






The following tables list the members exposed by [MappingInfo](#).

### Public Constructors

	Name	Description
	<a href="#">MappingInfo Constructor</a>	Initializes a new instance of the <a href="#">MappingInfo</a> class.


[Top](#)

### Public Properties

	Name	Description
	<a href="#">DefaultValue</a>	The default value which will be used when mapping is not mapped to a data field or data field value is empty.
	<a href="#">IsMapped</a>	Indicates whether objects is mapped to the data field or property. (read only)
	<a href="#">MappingName</a>	Name of the data field or property to be bound to this object.
	<a href="#">PropertyName</a>	Name of the property which should be mapped. (read only)
	<a href="#">Required</a>	Indicates whether mapping for this property is required or optional. (read only)

[Top](#)

### Public Methods

	Name	Description
	<a href="#">ToString</a>	Returns a System.String that represents the current object.

[Top](#)

## See Also

### Reference



[MappingInfo Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## MappingInfo Constructor

Name of the prop.

if set to `true` [required].

The default value.

The state bag.

Initializes a new instance of the [MappingInfo](#) class.

### Syntax

Visual Basic (Declaration)

```
Public Function New( _  
    ByVal propName As System.String, _  
    ByVal required As System.Boolean, _  
    ByVal defaultValue As System.String, _  
    ByVal stateBag As System.Web.UI.StateBag _  
)
```

C#

```
public MappingInfo(  
    System.string propName,  
    System.bool required,  
    System.string defaultValue,  
    System.Web.UI.StateBag stateBag  
)
```

### Parameters

*propName*

Name of the prop.

*required*

if set to `true` [required].

*defaultValue*

The default value.

*stateBag*

The state bag.

## See Also

### Reference


[MappingInfo Class](#)

[MappingInfo Members](#)

## Methods

For a list of all members of this type, see [MappingInfo members](#).

## Public Methods

	Name	Description
	<a href="#">ToString</a>	Returns a System.String that represents the current object.

[Top](#)

## See Also

### Reference

[MappingInfo Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## ToString Method

Returns a System.String that represents the current object.

## Syntax

Visual Basic (Declaration)	
Public Overrides Function ToString() As System.String	
C#	
public override System.string ToString()	

## See Also

### Reference






[MappingInfo Class](#)

[MappingInfo Members](#)

## Properties

For a list of all members of this type, see [MappingInfo members](#).

### Public Properties

	Name	Description
	<a href="#">DefaultValue</a>	The default value which will be used when mapping is not mapped to a data field or data field value is empty.
	<a href="#">IsMapped</a>	Indicates whether objects is mapped to the data field or property. (read only)
	<a href="#">MappingName</a>	Name of the data field or property to be bound to this object.
	<a href="#">PropertyName</a>	Name of the property which should be mapped. (read only)
	<a href="#">Required</a>	Indicates whether mapping for this property is required or optional. (read only)

[Top](#)

### See Also

#### Reference

[MappingInfo Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

### DefaultValue Property

The default value which will be used when mapping is not mapped to a data field or data field value is empty.

### Syntax

Visual Basic (Declaration)	
<pre>&lt;System.ComponentModel.NotifyParentPropertyAttribute(True)&gt; &lt;System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)&gt; &lt;C1DescriptionAttribute(Key="C1EventsCalendar.MappingInfo.DefaultValue", Description="The default value which will be used when mapping is not mapped to a data field or data field value is empty.")&gt;</pre>	

Public Property DefaultValue As System.String
---

C#
----

<pre>[System.ComponentModel.NotifyParentProperty(true)] [System.ComponentModel.RefreshProperties(RefreshProperties.All)] [C1Description(Key="C1EventsCalendar.MappingInfo.DefaultValue", Description="The default value which will be used when mapping is not mapped to a data field or data field value is empty.")] public System.string DefaultValue {get; set;}</pre>
--

## See Also

### Reference

[MappingInfo Class](#)

[MappingInfo Members](#)

## IsMapped Property

Indicates whether objects is mapped to the data field or property. (read only)

## Syntax

Visual Basic (Declaration)
----------------------------

<pre>&lt;C1DescriptionAttribute(Key="C1EventsCalendar.MappingInfo.IsMapped", Description="Indicates whether objects is mapped to the data field or property. (read only)")&gt; Public ReadOnly Property IsMapped As System.Boolean</pre>
--

C#
----

<pre>[C1Description(Key="C1EventsCalendar.MappingInfo.IsMapped", Description="Indicates whether objects is mapped to the data field or property. (read only)")]</pre> <pre>public System.bool IsMapped {get;}</pre>
---

## See Also

### Reference

[MappingInfo Class](#)

[MappingInfo Members](#)

## MappingName Property

Name of the data field or property to be bound to this object.

## Syntax

### Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<C1DescriptionAttribute(Key="C1EventsCalendar.MappingInfo.MappingName",
Description="Name of the data field or property to be bound to this
object.")>
Public Property MappingName As System.String
```

### C#

```
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[C1Description(Key="C1EventsCalendar.MappingInfo.MappingName",
Description="Name of the data field or property to be bound to this
object.")]
public System.string MappingName {get; set;}
```

## See Also

### Reference

[MappingInfo Class](#)

[MappingInfo Members](#)

## PropertyName Property

Name of the property which should be mapped. (read only)

## Syntax

### Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.MappingInfo.PropertyName",
Description="Name of the property which should be mapped. (read only)")>
Public ReadOnly Property PropertyName As System.String
```

### C#

```
[C1Description(Key="C1EventsCalendar.MappingInfo.PropertyName",
Description="Name of the property which should be mapped. (read only)")]
```

```
public System.string PropertyName {get;}
```

## See Also

### Reference

[MappingInfo Class](#)

[MappingInfo Members](#)

## Required Property

Indicates whether mapping for this property is required or optional. (read only)

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.MappingInfo.Required",  
Description="Indicates whether mapping for this property is required or  
optional. (read only)")>  
Public ReadOnly Property Required As System.Boolean
```

C#

```
[C1Description(Key="C1EventsCalendar.MappingInfo.Required",  
Description="Indicates whether mapping for this property is required or  
optional. (read only)"]  
public System.bool Required {get;}
```

## See Also

### Reference

[MappingInfo Class](#)

[MappingInfo Members](#)

## RecurrencePattern

Represents the recurrence attributes of an event. Only master events can have this field (recurrenceState is "master").

## Object Model

RecurrencePattern

## Syntax

Visual Basic (Declaration)	
Public Class RecurrencePattern	
C#	
public class RecurrencePattern	

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.RecurrencePattern**

## See Also

### Reference

[RecurrencePattern Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

Represents the recurrence attributes of an event. Only master events can have this field (recurrenceState is "master").

## Object Model

RecurrencePattern

## Syntax

Visual Basic (Declaration)	
Public Class RecurrencePattern	
C#	
public class RecurrencePattern	

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.RecurrencePattern**

## See Also


### Reference

[RecurrencePattern Members](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members




The following tables list the members exposed by [RecurrencePattern](#).

### Public Constructors






	Name	Description
	<a href="#">RecurrencePattern Constructor</a>	Overloaded.








[Top](#)


### Public Properties

	Name	Description
	<a href="#">Data</a>	Gets or sets the raw pattern data.
	<a href="#">DayOfMonth</a>	The number of the day in its respective month on which each occurrence will occur. Applicable only when the recurrenceType field is set to "monthly" or "yearly". The default value is 1.
	<a href="#">dayOfWeekMask</a>	String, contains set of values representing the mask for week days on which the recurring event occurs. Monthly and yearly patterns are only valid for a single day. The default value is "none". When the "RecurrenceType" field is set to "daily", the "DayOfWeekMask" field can only be set to "everyDay"; setting the field to any other value will result in an exception. When the "recurrenceType" field is set to "workdays", the "dayOfWeekMask" field can only be set to "workDays"; setting the field to any other value will result in an exception. When the "recurrenceType" field is set to "weekly", the "dayOfWeekMask" field cannot be set to "none"; doing so will result in an exception being thrown. When the recurrenceType" field is set to "monthly" or "yearly" the "dayOfWeekMask" field is not applicable. Possible values are: none - no specific value; the actual value is obtained from the root event object. sunday - specifies Sunday. monday - Specifies Monday. tuesday - Specifies Tuesday. wednesday - Specifies Wednesday. thursday - Specifies Thursday.



		friday - Specifies Friday. saturday - Specifies Saturday. weekendDays - Specifies Saturday and Sunday (or what ever days according to the settings of the C1EventsCalendar). workDays - Specifies work days (all days except weekend). everyDay - Specifies every day of the week.
	EndTime	Indicates the end time for the given occurrence of the recurrence pattern.
	Exceptions	Holds the list of event object ids that define the exceptions to that series of events.
	Instance	Determines the week in a month in which the event will occur. This field is only valid for recurrences of the "monthlyNth" and "yearlyNth" types and allows the definition of a recurrence pattern that is only valid for the Nth occurrence, such as "the 2nd Sunday in March" pattern. The default value is "first". Possible values are: first - the recurring event will occur on the specified day or days of the first week in the month. second - The recurring event will occur on the specified day or days of the second week in the month. third - The recurring event will occur on the specified day or days of the third week in the month. fourth - The recurring event will occur on the specified day or days of the fourth week in the month. last - The recurring event will occur on the specified day or days of the last week in the month.
	Interval	Specifies the interval between occurrences of the recurrence. The interval field works in conjunction with the "recurrenceType" field to determine the cycle of the recurrence. The maximum allowable value is 99 for weekly patterns and 999 for daily patterns. The default value is 1. For example, if the recurrenceType is set to daily, and the "interval" is set to 3, the recurrence will occur every third day.
	MonthOfYear	Number, indicates which month of the year is valid for the specified recurrence pattern. Can be a number from 1 to 12. This field is only valid for recurrence patterns whose recurrenceType" field is set to

		"yearlyNth" or "yearly". The default value is 1.
	<b>NoEndDate</b>	Indicates if the recurrence pattern is endless. The default value is True. This field must be coordinated with other fields when setting up a recurrence pattern. If the patternEndDate field or the occurrences field is set, the pattern is considered to be finite and the "noEndDate" field is false. If neither patternEndDate nor occurrences is set, the pattern is considered infinite and "noEndDate" is true.
	<b>Occurrences</b>	The number of occurrences for the recurrence pattern. This field allows the definition of a recurrence pattern that is only valid for the specified number of subsequent occurrences. For example, you can set this property to 10 for a formal training course that will be held on the next ten Thursday evenings. The default value is 0. This field must be coordinated with other fields when setting up a recurrence pattern. If the "patternEndDate" field or the "occurrences" field is set, the pattern is considered to be finite and the "noEndDate" field is false. If neither "patternEndDate" nor "occurrences" is set, the pattern is considered infinite and "noEndDate" is true.
	<b>ParentRecurrenceId</b>	Id of the event object which represents the master event for this RecurrencePattern object.
	<b>PatternEndDate</b>	Indicates the end date of the recurrence pattern.
	<b>PatternStartDate</b>	Indicates the start date of the recurrence pattern.
	<b>RecurrenceType</b>	Determines the type of recurrence: daily - the recurring event reoccurs on a daily basis. workdays - the recurring event reoccurs every working day. monthly - the recurring event reoccurs on a monthly basis. monthlyNth - the recurring event reoccurs every N months. weekly - the recurring event reoccurs on a weekly basis. yearly - the recurring event reoccurs on an yearly basis. yearlyNth - the recurring event reoccurs every N years.
	<b>RemovedOccurrences</b>	Holds the list of event object's ids removed from that series of events.

	<a href="#">StartTime</a>	Indicates the start time for the given occurrence of the recurrence pattern.
---	---------------------------	--

[Top](#)

## See Also

### Reference

[RecurrencePattern Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## RecurrencePattern Constructor

### Overload List

Overload	Description
<a href="#">RecurrencePattern Constructor()</a>	
<a href="#">RecurrencePattern Constructor(String,Hashtable)</a>	Initializes a new instance of the <a href="#">RecurrencePattern</a> class.

## See Also

### Reference

[RecurrencePattern Class](#)[RecurrencePattern Members](#)

## RecurrencePattern Constructor()

### Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public RecurrencePattern()	

## See Also

### Reference

[RecurrencePattern Class](#)[RecurrencePattern Members](#)[Overload List](#)

## RecurrencePattern Constructor(String,Hashtable)

Initializes a new instance of the [RecurrencePattern](#) class.

### Syntax

Visual Basic (Declaration)

```
Public Function New( _  
    ByVal parentRecurrenceId As System.String, _  
    ByVal rawData As System.Collections.Hashtable _  
)
```

C#

```
public RecurrencePattern(  
    System.string parentRecurrenceId,  
    System.Collections.Hashtable rawData  
)
```

### Parameters

*parentRecurrenceId*

The parent id.

*rawData*

The raw pattern data.

### See Also







### Reference





[RecurrencePattern Class](#)[RecurrencePattern Members](#)[Overload List](#)







## Properties

For a list of all members of this type, see [RecurrencePattern members](#).

## Public Properties

	Name	Description
	Data	Gets or sets the raw pattern data.
	DayOfMonth	The number of the day in its respective month on which each occurrence will occur. Applicable only when the recurrenceType field is set to "monthly" or "yearly". The default value is 1.
	dayOfWeekMask	String, contains set of values representing the mask for week days on which the recurring event occurs. Monthly and yearly patterns are only valid for a single day. The default value is "none". When the "RecurrenceType" field is set to "daily", the "DayOfWeekMask" field can only be set to "everyDay"; setting the field to any other value will result in an exception. When the "recurrenceType" field is set to "workdays", the "dayOfWeekMask" field can only be set to "workDays"; setting the field to any other value will result in an exception. When the "recurrenceType" field is set to "weekly", the "dayOfWeekMask" field cannot be set to "none"; doing so will result in an exception being thrown. When the recurrenceType" field is set to "monthly" or "yearly" the "dayOfWeekMask" field is not applicable. Possible values are: none - no specific value; the actual value is obtained from the root event object. sunday - specifies Sunday. monday - Specifies Monday. tuesday - Specifies Tuesday. wednesday - Specifies Wednesday. thursday - Specifies Thursday. friday - Specifies Friday. saturday - Specifies Saturday. weekendDays - Specifies Saturday and Sunday (or what ever days according to the settings of the C1EventsCalendar). workDays - Specifies work days (all days except weekend). everyDay - Specifies every day of the week.
	EndTime	Indicates the end time for the given occurrence of the recurrence pattern.
	Exceptions	Holds the list of event object ids that define the exceptions to that series of events.
	Instance	Determines the week in a month in which the event will occur. This

		field is only valid for recurrences of the "monthlyNth" and "yearlyNth" types and allows the definition of a recurrence pattern that is only valid for the Nth occurrence, such as "the 2nd Sunday in March" pattern. The default value is "first". Possible values are: first - the recurring event will occur on the specified day or days of the first week in the month. second - The recurring event will occur on the specified day or days of the second week in the month. third = - The recurring event will occur on the specified day or days of the third week in the month. fourth - The recurring event will occur on the specified day or days of the fourth week in the month. last - The recurring event will occur on the specified day or days of the last week in the month.
	<b>Interval</b>	Specifies the interval between occurrences of the recurrence. The interval field works in conjunction with the "recurrenceType" field to determine the cycle of the recurrence. The maximum allowable value is 99 for weekly patterns and 999 for daily patterns. The default value is 1. For example, if the recurrenceType is set to daily, and the "interval" is set to 3, the recurrence will occur every third day.
	<b>MonthOfYear</b>	Number, indicates which month of the year is valid for the specified recurrence pattern. Can be a number from 1 to 12. This field is only valid for recurrence patterns whose recurrenceType" field is set to "yearlyNth" or "yearly". The default value is 1.
	<b>NoEndDate</b>	Indicates if the recurrence pattern is endless. The default value is True. This field must be coordinated with other fields when setting up a recurrence pattern. If the patternEndDate field or the occurrences field is set, the pattern is considered to be finite and the "noEndDate" field is false. If neither patternEndDate nor occurrences is set, the pattern is considered infinite and "noEndDate" is true.
	<b>Occurrences</b>	The number of occurrences for the recurrence pattern. This field allows the definition of a recurrence pattern that is only valid for the specified number of subsequent occurrences. For example, you can set this property to 10 for a formal training course that will be held

		on the next ten Thursday evenings. The default value is 0. This field must be coordinated with other fields when setting up a recurrence pattern. If the "patternEndDate" field or the "occurrences" field is set, the pattern is considered to be finite and the "noEndDate" field is false. If neither "patternEndDate" nor "occurrences" is set, the pattern is considered infinite and "noEndDate" is true.
	<a href="#">ParentRecurrenceId</a>	Id of the event object which represents the master event for this RecurrencePattern object.
	<a href="#">PatternEndDate</a>	Indicates the end date of the recurrence pattern.
	<a href="#">PatternStartDate</a>	Indicates the start date of the recurrence pattern.
	<a href="#">RecurrenceType</a>	Determines the type of recurrence: daily - the recurring event reoccurs on a daily basis. workdays - the recurring event reoccurs every working day. monthly - the recurring event reoccurs on a monthly basis. monthlyNth - the recurring event reoccurs every N months. weekly - the recurring event reoccurs on a weekly basis. yearly - the recurring event reoccurs on an yearly basis. yearlyNth - the recurring event reoccurs every N years.
	<a href="#">RemovedOccurrences</a>	Holds the list of event object's ids removed from that series of events.
	<a href="#">StartTime</a>	Indicates the start time for the given occurrence of the recurrence pattern.

[Top](#)

## See Also

### Reference

[RecurrencePattern Class](#)[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Data Property

Gets or sets the raw pattern data.

## Syntax

Visual Basic (Declaration)	
Public Property Data As System.Collections.Hashtable	
C#	
public System.Collections.Hashtable Data {get; set;}	

### Property Value

The data.

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## DayOfMonth Property

The number of the day in its respective month on which each occurrence will occur. Applicable only when the recurrenceType field is set to "monthly" or "yearly". The default value is 1.

## Syntax

Visual Basic (Declaration)	
<WidgetOptionAttribute(> <WidgetOptionNameAttribute("dayOfMonth")> Public Property DayOfMonth As System.Integer	
C#	
[WidgetOption()] [WidgetOptionName("dayOfMonth")] public System.int DayOfMonth {get; set;}	

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)



## dayOfWeekMask Property

String, contains set of values representing the mask for week days on which the recurring event occurs. Monthly and yearly patterns are only valid for a single day. The default value is "none". When the "RecurrenceType" field is set to "daily", the "DayOfWeekMask" field can only be set to "everyDay"; setting the field to any other value will result in an exception. When the "recurrenceType" field is set to "workdays", the "dayOfWeekMask" field can only be set to "workDays"; setting the field to any other value will result in an exception. When the "recurrenceType" field is set to "weekly", the "dayOfWeekMask" field cannot be set to "none"; doing so will result in an exception being thrown. When the recurrenceType" field is set to "monthly" or "yearly" the "dayOfWeekMask" field is not applicable. Possible values are: none - no specific value; the actual value is obtained from the root event object. sunday - specifies Sunday. monday - Specifies Monday. tuesday - Specifies Tuesday. wednesday - Specifies Wednesday. thursday - Specifies Thursday. friday - Specifies Friday. saturday - Specifies Saturday. weekendDays - Specifies Saturday and Sunday (or what ever days according to the settings of the C1EventsCalendar). workDays - Specifies work days (all days except weekend). everyDay - Specifies every day of the week.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("dayOfWeekMask")>  
<WidgetOptionAttribute()>  
Public Property dayOfWeekMask As System.String
```

C#

```
[WidgetOptionName("dayOfWeekMask")]  
[WidgetOption()]  
public System.string dayOfWeekMask {get; set;}
```

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## EndTime Property

Indicates the end time for the given occurrence of the recurrence pattern.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute(>
<WidgetOptionNameAttribute("endTime")>
Public Property EndTime As System.Date
```

C#

```
[WidgetOption()]
[WidgetOptionName("endTime")]
public System.DateTime EndTime {get; set;}
```

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## Exceptions Property

Holds the list of event object ids that define the exceptions to that series of events.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("exceptions")>
<WidgetOptionAttribute(>
Public ReadOnly Property Exceptions As System.Collections.Generic.List(Of
String)
```

C#

```
[WidgetOptionName("exceptions")]
[WidgetOption()]
public System.Collections.Generic.List<string> Exceptions {get;}
```

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## Instance Property

Determines the week in a month in which the event will occur. This field is only valid for recurrences of the "monthlyNth" and "yearlyNth" types and allows the definition of a recurrence pattern that is only valid for the Nth occurrence, such as "the 2nd Sunday in March" pattern.

The default value is "first". Possible values are: first - the recurring event will occur on the specified day or days of the first week in the month. second - The recurring event will occur on the specified day or days of the second week in the month. third = - The recurring event will occur on the specified day or days of the third week in the month. fourth - The recurring event will occur on the specified day or days of the fourth week in the month. last - The recurring event will occur on the specified day or days of the last week in the month.

## Syntax

Visual Basic (Declaration)	
<pre>&lt;WidgetOptionNameAttribute("instance")&gt; &lt;WidgetOptionAttribute()&gt; Public Property Instance As System.String</pre>	
C#	
<pre>[WidgetOptionName("instance")] [WidgetOption()] public System.string Instance {get; set;}</pre>	

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## Interval Property

Specifies the interval between occurrences of the recurrence. The interval field works in conjunction with the "recurrenceType" field to determine the cycle of the recurrence. The maximum allowable value is 99 for weekly patterns and 999 for daily patterns. The default value is 1. For example, if the recurrenceType is set to daily, and the "interval" is set to 3, the recurrence will occur every third day.

## Syntax

Visual Basic (Declaration)	
<pre>&lt;WidgetOptionAttribute()&gt; &lt;WidgetOptionNameAttribute("interval")&gt; Public Property Interval As System.Integer</pre>	
C#	

```
[WidgetOption()]
[WidgetOptionName("interval")]
public System.int Interval {get; set;}
```

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## MonthOfYear Property

Number, indicates which month of the year is valid for the specified recurrence pattern. Can be a number from 1 to 12. This field is only valid for recurrence patterns whose recurrenceType" field is set to "yearlyNth" or "yearly". The default value is 1.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("monthOfYear")>
<WidgetOptionAttribute()>
Public Property MonthOfYear As System.Integer
```

C#

```
[WidgetOptionName("monthOfYear")]
[WidgetOption()]
public System.int MonthOfYear {get; set;}
```

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## NoEndDate Property

Indicates if the recurrence pattern is endless. The default value is True. This field must be coordinated with other fields when setting up a recurrence pattern. If the patternEndDate field or the occurrences field is set, the pattern is considered to be finite and the "noEndDate" field is false. If neither patternEndDate nor occurrences is set, the pattern is considered infinite and "noEndDate" is true.

## Syntax

## Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<WidgetOptionNameAttribute("noEndDate")>
Public Property NoEndDate As System.Boolean
```

## C#

```
[WidgetOption()]
[WidgetOptionName("noEndDate")]
public System.bool NoEndDate {get; set;}
```

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## Occurrences Property

The number of occurrences for the recurrence pattern. This field allows the definition of a recurrence pattern that is only valid for the specified number of subsequent occurrences. For example, you can set this property to 10 for a formal training course that will be held on the next ten Thursday evenings. The default value is 0. This field must be coordinated with other fields when setting up a recurrence pattern. If the "patternEndDate" field or the "occurrences" field is set, the pattern is considered to be finite and the "noEndDate" field is false. If neither "patternEndDate" nor "occurrences" is set, the pattern is considered infinite and "noEndDate" is true.

## Syntax

## Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<WidgetOptionNameAttribute("occurrences")>
Public Property Occurrences As System.Integer
```

## C#

```
[WidgetOption()]
[WidgetOptionName("occurrences")]
public System.int Occurrences {get; set;}
```

## See Also

### Reference

[RecurrencePattern Class](#)[RecurrencePattern Members](#)

## ParentRecurrenceId Property

Id of the event object which represents the master event for this RecurrencePattern object.

### Syntax

Visual Basic (Declaration)	
<code>&lt;WidgetOptionAttribute()&gt; &lt;WidgetOptionNameAttribute("parentRecurrenceId")&gt; Public ReadOnly Property ParentRecurrenceId As System.String</code>	
C#	
<code>[WidgetOption()] [WidgetOptionName("parentRecurrenceId")] public System.string ParentRecurrenceId {get;}</code>	

### See Also

#### Reference

[RecurrencePattern Class](#)[RecurrencePattern Members](#)

## PatternEndDate Property

Indicates the end date of the recurrence pattern.

### Syntax

Visual Basic (Declaration)	
<code>&lt;WidgetOptionNameAttribute("patternEndDate")&gt; &lt;WidgetOptionAttribute()&gt; Public Property PatternEndDate As System.Date</code>	
C#	
<code>[WidgetOptionName("patternEndDate")] [WidgetOption()] public System.DateTime PatternEndDate {get; set;}</code>	

### See Also

## Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## PatternStartDate Property

Indicates the start date of the recurrence pattern.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute(>  
<WidgetOptionNameAttribute("patternStartDate")>  
Public Property PatternStartDate As System.Date
```

C#

```
[WidgetOption()]  
[WidgetOptionName("patternStartDate")]  
public System.DateTime PatternStartDate {get; set;}
```

## See Also

## Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## RecurrenceType Property

Determines the type of recurrence: daily - the recurring event reoccurs on a daily basis.  
workdays - the recurring event reoccurs every working day. monthly - the recurring event reoccurs on a monthly basis. monthlyNth - the recurring event reoccurs every N months. weekly - the recurring event reoccurs on a weekly basis. yearly - the recurring event reoccurs on an yearly basis. yearlyNth - the recurring event reoccurs every N years.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute(>  
<WidgetOptionNameAttribute("recurrenceType")>  
Public Property RecurrenceType As System.String
```

C#

```
[WidgetOption()]
[WidgetOptionName("recurrenceType")]
public System.string RecurrenceType {get; set;}
```

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## RemovedOccurrences Property

Holds the list of event object's ids removed from that series of events.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("removedOccurrences")>
<WidgetOptionAttribute()>
Public ReadOnly Property RemovedOccurrences As
System.Collections.Generic.List(Of String)
```

C#

```
[WidgetOptionName("removedOccurrences")]
[WidgetOption()]
public System.Collections.Generic.List<string> RemovedOccurrences {get;}
```

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## StartTime Property

Indicates the start time for the given occurrence of the recurrence pattern.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<WidgetOptionNameAttribute("startTime")>
```



Public Property StartTime As System.Date
--

C#
----

[WidgetOption()] [WidgetOptionName("startTime")] public System.DateTime StartTime {get; set;}
---

## See Also

### Reference

[RecurrencePattern Class](#)

[RecurrencePattern Members](#)

## View

The view can be added to C1EventsCalendar's Views.

## Object Model

View

## Syntax

Visual Basic (Declaration)
----------------------------

Public Class View Inherits C1.Web.Wijmo.Controls.Settings
--

C#
----

public class View : C1.Web.Wijmo.Controls.Settings
--

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.View**

## See Also

### Reference

[View Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Overview

The view can be added to C1EventsCalendar's Views.

## Object Model

View

## Syntax

Visual Basic (Declaration)	
<pre>Public Class View     Inherits C1.Web.Wijmo.Controls.Settings</pre>	
C#	
<pre>public class View : C1.Web.Wijmo.Controls.Settings</pre>	

## Inheritance Hierarchy

System.Object

**C1.Web.Wijmo.Controls.C1EventsCalendar.View**

## See Also

### Reference

[View Members](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Members






The following tables list the members exposed by [View](#).

### Public Constructors

	Name	Description
	<a href="#">View Constructor</a>	Overloaded.

[Top](#)

### Public Properties

	Name	Description
	<a href="#">Count</a>	The count of time span, depends on the unit. It is effective only when the view is custom view.
	<a href="#">IsActive</a>	Determine if this view is active.
	<a href="#">Name</a>	The unique name of custom view which is displayed on the toolbar. It is effective only when the view is custom view.
	<a href="#">Type</a>	The type of view. If it is a custom view, it should be set to <code>C1EventsCalendarViewType.Custom</code> .
	<a href="#">Unit</a>	The time unit of custom view. Possible values are day, week, month, year. It is effective only when the view is custom view.

[Top](#)

## See Also

### Reference

[View Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## View Constructor

### Overload List

Overload	Description
<a href="#">View Constructor(C1EventsCalendarViewType)</a>	Initializes a new instance of the <a href="#">View</a> class.
<a href="#">View Constructor(C1EventsCalendarViewType,Boolean)</a>	Initializes a new instance of the <a href="#">View</a> class.
<a href="#">View Constructor()</a>	Initializes a new instance of the <a href="#">View</a> class.

## See Also

## Reference

[View Class](#)

[View Members](#)

## View Constructor(C1EventsCalendarViewType)

Initializes a new instance of the [View](#) class.

## Syntax

Visual Basic (Declaration)	
<pre>Public Function New( _     ByVal type As C1EventsCalendarViewType _ )</pre>	
C#	
<pre>public View(     C1EventsCalendarViewType type )</pre>	

## Parameters

*type*

## See Also

## Reference

[View Class](#)

[View Members](#)

[Overload List](#)

## View Constructor(C1EventsCalendarViewType,Boolean)

Initializes a new instance of the [View](#) class.

## Syntax

Visual Basic (Declaration)	
<pre>Public Function New( _     ByVal type As C1EventsCalendarViewType, _     ByVal isActive As System.Boolean _ )</pre>	

**C#**

```
public View(  
    C1EventsCalendarViewType type,  
    System.bool isActive  
)
```

### Parameters

*type*

*isActive*

## See Also

### Reference

[View Class](#)

[View Members](#)

[Overload List](#)

## View Constructor()

Initializes a new instance of the [View](#) class.

## Syntax

**Visual Basic (Declaration)**

```
Public Function New()
```

**C#**

```
public View()
```

## See Also

### Reference

[View Class](#)






[View Members](#)

[Overload List](#)

## Properties

For a list of all members of this type, see [View members](#).

## Public Properties

	Name	Description
	<a href="#">Count</a>	The count of time span, depends on the unit. It is effective only when the view is custom view.
	<a href="#">IsActive</a>	Determine if this view is active.
	<a href="#">Name</a>	The unique name of custom view which is displayed on the toolbar. It is effective only when the view is custom view.
	<a href="#">Type</a>	The type of view. If it is a custom view, it should be set to C1EventsCalendarViewType.Custom.
	<a href="#">Unit</a>	The time unit of custom view. Possible values are day, week, month, year. It is effective only when the view is custom view.

[Top](#)

## See Also

### Reference

[View Class](#)

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## Count Property

The count of time span, depends on the unit. It is effective only when the view is custom view.

## Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.View.Count", Description="The
count of time span, depends on the unit.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Count As System.Integer
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.View.Count", Description="The count of
time span, depends on the unit.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public System.int Count {get; set;}
```

## See Also

### Reference

[View Class](#)

[View Members](#)

## IsActive Property

Determine if this view is active.

## Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1EventsCalendar.View.IsActive",
Description="Determine if the view is active.")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public Property IsActive As System.Boolean
```

C#

```
[C1Description(Key="C1EventsCalendar.View.IsActive", Description="Determine
if the view is active.")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
public System.bool IsActive {get; set;}
```

## See Also

### Reference

[View Class](#)[View Members](#)

## Name Property

The unique name of custom view which is displayed on the toolbar. It is effective only when the view is custom view.

### Syntax

Visual Basic (Declaration)
<pre>&lt;WidgetOptionAttribute()&gt; &lt;System.ComponentModel.DefaultValueAttribute()&gt; &lt;C1CategoryAttribute("Appearance")&gt; &lt;C1DescriptionAttribute(Key="C1EventsCalendar.View.Name", Description="The name of custom view which is displayed on the toolbar.")&gt; &lt;System.ComponentModel.NotifyParentPropertyAttribute(True)&gt; &lt;LayoutAttribute(LayoutType.Appearance)&gt; Public Property Name As System.String</pre>
C#
<pre>[WidgetOption()] [System.ComponentModel.DefaultValue()] [C1Category("Appearance")] [C1Description(Key="C1EventsCalendar.View.Name", Description="The name of custom view which is displayed on the toolbar.")] [System.ComponentModel.NotifyParentProperty(true)] [Layout(LayoutType.Appearance)] public System.string Name {get; set;}</pre>

### See Also

#### Reference

[View Class](#)[View Members](#)

## Type Property

The type of view. If it is a custom view, it should be set to C1EventsCalendarViewType.Custom.

### Syntax

Visual Basic (Declaration)
----------------------------



```

<C1CategoryAttribute("Appearance")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1EventsCalendar.View.Type", Description="The
type of view")>
Public Property Type As C1EventsCalendarViewType

```

C#

```

[C1Category("Appearance")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1EventsCalendar.View.Type", Description="The type of
view")]
public C1EventsCalendarViewType Type {get; set;}

```

## See Also

### Reference

[View Class](#)

[View Members](#)

## Unit Property

The time unit of custom view. Possible values are day, week, month, year. It is effective only when the view is custom view.

## Syntax

Visual Basic (Declaration)

```

<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1EventsCalendar.View.Unit", Description="The
time unit of custom view.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Unit As C1EventsCalendarCustomViewUnit

```

C#

```
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Category("Appearance")]
[C1Description(Key="C1EventsCalendar.View.Unit", Description="The time unit
of custom view.")]
[System.ComponentModel.NotifyParentProperty(true)]
[Layout(LayoutType.Appearance)]
public C1EventsCalendarCustomViewUnit Unit {get; set;}
```

## See Also

### Reference

[View Class](#)[View Members](#)

## Enumerations

### C1EventsCalendarCustomViewUnit

Specifies the unit of the calendar custom view.

### Syntax

Visual Basic (Declaration)

```
Public Enum C1EventsCalendarCustomViewUnit
    Inherits System.Enum
```

C#

```
public enum C1EventsCalendarCustomViewUnit : System.Enum
```

### Members

Member	Description
<b>Day</b>	Day
<b>Month</b>	Month

<b>Week</b>	Week
<b>Year</b>	Year

## Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendarCustomViewUnit**

## See Also

### Reference

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## C1EventsCalendarViewType

Specifies the type of the calendar view.

## Syntax

Visual Basic (Declaration)	
Public Enum C1EventsCalendarViewType Inherits System.Enum	
C#	
public enum C1EventsCalendarViewType : System.Enum	

## Members

Member	Description
<b>Custom</b>	Custom
<b>Day</b>	Day
<b>List</b>	List
<b>Month</b>	Month

<b>Week</b>	Week
-------------	------

## Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

**C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendarViewType**

## See Also

### Reference

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## FileFormatEnum

Specifies the format of the exported or imported file.

## Syntax

Visual Basic (Declaration)	
<pre>Public Enum FileFormatEnum     Inherits System.Enum</pre>	
C#	
<pre>public enum FileFormatEnum : System.Enum</pre>	

## Members

Member	Description
<b>Binary</b>	Binary format.
<b>iCal</b>	iCal format according to RFC 2445. Format specification could be found here: <a href="http://tools.ietf.org/rfc/rfc2445.txt">http://tools.ietf.org/rfc/rfc2445.txt</a>
<b>XML</b>	XML format (ComponentOne's own format).

## Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

**C1.Web.Wijmo.Controls.C1EventsCalendar.FileFormatEnum**

## See Also

### Reference

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

# RecurrencePattern.RecurrenceTypeEnum

Determines the types of recurrent appointments.

## Syntax

Visual Basic (Declaration)	
<pre>Public Enum RecurrencePattern.RecurrenceTypeEnum     Inherits System.Enum</pre>	
C#	
<pre>public enum RecurrencePattern.RecurrenceTypeEnum : System.Enum</pre>	

## Members

Member	Description
<b>Daily</b>	The recurring appointment reoccurs on a daily basis.
<b>Monthly</b>	The recurring appointment reoccurs on a monthly basis.
<b>MonthlyNth</b>	The recurring appointment reoccurs every N months.
<b>Weekly</b>	The recurring appointment reoccurs on a weekly basis.
<b>Workdays</b>	The recurring appointment reoccurs every working day.
<b>Yearly</b>	The recurring appointment reoccurs on an yearly basis.
<b>YearlyNth</b>	The recurring appointment reoccurs every N years.

## Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

**C1.Web.Wijmo.Controls.C1EventsCalendar.RecurrencePattern.RecurrenceTypeEnum**

## See Also

### Reference

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

# RecurrencePattern.WeekDaysEnum

Determines the set of days and groups of days for recurrence patterns.

## Syntax

Visual Basic (Declaration)	
<pre>&lt;System.FlagsAttribute(&gt;&gt; Public Enum RecurrencePattern.WeekDaysEnum     Inherits System.Enum</pre>	
C#	
<pre>[System.Flags()] public enum RecurrencePattern.WeekDaysEnum : System.Enum</pre>	

## Members

Member	Description
<b>EveryDay</b>	Specifies every day of the week.
<b>Friday</b>	Specifies Friday.
<b>Monday</b>	Specifies Monday.
<b>None</b>	No specific value; the actual value is obtained from the root <b>Appointment</b> object.
<b>Saturday</b>	Specifies Saturday.

<b>Sunday</b>	Specifies Sunday.
<b>Thursday</b>	Specifies Thursday.
<b>Tuesday</b>	Specifies Tuesday.
<b>Wednesday</b>	Specifies Wednesday.
<b>WeekendDays</b>	Specifies Saturday and Sunday (or what ever days according to the settings of the C1Schedule).
<b>WorkDays</b>	Specifies work days (all days except weekend).

## Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

**C1.Web.Wijmo.Controls.C1EventsCalendar.RecurrencePattern.WeekDaysEnum**

## See Also

### Reference

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)

## RecurrencePattern.WeekOfMonthEnum

Determines the week in a month in which the appointment will occur.

## Syntax

Visual Basic (Declaration)	
<pre>Public Enum RecurrencePattern.WeekOfMonthEnum     Inherits System.Enum</pre>	
C#	
<pre>public enum RecurrencePattern.WeekOfMonthEnum : System.Enum</pre>	

## Members

Member	Description
<b>First</b>	The recurring appointment will occur on the specified day or days of the first week in the month.
<b>Fourth</b>	The recurring appointment will occur on the specified day or days of the fourth week in the month.
<b>Last</b>	The recurring appointment will occur on the specified day or days of the last week in the month.
<b>Second</b>	The recurring appointment will occur on the specified day or days of the second week in the month.
<b>Third</b>	The recurring appointment will occur on the specified day or days of the third week in the month.

## Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

**C1.Web.Wijmo.Controls.C1EventsCalendar.RecurrencePattern.WeekOfMonthEnum**

## See Also

### Reference

[C1.Web.Wijmo.Controls.C1EventsCalendar Namespace](#)