
ComponentOne

Menu 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
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	10
Help with ASP.NET Web Forms Edition	10
Key Features.....	10
Top Tips.....	11
Quick Start	11
Step 1 of 3: Adding C1Menu to the Page.....	12
Step 2 of 3: Populating the Menu with a SiteMap.....	12
Step 3 of 3: Running the Project	13
Design-Time Support	14
Smart Tag	15
Designer Form.....	16
Designer Form Context Menu.....	18
Bindings Collection Editor.....	19
Menu Types	20
Top-level menu	20
Drop-down Menu	20
Group Menu.....	21
Flyout Menu.....	21
Sliding menu	22
Menu Creation	23
Static Menu Creation	23
Dynamic Menu Creation	24
Data Source Menu Creation.....	24
Appearance	25
Themes.....	25
Menu Item Icons	27
Templates	27
Menu Navigation and Shortcuts	29
CSS Selectors.....	29
Working with the Client-Side	30
Client-Side Events	30
Task-Based Help.....	31

Creating a C1Menu Control in Code	32
Working with Themes	33
Using a Built-In Theme	34
Using a Custom Theme	36
Working with Templates.....	38
Creating an Individual Item Template	38
Creating an ItemsTemplate.....	39
Creating a Child Items Template.....	40
Creating a Top-Level Item Template.....	42
Working with CSS Selectors	43
Customizing Appearance with CSS Selectors.....	43
Customizing Link Appearance with CSS Selectors.....	45
Displaying a C1Menu Control as a Context Menu	46
Adding a Top-Level Item to a Menu.....	47
Creating a Drop-Down Menu.....	49
Creating a Sliding Menu	51
Animating C1Menu	54
Changing Menu Item Triggers.....	55
Adding Icons to C1Menu Items.....	56
Setting the Position of the Context Menu	57
Item Functions	59
Dynamically Adding Items to C1Menu.....	66
Populating C1Menu with a Site Map	68
Populating C1Menu with XML	71
Saving and Loading a C1Menu from XML	74
Client-Side Reference.....	75
Using the Wijmo CDN	75
C1.Web.Wijmo.Controls.4 Assembly.....	77
Namespaces.....	77
C1.Web.Wijmo.Controls Namespace.....	77
Overview	77
Classes.....	78
C1BaseStateManager.....	78
Overview	79

Members	79
C1TargetControlBase	79
Overview	80
Members	81
Methods	86
GetProperty<V> Method	88
GetScriptDescriptors Method	88
GetScriptReferences Method	89
RegisterDesignTimeStyleSheet Method	90
SetProperty<V> Method	90
Properties	91
CDNDependencyPaths Property	94
CDNPath Property	95
DisplayVisible Property	96
EnableCombinedJavaScripts Property	97
Enabled Property	98
Theme Property	99
ThemeSwatch Property	100
UseCDN Property	101
WijmoControlMode Property	101
WijmoCssAdapter Property	102
C1TargetControlHelper	103
Overview	104
Members	104
C1TargetControlHelper Constructor	105
Methods	105
ResolveEmbeddedResourceUrl Method	106
C1TargetControlHelper<T>	106
Overview	107
Members	108
C1TargetControlHelper<T> Constructor	109
Methods	110
EnsureEnabledState Method	111
GetScriptDescriptors Method	111

GetScriptDescriptors() Method.....	112
GetScriptDescriptors(String) Method	112
RegisterDesignTimeStyleSheet Method	113
RegisterDesignTimeStyleSheets Method.....	114
RegisterIncludes Method.....	115
RegisterOnSubmitStatement Method	115
RegisterRunTimeStylesheets Method.....	115
RegisterScriptDescriptors Method.....	116
RegisterScriptDescriptorsWithoutScriptManager Method	116
RegisterScriptDescriptorsWithScriptManager Method	117
RenderJsonDataField Method	117
ShowAbout Method.....	118
Properties.....	118
Control Property	119
JsonSerializableHelper Property	119
Page Property.....	119
ScriptManager Property.....	120
WijChartNavigator	120
Overview	121
Members.....	121
WijChartNavigator Constructor	122
C1.Web.Wijmo.Controls.C1Menu Namespace.....	122
Overview	122
Classes.....	124
C1Menu.....	124
Overview	125
Members.....	127
C1Menu Constructor.....	134
Methods.....	134
LoadLayout Method.....	136
LoadLayout(String) Method	136
LoadLayout(Stream) Method.....	137
LoadLayout(String,LayoutType) Method	137
LoadLayout(Stream,LayoutType) Method	138

OnItemClick Method	139
RenderBeginTag Method	140
RenderEndTag Method	140
SaveLayout Method	141
SaveLayout(String) Method	142
SaveLayout(Stream) Method	142
Properties.....	143
Animation Property.....	148
AutoPostBack Property	149
BackLink Property	150
BackLinkText Property	150
Checkable Property.....	151
ChildItemsTemplate Property.....	152
CrumbDefaultText Property.....	153
DataBindings Property	154
Direction Property.....	154
EnsureSubmenuOnBody Property	155
Height Property.....	155
HideAnimation Property	156
HideDelay Property	157
IsDirty Property	158
Items Property	158
ItemsTemplate Property	159
MaxHeight Property.....	160
Mode Property.....	161
OnClientBlur Property.....	162
OnClientFocus Property	163
OnClientHidden Property.....	163
OnClientHidding Property.....	164
OnClientSelect Property.....	165
OnClientShowing Property.....	166
OnClientShown Property	166
Orientation Property.....	167
Owner Property	168

Position Property	168
ShowAnimation Property.....	169
ShowDelay Property	170
SlidingAnimation Property.....	171
SubMenuTriggerEvent Property	172
TopItemsTemplate Property	173
TopLinkText Property.....	174
Trigger Property	174
TriggerEvent Property.....	175
Width Property	176
Events.....	176
ItemClick Event	177
C1MenuEventArgs	178
Overview	179
Members.....	180
C1MenuEventArgs Constructor	180
C1MenuEventArgs Constructor(C1MenuItem).....	181
C1MenuEventArgs Constructor(C1MenuItem,Object,CommandEventArgs) ...	181
Properties.....	182
CommandSource Property.....	183
Item Property	183
C1MenuItem	184
Overview	185
Members.....	186
C1MenuItem Constructor	190
Methods.....	191
C1DeserializeItems Method.....	192
RenderBeginTag Method	193
Properties.....	193
Enabled Property	196
Header Property.....	197
IconClass Property	197
ImagePosition Property	198
Items Property	199

NavigateUrl Property	200
Owner Property	201
Selected Property.....	201
Separator Property	202
Target Property	202
Template Property	203
TemplateContainer Property	204
Text Property	205
Value Property	206
C1MenuItemBinding.....	206
Overview	207
Members.....	208
C1MenuItemBinding Constructor	209
C1MenuItemBinding Constructor().....	209
C1MenuItemBinding Constructor(String)	210
Properties.....	211
DataMember Property.....	212
Depth Property	212
FormatString Property	213
HeaderField Property.....	214
ImageUrlField Property.....	214
NavigateUrlField Property	215
SeparatorField Property.....	216
TextField Property.....	217
ValueField Property	218
C1MenuItemBindingCollection	219
Overview	219
Members.....	220
C1MenuItemCollection	221
Overview	222
Members.....	223
C1MenuItemCollection Constructor	224
Methods.....	225
Add Method	226

Clear Method	227
Insert Method	227
Remove Method	228
RemoveAt Method.....	228
Properties.....	229
Events Property.....	230
Events.....	230
OnCollectionChanged Event	230
C1MenuItemCollectionChangedEventArgs.....	231
Overview	231
Members.....	232
C1MenuItemCollectionChangedEventArgs Constructor	233
Properties.....	233
Item Property	234
Status Property	234
C1MenuItemTemplateContainer	235
Overview	236
Members.....	237
C1MenuItemTemplateContainer Constructor	240
Properties.....	241
DataItem Property	242
ItemIndex Property	243
C1MenuSerializer	243
Overview	244
Members.....	244
C1MenuSerializer Constructor	245
Enumerations.....	245
ChangeStatus	245
ImagePosition	246
MenuMode	247
SubmenuTriggerEvent	248
TriggerEvent.....	249
Interfaces	250
IC1MenuItemCollectionOwner	250

Overview	250
Members.....	251
Properties.....	251
Items Property	252
Owner Property	252
Delegates	253
C1MenuEventHandler.....	253

Overview

Create multi-level menus with animation effects, image and check box items, interactive item scrolling, and more. You can even create a pop-up menu for context help within your application. **Menu for ASP.NET Web Forms** makes it easy with design-time support and client-side API. **Getting Started**

To get started, review the following topics:

- [Quick Start](#)
- [Menu Types](#)
- [Appearance](#)
- [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 ASP.NET Web Forms Edition](#).

Key Features

The **C1Menu** control includes several unique features, including the following:

- **Vertical and Horizontal Menus**
Menus and its submenus can be rendered either horizontally or vertically and include other layout options.
- **Scrolling**
C1Menu can scroll top level menus, sub menus, and sub groups. You can set the scroll mode option to button click scrolling, button hover scrolling, edge hover scrolling, or scroll bars.
- **Menu Item Icons**
Menu items can have their own icons. Choose from the many built-in icons or add your own icon to the menu.
- **Animation**
Menu supports different expand and collapse animation effects. For example, fade in, scroll in from the top, open horizontally, bounce, and more.
- **Overlay Flash, ActiveX, and Windowed Objects**
Unlike other menu controls, **C1Menu** can overlay any windowed objects, Flash, ActiveX and other standard and 3rd party components so you're not limited in your website design.

- **Data Binding Support**

Bind the **C1Menu** control to a data source you can bind to an XML data source or SiteMap data source, or you can even read data from Access data source and create the **C1Menu** hierarchy dynamically.

- **Keyboard Support**

Add access key support to give the **C1Menu** control focus with a chosen key combination. This enables end-users to use the keyboard arrow keys to navigate through the menu and menu items and the ENTER key to open a link in a menu item.

- **Theming**

With just a click of the SmartTag, change the menu's look by selecting one of the 5 premium themes (Midnight, Aristo, Rocket, Cobalt, and Sterling). Optionally, useThemeRollerfrom jQuery UI to create a customized theme!

- **CSS Support**

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

Top Tips

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

Tip 1: Prevent poor page rendering in quirks mode by editing the meta tag to fix rendering.

If a user's browser is rendering a page in quirks mode, widgets and controls may not appear correctly on the page. This is indicated by a broken page icon in the address bar. In

Compatibility View, the browser uses an older rendering engine.



Users can set this view that causes the issue. To prevent rendering in quirks mode, you can force the page to render with the latest browser. Add the following meta tag to the header of the page:

```
<meta http-equiv="X-UA-Compatible" content="IE=edge, chrome=1" />
```

Quick Start

The **C1Menu** Quick Start describes how to get started with the ASP.NET control, **C1Menu**. In the quick start you'll create an ASP.NET Web Site, add a **C1Menu** control to the page, apply different binding methods, add menu items and sub-menu items using the editor, change the menu's orientation, and more!

We have made it easy for you so you can jump to any topic you like since they don't follow chronological order.

See Also

[Step 1 of 3: Adding C1Menu to the Page](#)

[Step 2 of 3: Populating the Menu with a SiteMap](#)

[Step 3 of 3: Running the Project](#)

Step 1 of 3: Adding C1Menu to the Page

In this lesson you will learn how to create a new ASP.NET Web site and add a C1Menu control to your project. To begin the Quick Start, complete the following steps:

1. Begin by creating a new Web Site. Note that as you've created an AJAX-Enabled Web site, a **ScriptManager** control initially appears on the page.
2. While in Design view navigate to the Visual Studio Toolbox and double-click the **C1Menu** icon to add the **C1Menu** control to your page.

The page will appear similar to the following:



Step 2 of 3: Populating the Menu with a SiteMap

In this section of the quick start, you'll learn how to bind C1Menu to a SiteMapDataSource.

To create an XML file and bind it to **C1Menu**, add the **XMLDataSource** component to the Web site, and then assign it to the **C1Menu** control.

1. Start a new **AJAX 1.0-Enabled ASP.NET 2.0 Web Site** project.
2. Right-click on the **App_Data** in the Solution Explorer and select **Add New Item**. The **Add New Item** dialog box appears.
3. Select the XML File and rename it "Menu.xml".
4. Click on the **Add** button in the **Add New Item** dialog box.
5. Switch to the XML view and add the following data to **Menu.xml**:

[To write code in Source View](#)

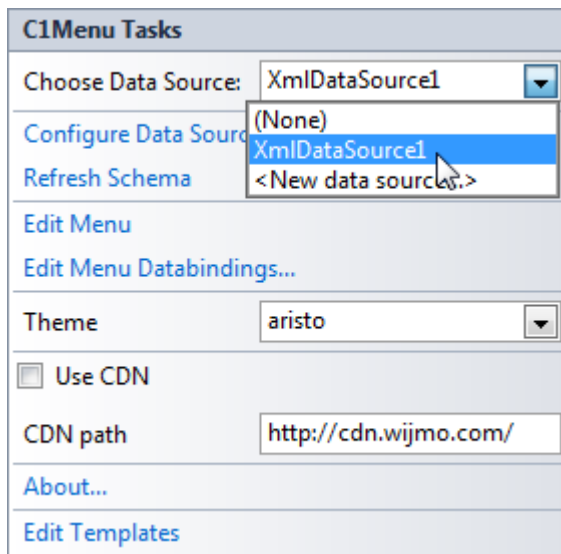
```
<root>
  <menuitem Text="Home">
  </menuitem>
  <menuitem Text="Products">
    <menuitem Text="Hardware">
```

```

    </menuItem>
    <menuItem Text="Software">
    </menuItem>
  </menuItem>
  <menuItem Text="Services">
    <menuItem Text="Training">
    </menuItem>
    <menuItem Text="Consulting">
    </menuItem>
  </menuItem>
</root>

```

6. Switch back to the .aspx page and select the **Design** tab to switch to the design view.
7. Expand the **Data** node in the Visual Studio Toolbox and double-click on the **XmlDataSource** component to add it to the Web page.
8. Select the **XmlDataSource1** on the Web Page and navigate to the **Properties** window.
9. Click on the ellipsis button next to the **DataFile** property to open the **Select XML File** dialog box. Select **App_Data** and click **Menu.xml** file. Click **OK** to add it to the **XmlDataSource1.DataFile** property.
10. Set the **XmlDataSource1.XPath** property to root/menuitem.
11. Click the smart tag to open the **C1Menu Tasks** menu and select **XmlDataSource1** from the **Choose Data Source** drop-down listbox

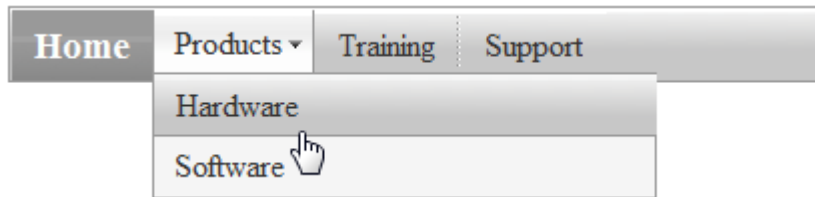


In the next step, you'll run the program and observe how the XML file populates the menu.

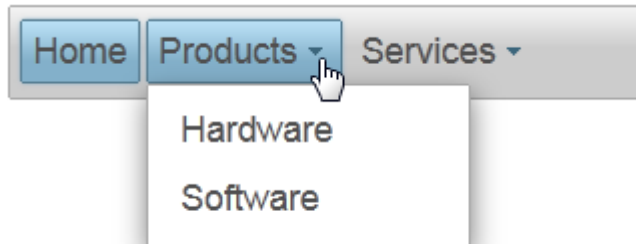
Step 3 of 3: Running the Project

In this section of the quick start, you'll run the project and see the result of populating a **C1Menu** control with an XML file.

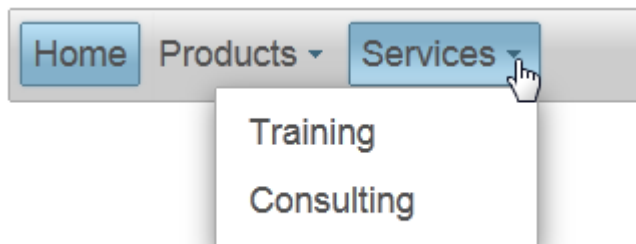
1. Save and build your project. Observe that there are three top-level menu items:



2. Hover your cursor over Products and observe that two items appear in its sub-menu:



3. Hover your cursor over Training and observe that two items appear in its sub-menu.



Congratulations! You have completed the Menu for ASP.NET Web Forms quick start.

Design-Time Support

C1Menu provides smart tags, designer, and a bindings collection editor that offers rich design-time support and simplifies working with the object model.

The following topics describe how to use C1Menus design-time environment to configure the C1Menu control.

See Also

[Smart Tag](#)

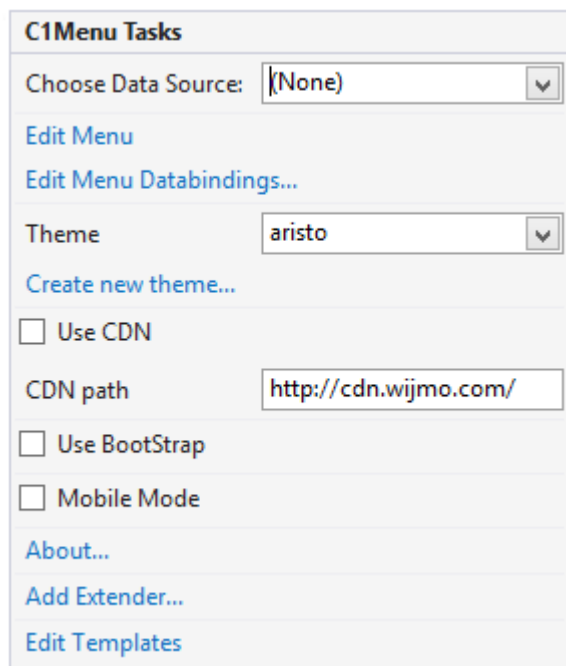
[Designer Form](#)

[Bindings Collection Editor](#)

Smart Tag

The **C1Menu** control includes a smart tag in Visual Studio. A smart tag represents a shortcut tasks menu that provides the most commonly used properties in **C1Menu**.

To access the **C1Menu Tasks** menu, click on the smart tag in the upper-right corner of the **C1Menu** control. This will open the **C1Menu Tasks** menu.

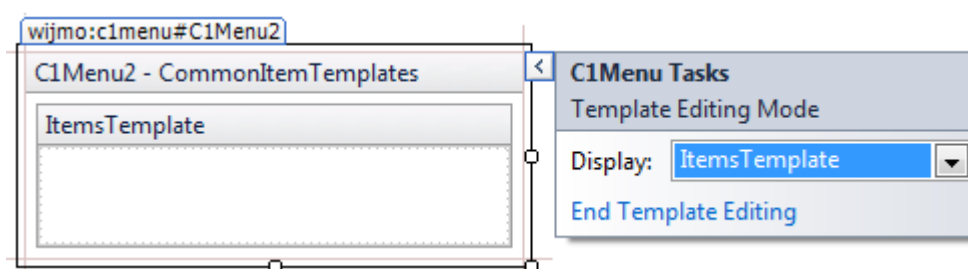


The **C1Menu Tasks** menu operates as follows:

- **Choose Data Source**
Clicking on the **Choose Data Source** item opens a drop-down list where you can choose an existing data source or select a new data source to bind to.
- **Edit Menu**
Clicking on the **Edit Menu** item opens the **C1Menu Designer Form** where you can quickly configure **C1Menu**'s elements without having to scroll through its Properties window. You can load and save the control's content and can add LinkItem, Header, Group, and Separator elements. For more information on the **C1 Menu Designer Form**, see [C1Menu Designer Form](#).
- **Edit Menu DataBindings**
Clicking on the **Edit Databindings** item opens the **Bindings Collection Editor** dialog box where you can add and remove bindings and edit properties.
- **Theme**
Clicking the **Theme** drop-down arrow enables you to select from different built-in skins.
- **Create new theme**
The **Create new theme** options opens the **ThemeRoller for Visual Studio**. This allows you to customize a theme without leaving your design environment. To find more

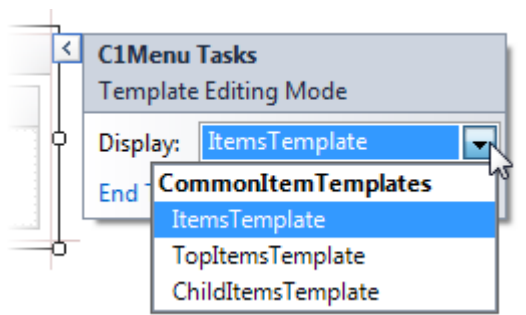
information on using **ThemeRoller** in your application, see [ThemeRoller for Visual Studio](#).

- **Use Bootstrap**
Selecting **Use Bootstrap** applies Bootstrap theming to your control. To find more information on using **Bootstrap** theming in your application, see [Bootstrap Theming](#).
- **About**
Clicking the **About** item displays a dialog box, which is helpful in finding the version number of **ASP.NET Web Forms Edition** and online resources.
- **Edit Templates**
Clicking on the **Edit Templates** item switches the **C1Menu** control to **Template Editing Mode**:



In Template Editing Mode, the **C1Menu Tasks** menu appears with different options:

- **Display**
Selecting the **Display** drop-down arrow will open a list of template areas that can be customized:



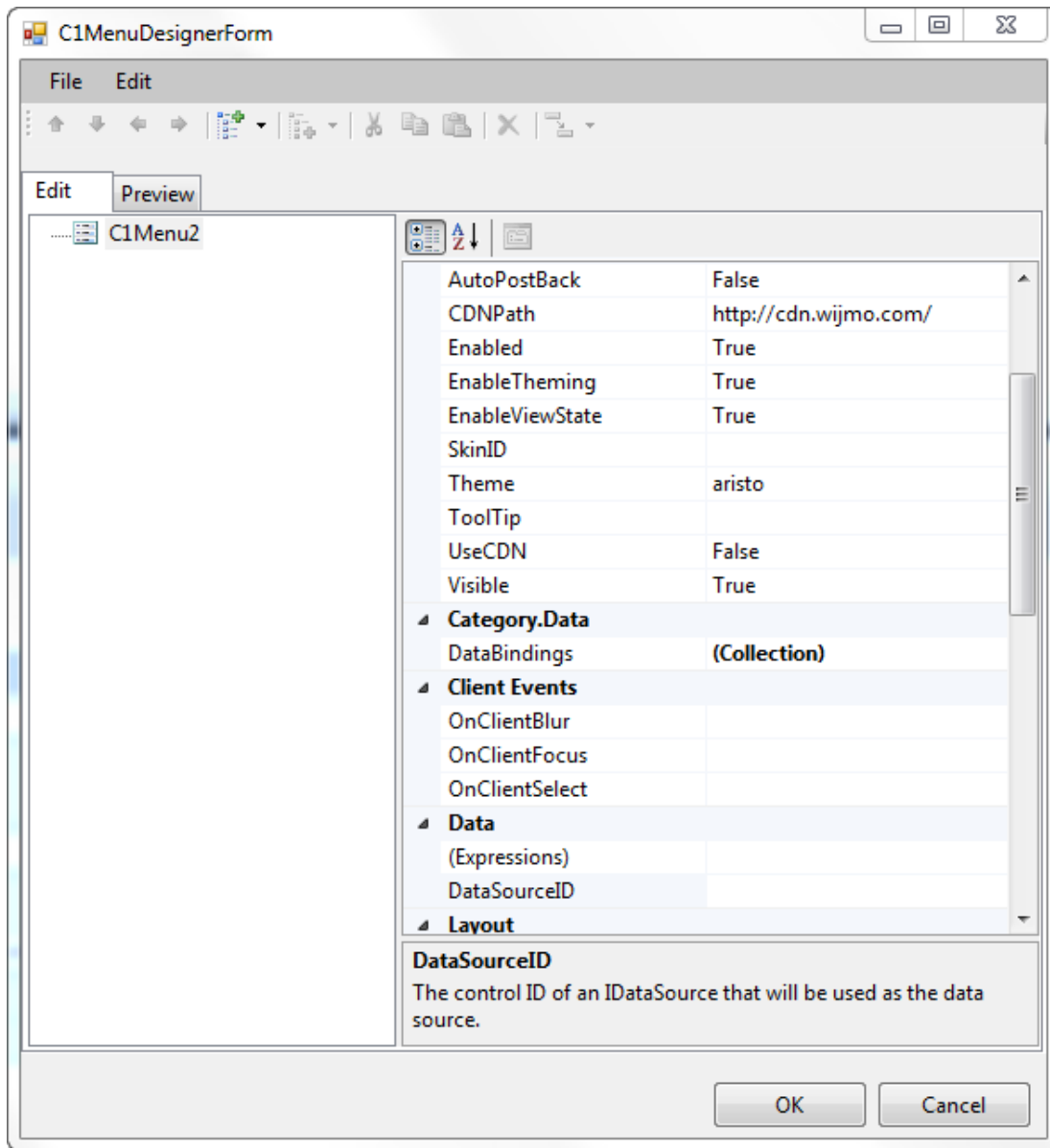
Select a template from this list to open that template to be edited.

- **End Template Editing**
Clicking the **End Template Editing** item will end Template Editing Mode and return you to the main **C1Menu Tasks** menu.

Designer Form

The **C1Menu Designer Form** dialog box lets you quickly configure **C1Menu**'s elements without having to scroll through the control's Properties window. Using the **C1Menu Designer Form** you can add, manipulate, and delete LinkItem, Header, Group, and Separator elements in the **C1Menu** control and load and save the control's content.

To access the **C1Menu Designer Form** dialog box select the **Edit Menu** item from the **C1Menu Tasks** menu (see [Smart Tag](#) for details) or right-click on the **C1Menu** control at design time and select **Edit Menu**. The designer, with items added, looks similar to the following:



The **C1Menu Designer Form** includes an **Edit** tab and a **Preview** tab. The **Edit** tab, pictured above, consists of a left pane listing the order of added menu items. The right side of the **Edit** tab consists of a properties grid allowing you to quickly customize added items.

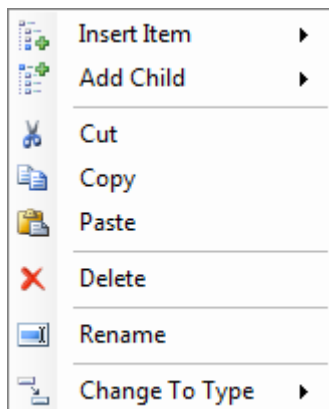
The **Preview** tab allows you to view the **C1Menu** control and quickly preview any changes you are making.

See Also
















[Designer Form Context Menu](#)






Designer Form Context Menu

The **C1Menu Designer Form** context menu lets you quickly configure **C1Menu**'s elements. Using the **C1Menu Designer Form** you can load and save the control's content and can add and remove LinkItem, Header, Group, and Separator elements. Access the **C1Menu Designer Form** context menu by right-clicking in the left pane of the **Edit** tab. The context menu will look similar to the following:



The following table describes the function of each item in the **C1Menu Designer Form** context menu.

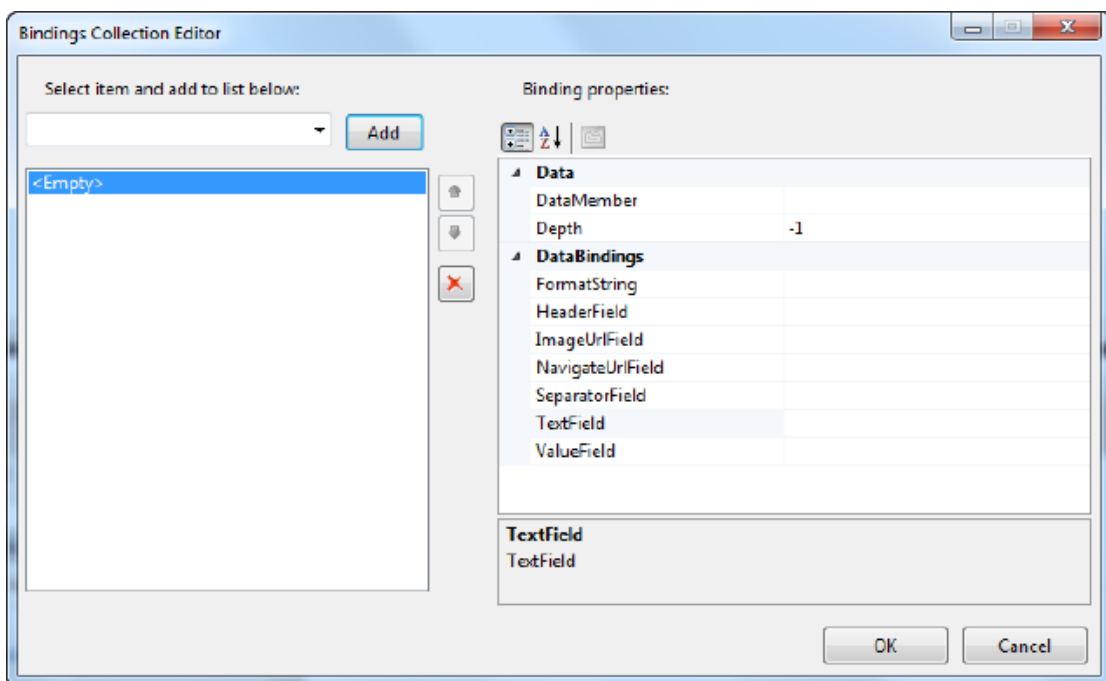
Button	Name	Description
	Insert Item	Inserts a new item in the menu. Choices include the LinkItem () , Header () , Group () , and Separator () elements.
	Add Child	Inserts a child item under the current item. Choices include the LinkItem () , Header () , Group () , and Separator () elements.
	Cut	Cuts the currently selected item.
	Copy	Copies the currently selected item.
	Paste	Pastes a cut or copied item to the selected location.
	Delete	Deletes the currently selected item.
	Rename	Renames the currently selected item. Select this option and type in a new name for the selected item.

	Change to Type	Changes the current item to an item of a different type. Choices include the LinkItem () , Header () , Group () , and Separator () elements.
---	----------------	---

Bindings Collection Editor

The **Bindings Collection Editor** dialog box lets you easily define the relationship between a data item and the menu item it is bound to. You can access the **Bindings Collection Editor** dialog box by selecting the **ellipses** button next to the C1Menu.DataBindings property in the Properties window, or by selecting the **Edit Databindings** item from the **C1Menu Tasks** menu.

The **Bindings Collection Editor** dialog box will appear similar to the following:



The **Bindings Collection Editor** consists of a drop-down box where you can choose an existing item to add, a left-side list box listing added items, and a right-side properties grid where you can change the data and databinding properties for that item.

Note that any changes you make in the **Bindings Collection Editor** will be reflected in the `<DataBindings>` tag in the Source view, for example:

To write code in Source View

```
<c1:clmenu id="C1Menu1" runat="server"
datasourceid="SiteMapDataSource1" visualstyle="Default"
visualstylepath="~/C1WebControls/C1Menu/VisualStyles">
<DataBindings>
<c1:C1MenuItemBinding DataMember="SiteMapNode" Depth="5"
```

```
NestedGroupHeight="" NestedGroupWidth="" />  
</DataBindings>  
</cc1:clmenu>
```

Menu Types

Typically top-level or submenus are used to create a navigation system, but the [C1Menu](#) control extends the menu functionality to create menus with groups so items can be organized into categories

This section details the common types of menus you can create using the [C1Menu](#) control.

See Also

[Top-level menu](#)

[Drop-down Menu](#)

[Group Menu](#)

[Flyout Menu](#)

[Sliding menu](#)

Top-level menu

The top level menu is the main menu. It consists of menu items which are arranged on a horizontal or vertical menu bar. Each menu item may or may not contain a list of submenu items. A top-level menu is always visible on the form. Typically top-level menus have 15 menu items or less.

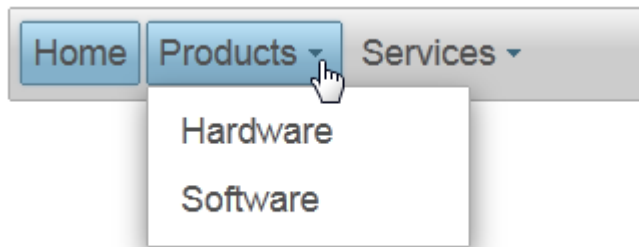
A typical top-level menu looks like the following:



Drop-down Menu

The menu bar contains all of the drop-down menus and submenus in your application. Each menu in the menu bar is represented by its menu title.

Submenus, or drop-down menus, are additional menu items that appear within a menu item. Submenu items appear when you hover or click on the menu item on the top-level menu or submenu level. Depending on the style of the menu, an arrow or some other graphic is used to indicate when the menu item opens the submenu items. A submenu can have a submenu and so on.

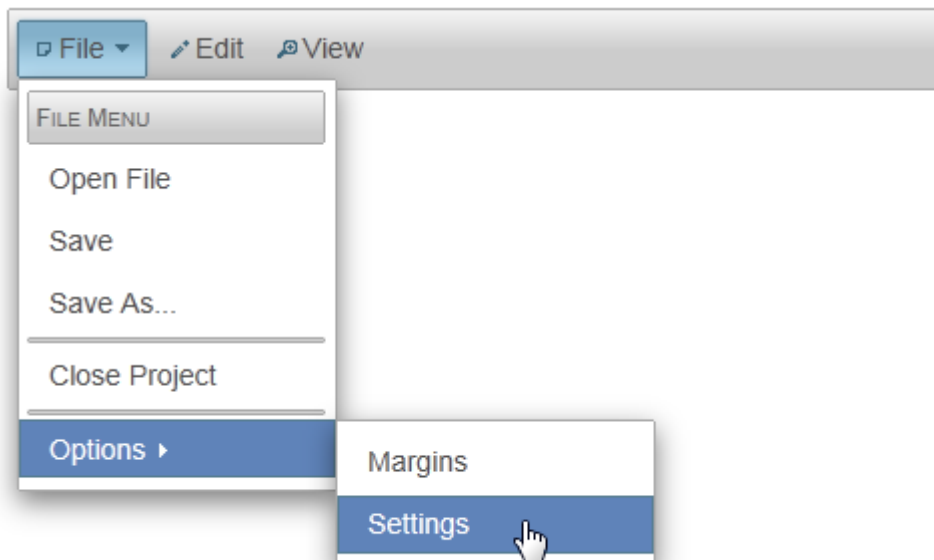


Group Menu

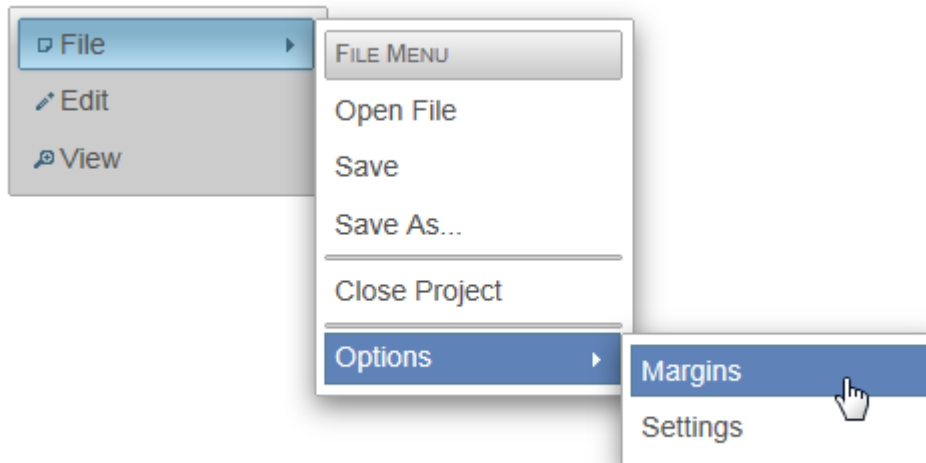
Grouped menus are multiple sets of menu items organized into one or more categories. The menu items in a group share a common function. Each group may include a heading item type that represents the category name for the group. Typically group menus appear as drop-down menus with at least one heading item to label the grouped menu items. Groupings can be separated by heading items or separator items.

Flyout Menu

By default, the type of menu displayed is a flyout menu in a horizontal position. This is the basic menu that you see atop most applications.

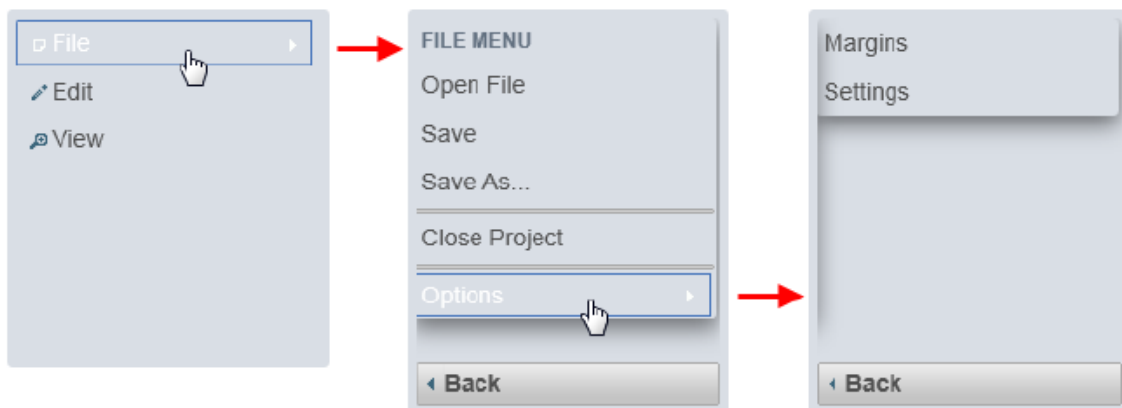


The flyout menu can also be created in a vertical position by setting the [C1Menu](#) control's [Orientation](#) property to Vertical.

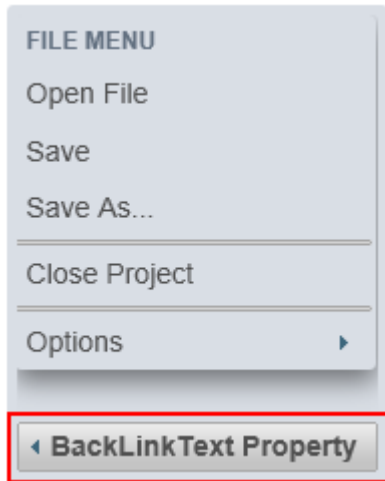


Sliding menu

A sliding menu is a type of menu that is used primarily on smartphone devices. Instead of having a submenu fly out when the user clicks or hovers over a menu item, submenus will slide into view, replacing the previous menu.



Note that the above picture shows a "Back" button on the two submenus. A sliding menu, by default, will apply a "Back" navigation button to your submenus. If you'd like, you can change the string of the "Back" button by setting the [C1Menu](#) control's [BackLinkText](#) property.



You may also use breadcrumb navigation in lieu of the "Back" button by setting the **C1Menu** control's [BackLink](#) property to **False**.

Menu Creation

A menu system can be created using one of the following methods:

- Static creation using declarative syntax
- Dynamic creation using a constructor to create new instances of the [C1MenuItem](#) class.
- Data source creation through binding **C1Menu** to a **SiteMapDataSource**, **XMLDataSource**, or an **AccessDataSource**.

See Also

[Static Menu Creation](#)

[Dynamic Menu Creation](#)

[Data Source Menu Creation](#)

Static Menu Creation

A static menu is the simplest way to create the menu structure.

You can use the **C1Menu Designer Form** designer to build the menu system or you can use declarative syntax in the .aspx file to specify the menu items. To display static menu items using declarative syntax, first nest opening and closing `<Item>` tags between opening and closing tags of the Menu control. Next, create the menu structure by nesting `<cc1:C1MenuItem>` elements between opening and closing `<Items>` tags. Each `<cc1:C1MenuItem>` element represents a menu item in the control and maps to a `C1MenuItem` object.

[To write code in Source View](#)

```
<HideAnimation>
<Animated Effect="fade"></Animated>
</HideAnimation>
  <Items>
    <cc1:C1MenuItem runat="server" ImagePosition="Left"
Text="LinkItem1">
      <Items>
        <cc1:C1MenuItem runat="server"
ImagePosition="Left" Text="LinkItem1">
          </cc1:C1MenuItem>
        <cc1:C1MenuItem runat="server"
ImagePosition="Left" Text="LinkItem2">
          </cc1:C1MenuItem>
        </Items>
      </cc1:C1MenuItem>
    <cc1:C1MenuItem runat="server" ImagePosition="Left"
Text="LinkItem2">
      <Items>
        <cc1:C1MenuItem runat="server"
ImagePosition="Left" Text="LinkItem1">
          </cc1:C1MenuItem>
        <cc1:C1MenuItem runat="server"
ImagePosition="Left" Text="LinkItem2">
          </cc1:C1MenuItem>
        </Items>
      </cc1:C1MenuItem>
    </Items>
  </cc1:C1Menu>
```

Dynamic Menu Creation

Dynamic menus can be created on the server side or client side. When creating dynamic menus on the server side, use a constructor to dynamically create a new instance of the [C1Menu Item](#) class. For client-side, the **CreateInstance** constructor can be used to dynamically create a new instance of the [C1Menu](#) control. For example the follow script creates a new menu control on the client side:

[To write code in Source View](#)

```
aMenu = C1.Wijmo.Controls.C1Menu.createInstance ();
document.body.appendChild(aMenu.element);
```

Data Source Menu Creation

Menu items can be created from a hierarchal data source control such as an **XMLDataSource** or **SiteMapDataSource**. This allows you to update the menu items without having to edit code. Menu items can also be bound to a non-hierarchal control such as an **AccessDataSource** component.

See [Populating C1Menu with a Site Map](#) and [Populating C1Menu with XML](#) for tutorials about populating the [C1Menu](#) control with a data source.

Appearance

The following topics illustrate features that will modify the appearance of the C1Menu control.

See Also

[Themes](#)

[Menu Item Icons](#)

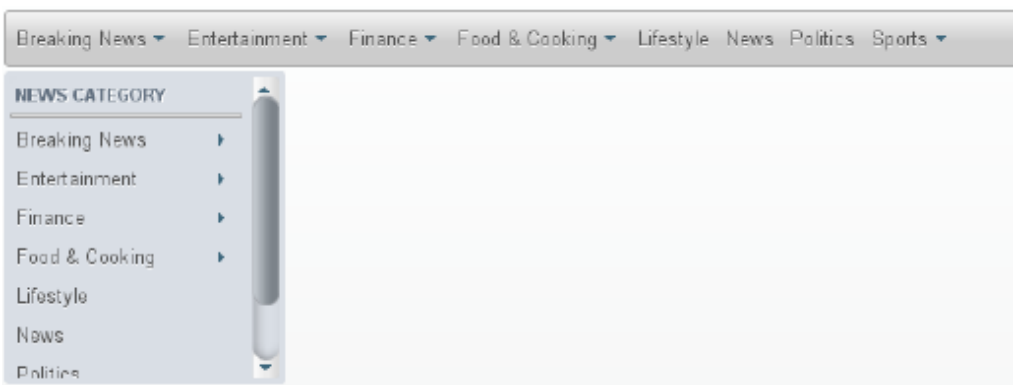

[Templates](#)

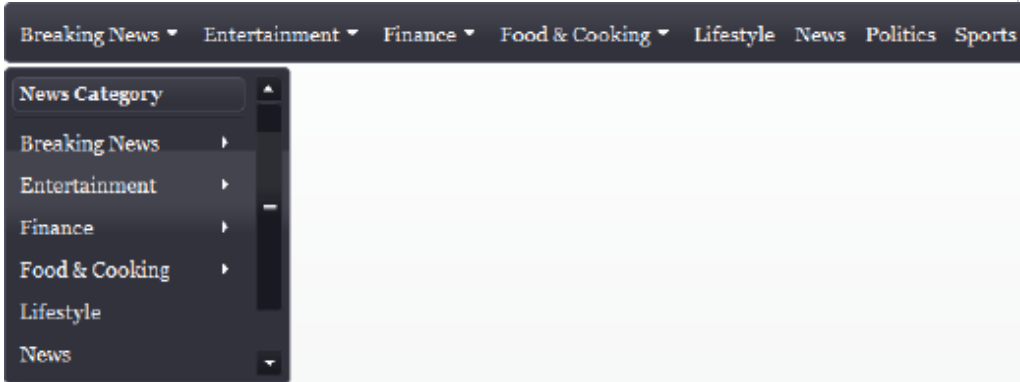
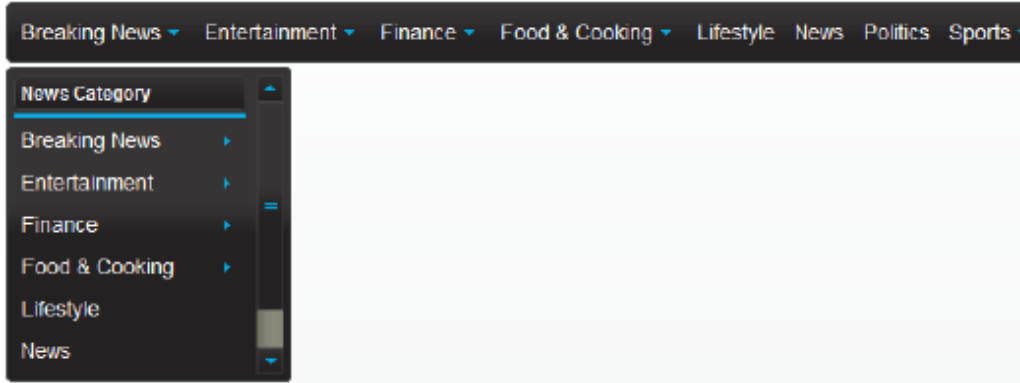

[Menu Navigation and Shortcuts](#)

[CSS Selectors](#)

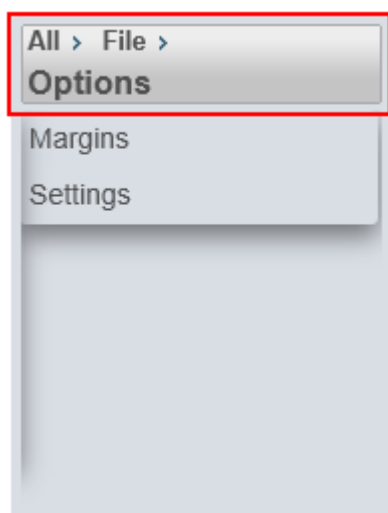
Themes

The [C1Menu](#) control contains five 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 **C1Menu** control as follows:

Aristo	 The Aristo theme features a light gray header bar with menu items: Breaking News, Entertainment, Finance, Food & Cooking, Lifestyle, News, Politics, and Sports. Below the header is a vertical list titled 'NEWS CATEGORY' containing the same items. The list has a light gray background and a vertical scrollbar on the right.
Cobalt	 The Cobalt theme features a blue header bar with menu items: Breaking News, Entertainment, Finance, Food & Cooking, Lifestyle, News, Politics, and Sports. Below the header is a vertical list titled 'News Category' containing the same items. The list has a blue background and a vertical scrollbar on the right.

Midnight	
Rocket	
Sterling	

To set the theme of the **C1Menu** control, simply set its **Theme** property to one of the built-in themes.



Menu Item Icons

You can easily add icons to individual menu items by setting that menu item's [IconClass](#) property. You can declare the jQuery UI "ui-icon" class and then add a second class to it describing the type of icon you'd want to use. The icon classes follow the following syntax:

```
ui-icon ui-icon-[icon name]
```

You can use any icon that's included in the jQuery UI set of framework icons. You can see a list of jQuery UI framework icons on the jQuery UI ThemeRoller page at <http://jqueryui.com/themeroller/>.

The default position for icons is to the left of a menu items text. If you'd like to change this, you can set the menu item's [ImagePosition](#) property to **Left**.

Templates

[C1Menu](#) includes several types of templates, from the basic per-item template to global templates, such as the [ChildItemsTemplate](#), which are used to control templates for entire portions of the control.

Individual Templates

Templates can be used to embed HTML controls in a menu item, or they can be used for something as simple as formatting. These templates can be created in Source view by adding `<Template>` tags to individual menu items. For example:

To write code in Source View

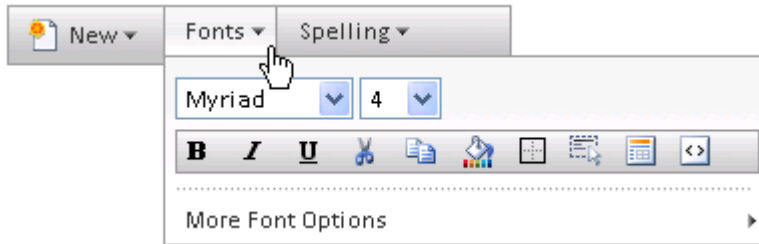
```
<Template>
  <div>
    <a href="#" class="wijmo-wijmenu-text">New</a><span
class="wijmo-wijmenu-icon-right">Ctrl+N</span>
  </div>
```

```
</Template>
</cc1:C1MenuItem>
```

If a global template, such as the [ItemsTemplate](#), is assigned to the **C1Menu** control, the individual template will override it.

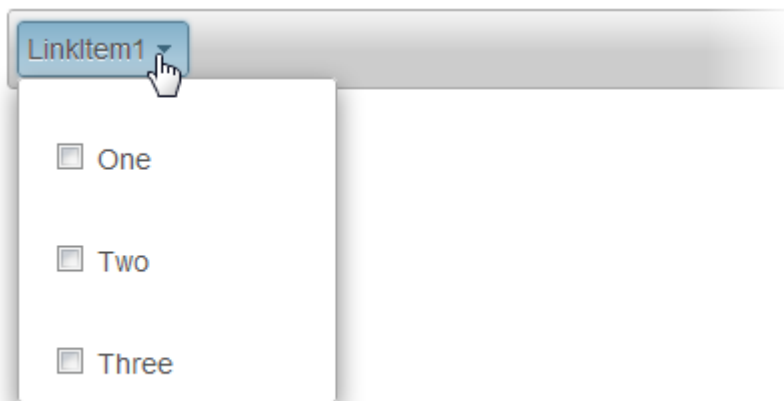
For a tutorial concerning individual templates, see [Creating an Individual Item Template](#).

Global Templates



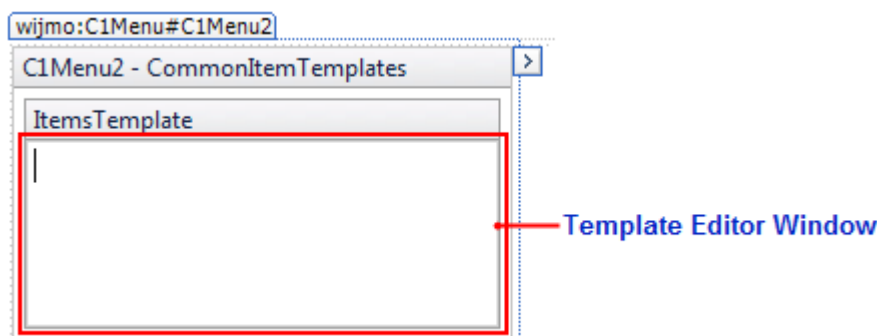
C1Menu also includes special template designers for customizing the top level menu items (the [TopItemsTemplate](#)), child menu items (the [ChildItemsTemplate](#)), or *all* menu items (the [ItemsTemplate](#)). Creating templates of different types allows you to control the design of all or even just a specific portion of your menu items. Templates are also useful as new item prototypes, which means that you can ensure dynamically added items adhere to design of other items in your menu.

The following **C1Menu** control has its **ChildItemsTemplate** customized to include an two HTML element, an Input (check box) control and a Label control, for each menu item:



To access a template:

1. Click the **C1Menu** control's smart tag to open the **C1Menu Tasks** menu. **Template Editing Mode** engages.
2. Click the Display drop-down arrow and select the type of template (**TopItemsTemplate**, **ChildItemsTemplate**, or **ItemsTemplate**) from the drop-down list.
3. Add HTML elements or formatting to the template designer window. (See [Working with Templates](#) for a tutorial.)



Menu Navigation and Shortcuts

Menu Navigation and Shortcuts

The two common approaches for navigating menu systems are:

- Point and click using the mouse
- Arrow keys using the keyboard

Mouse Navigation

When the user clicks a menu item, the Menu control can either navigate to a linked Web page or simply post back to the server. If the [C1MenuItem.NavigateUrl](#) property of a menu item is set, the Menu control navigates to the linked page; otherwise, it posts the page back to the server for processing. By default, a linked page is displayed in the same window or frame as the Menu control. To display the linked content in a different window or frame, use the [C1MenuItem.Target](#) property of the Menu control.

Keyboard Navigation

[C1Menu](#) supports access keys to enhance the users navigation of your menu items. Access keys are used for navigating through the menu and typically use the ALT key plus the specified key.

You can add access keys to any of the menu items and submenu items using the **AccessKey** property. One character should be used when you assign access keys to menu items.

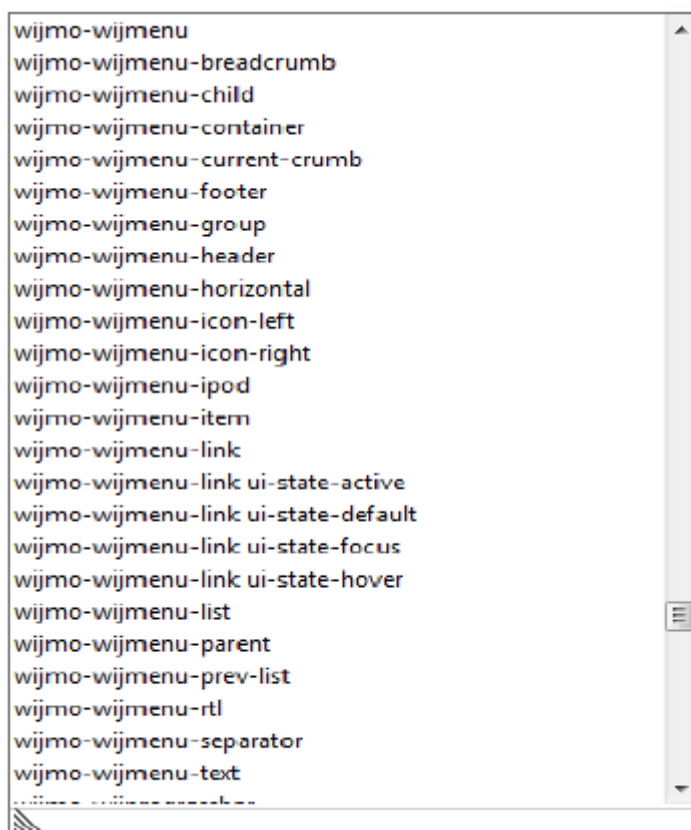
CSS Selectors

You can style many [C1Menu](#) elements using CSS to make their appearance unique. To make this customization easier, ComponentOne includes CSS selectors with each of its six built-in themes.

You can apply general CSS properties such as border, background, text, font, margin, padding, list, outline, and table to applicable CSS selectors.

For a list of common individual CSS selectors and grouped CSS selectors, select the **C1Menu** control in your project and view the drop-down list next to the **CssClass** property in the Visual Studio Properties window.

C1Menu CSS selectors begin with `wijmo-wijmenu`:



Working with the Client-Side

The Menu for ASP.NET Web Forms Edition 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 **C1Menu** control is rendered, an instance of the client-side control will be created automatically. This means that you can enjoy the convenience of accessing the properties and methods of the **C1Menu** controls 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. Using the client-side object model increases the efficiency of your Web site.

See Also

[Client-Side Events](#)

Client-Side Events

Menu for ASP.NET Web Forms includes several client-side events that allow you to manipulate the **C1Menu** controls when an action such as resizing the dialog window occurs.

You can use the server-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 "Focus" to respond when the Menu gains focus, you would set the [OnClientFocus](#) property to "focus".

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 of the name you declare for each JavaScript function are defined on the client side.

Event Server-Side Property Name	Event Name	Description
OnClientBlur	blur	Occurs when a menu item loses focus
OnClientFocus	focus	Occurs either on mouse hover or when the keyboard cursor keys are used for navigation
OnClientHidden	hidden	Occurs when the menu is closed.
OnClientHidding	hidding	Occurs when the menu is closing.
OnClientSelect	select	Occurs when a menu item has been selected.
OnClientShowing	showing	Occurs when the menu is opening.
OnClientShown	shown	Occurs when the menu is opened.

Task-Based Help

The task-based help section assumes that you are familiar with programming in the Visual Studio ASP.NET environment, and know how to use the **C1Menu** control in general. Each topic

provides a solution for specific tasks using the [C1Menu](#) control. Each task-based help topic also assumes that you have created a new ASP.NET project.

See Also

[Creating a C1Menu Control in Code](#)

[Working with Themes](#)

[Working with Templates](#)

[Working with CSS Selectors](#)

[Displaying a C1Menu Control as a Context Menu](#)

[Adding a Top-Level Item to a Menu](#)

[Creating a Drop-Down Menu](#)

[Creating a Sliding Menu](#)

[Animating C1Menu](#)

[Changing Menu Item Triggers](#)

[Adding Icons to C1Menu Items](#)

[Setting the Position of the Context Menu](#)

[Item Functions](#)

[Dynamically Adding Items to C1Menu](#)

[Populating C1Menu with a Site Map](#)

[Populating C1Menu with XML](#)

[Saving and Loading a C1Menu from XML](#)

Creating a C1Menu Control in Code

Creating a [C1Menu](#) control in code is an easy process. In this topic, you will add a **PlaceHolder** control to the page, add an import statement, add the **C1Menu** control, and add the control to the **PlaceHolder**.

Complete the following steps:

1. In Design View, navigate to the Visual Studio Toolbox and add a **PlaceHolder** control to the page.
2. Double-click the page to add a **Page_Load** event and switch to Code view.
3. Add the following statement to the top of the Code Editor to import the appropriate namespace.

To write code in Visual Basic

Visual Basic	Copy Code
<pre>Imports C1.Web.Wijmo.Controls.C1Menu</pre>	

To write code in C#

C#	Copy Code
<pre>using C1.Web.Wijmo.Controls.C1Menu;</pre>	

4. Add the following code to create the C1Menu control and add the control to the Placeholder.

To write code in Visual Basic

Visual Basic	Copy Code
<pre>'Create a new C1Menu control Dim C1M As New C1Menu() 'Add C1Menu to the Placeholder control Placeholder1.Controls.Add(C1M)</pre>	

To write code in C#

C#	Copy Code
<pre>//Create a new C1Menu control C1Menu C1M = new C1Menu(); //Add C1Menu to the Placeholder control Placeholder1.Controls.Add(C1M);</pre>	

5. Press F5 to run your program. You have created a basic C1Menu control in code.

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 a Custom Theme](#)

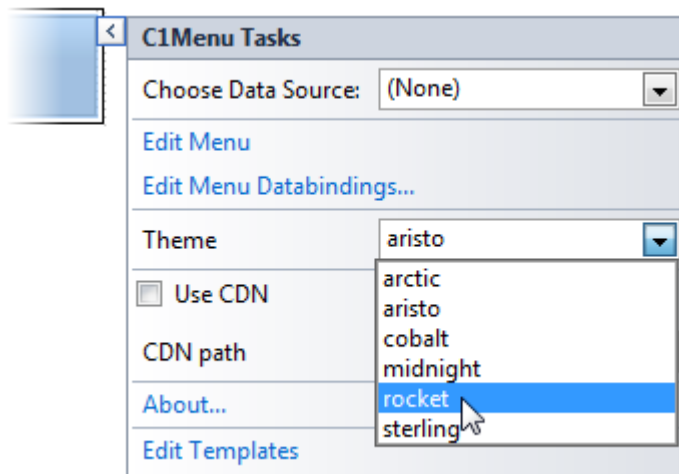
Using a Built-In Theme

A **C1Menu** 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, in Source view, and in code. For more information on themes, see [C1Menu Themes](#).

Changing the Theme in Design View

Complete the following steps:

1. Click the **C1Menu** smart tag  to open the **C1Menu 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 **C1Menu** control.

Changing the Theme in Source View

To change the theme of your **C1Menu** in Source view, add `VisualStyle="rocket"` to the `<cc1:C1Menu>` tag so that it resembles the following:

[To write code in Source View](#)

```
<cc1:C1Menu ID="C1Menu1" runat="server" Theme="rocket"/>
```

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>C1Menu1.Theme = "rocket"</pre>	

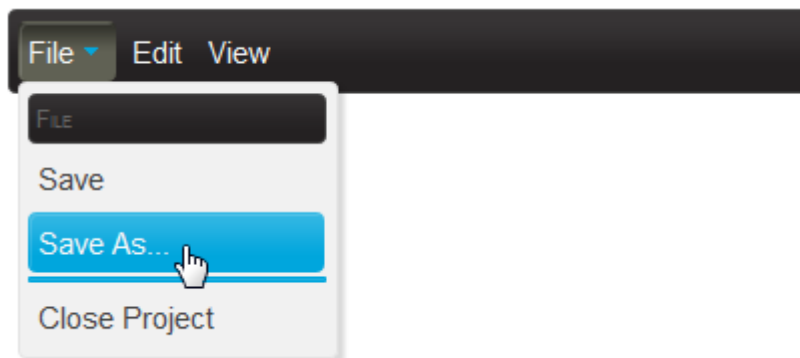
To write code in C#

C#	Copy Code
<pre>C1Menu1.Theme = "rocket";</pre>	

3. Run the program.

✔ **This topic illustrates the following:**

The following image shows a **C1Menu** control with the **rocket** theme:



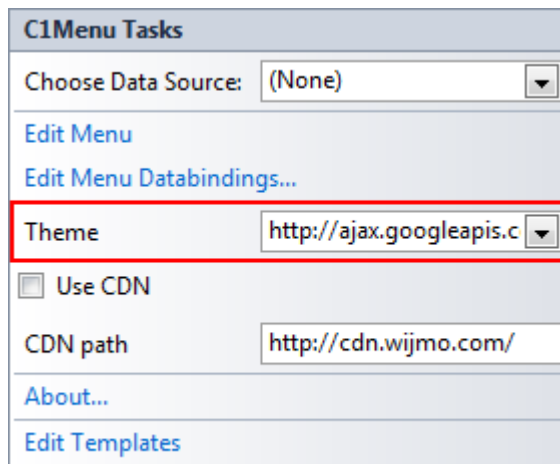
Using a Custom Theme

Tabs 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 **C1Menu** in the following examples.

Using a CDN URL

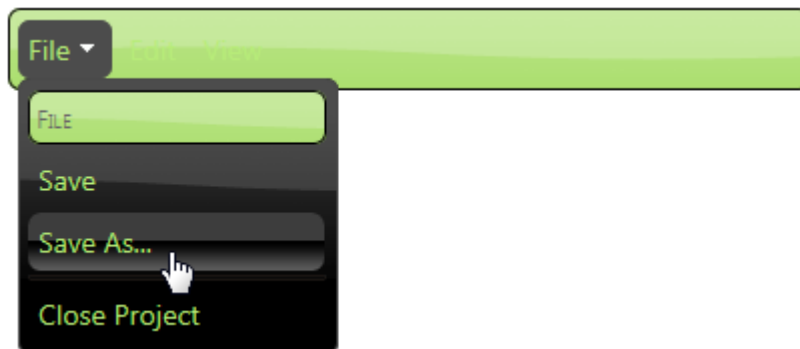
Complete the following steps:

1. Click the **C1Menu** smart tag to open the **C1Menu 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/2011/06/jquery-ui-1-8-14/>. In this example, we'll use the *trontastic* theme:
<http://ajax.googleapis.com/ajax/libs/jqueryui/1.8.14/themes/trontastic/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.

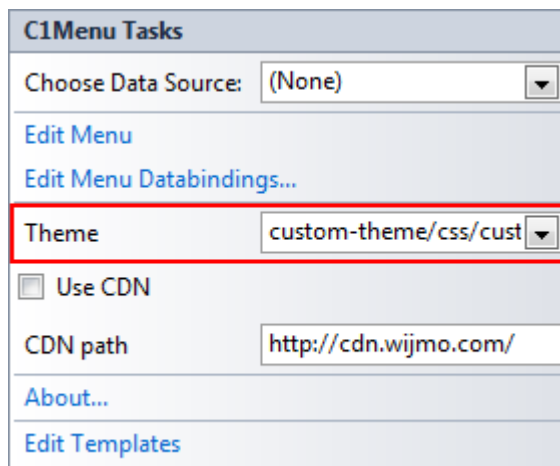
4. Run the project and notice the theme is applied to **C1Menu**.



Using jQuery ThemeRoller

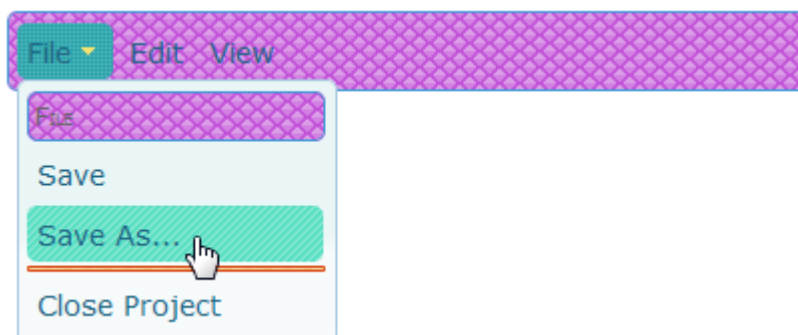
Complete the following steps:

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 **C1Menu** 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-1.8.15.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 **C1Menu**.



Working with Templates

This section contains tutorials that will teach you how to use the [ChildItemsTemplate](#), [ItemsTemplate](#), and the [TopItemsTemplate](#).

See Also

[Creating an Individual Item Template](#)

[Creating an ItemsTemplate](#)

[Creating a Child Items Template](#)

[Creating a Top-Level Item Template](#)

Creating an Individual Item Template

In this tutorial, you will learn how to create a template for an individual menu item.



Note: Individual templates trump global templates, so adding a template to an individual item will preventing formats that you may have used in [ItemsTemplates](#), [ChildItemsTemplates](#), and [TopItemsTemplates](#).

Complete the following steps:

1. Create a [C1Menu](#) control with one top-level menu item. Add two child items to that top-level menu item to create a submenu.
2. Switch to Source view and add the following markup between the `<cc1:C1MenuItem ID="C1MenuItem1">` tags:

[To write code in Source View](#)

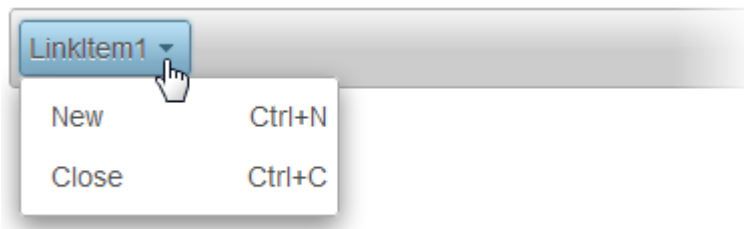
```
<Template>
  <div>
    <a href="#" class="wijmo-wijmenu-text">New</a><span
class="wijmo-wijmenu-icon-right">Ctrl+N</span>    </div>
</Template>
```

3. Now add the following markup between the `<cc1:C1MenuItem ID="C1MenuItem1">` tags:

[To write code in Source View](#)

```
<Template>
  <div>
    <a href="#" class="wijmo-wijmenu-text">Close</a><span
class="wijmo-wijmenu-icon-right">Ctrl+C</span>    </div>
</Template>
```


- Press F5 to run the project. When the project loads, click the top-level menu item to open its submenu and observe that the two menu items have adopted the templates you specified.

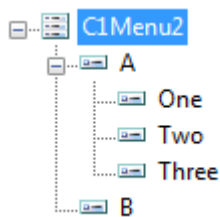



Creating an ItemsTemplate

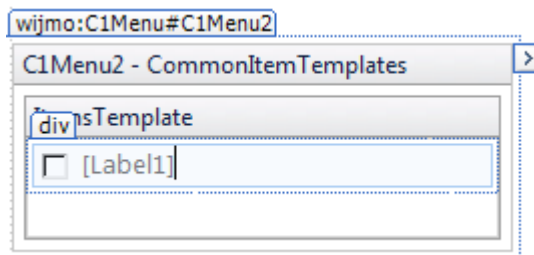
This tutorial will teach you how to create an [ItemsTemplate](#) for the [C1Menu](#) control. The template, which will be the prototype for *all* items in the **C1Menu** control, will consist of a **Div** element, an **Input (check box)** control, and a **Label** control. The label control's **Text** property will be bound to the **Text** property of the individual menu items.


Complete the following steps:

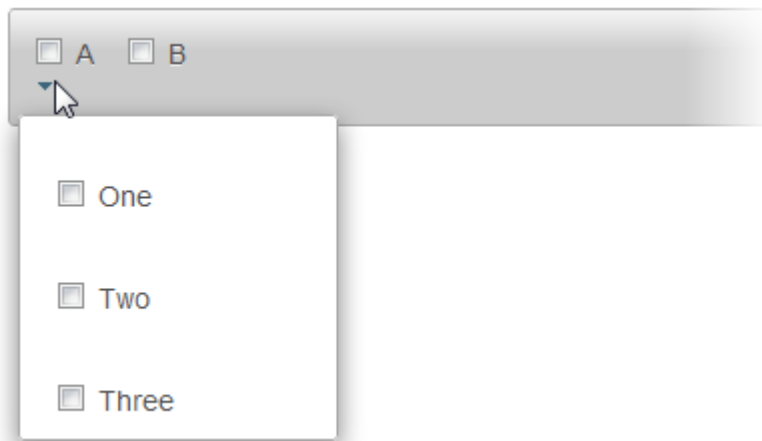
- Create a **C1Menu** control that has one two top-level menu items named "A" and "B". Give item "A" three child items and name them "One", "Two", and "Three".



- Click the [C1Menu](#) control's smart tag  and select **Edit Templates** from the **C1Menu Tasks** menu. The **C1Menu Tasks** menu changes to **Template Editing Mode**. The default mode listed in the **Description** drop-down list is [ItemsTemplate](#), so there's no need to change it.
- Navigate to the Toolbox and, using a drag-and-drop operation, add a **Div** element to the **ChildItems** template.
- Navigate to the Toolbox and, using a drag-and-drop operation, add an **Input (Checkbox)** control and a **Label** control to the **Div** element. The template will look like this:




5. Select the **Label** element, navigate to the **Properties** window, and set the **Text** property to "<% #DataBinder.Eval(Container.DataItem,"Text") %>". This will bind the label's **Text** property to a data binding, which we will specify in a later step.
6. Click **End Template Editing** to close the template.
7. Create a data binding by completing the following steps:
 - a. Click the **C1Menu** control's smart tag  and select **Edit Menu Databindings** from the **C1Menu Tasks** menu.
 - b. Complete the following steps in the **Bindings Collection Editor** dialog box:
 - Click **Add** to add an <Empty> databinding to the project.
 - Set the <Empty> databinding's **TextField** property to **Text**.
 - Click **OK** to close the **Bindings Collection Editor** dialog box.
8. Run the project and select **A**. Observe that each menu item in the **C1Menu** control top-level and child items - contains a check box and a label. Also observe that the label text is the same as the text you specified in step 1 of this tutorial because of the data bindings you specified in this tutorial.

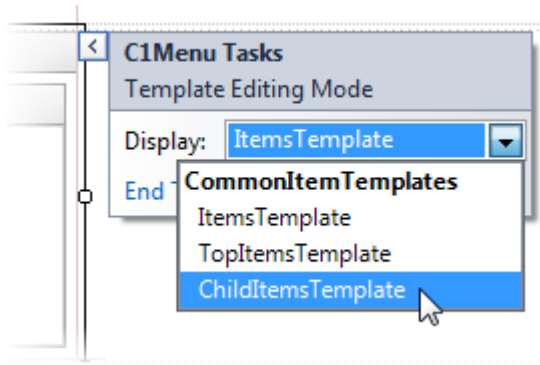


Creating a Child Items Template

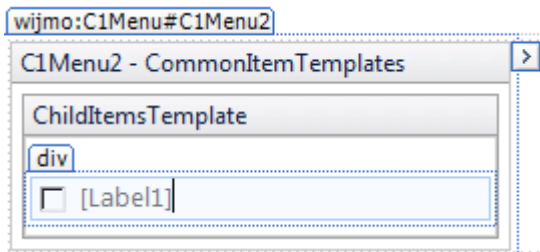
This tutorial will teach you how to create a [ChildItemsTemplate](#) for the **C1Menu** control. The template, which will be the prototype for all items in the C1Menu control's various submenus, will consist of a **Div** element, an **Input (check box)** control, and a **Label** control. The label control's **Text** property will be bound to the **Text** property of the individual menu items.


Complete the following steps:

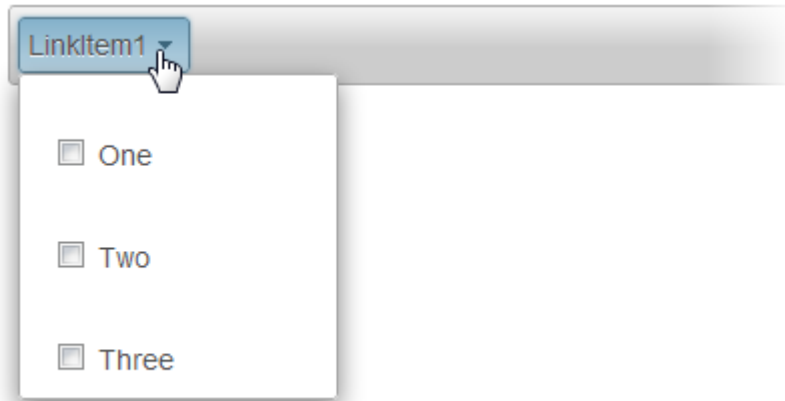
1. Create a **C1Menu** control that has one top-level menu item with three child items. Name the child items "One", "Two", and "Three".
2. Click the **C1Menu** control's smart tag  and select **Edit Templates** from the **C1Menu Tasks** menu. The **C1Menu Tasks** menu changes to **Template Editing Mode**.
3. Select **ChildTopItemsTemplate** from the **Display** drop-down list box.



4. Navigate to the Toolbox and, using a drag-and-drop operation, add a **Div** element to the **ChildItems** template.
5. Navigate to the Toolbox and, using a drag-and-drop operation, add an **Input (Checkbox)** control and a **Label** control to the **Div** element. The template will look like this:



6. Select the Label element, navigate to the Properties window, and set the Text property to "<% #DataBinder.Eval(Container.DataItem, 'Text') %>". This will bind the label's Text property to a data binding, which we will specify in a later step.
7. Click End Template Editing to close the template.
8. Create a data binding by completing the following steps:
 - a. Click the **C1Menu** control's smart tag  and select **Edit Menu Databindings** from the **C1Menu Tasks** menu.
 - b. Complete the following steps in the **Bindings Collection Editor** dialog box:
 - Click Add to add an <Empty> databinding to the project.
 - Set the <Empty> databinding's **TextField** property to **Text**.
 - Click **OK** to close the **Bindings Collection Editor** dialog box.
9. Run the project and select LinkItem1. Observe that each menu item in the sub menu contains a check box and a label. Also observe that the label text is the same as the text you specified in step 1 of this tutorial; that is because of the data bindings.

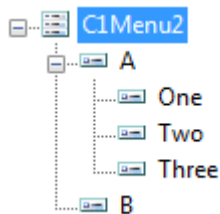



Creating a Top-Level Item Template

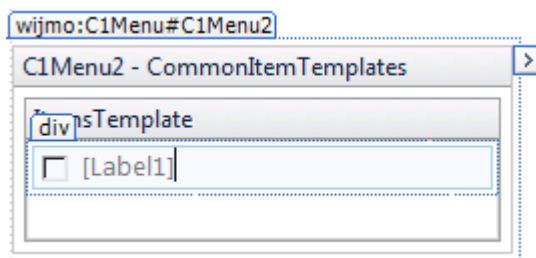
This tutorial will teach you how to create an [TopItemsTemplate](#) for the [C1Menu](#) control. The template, which will be the prototype for *all* items in the **C1Menu** control, will consist of a **Div** element, an **Input (check box)** control, and a **Label** control. The label control's **Text** property will be bound to the **Text** property of the individual menu items.


Complete the following steps:

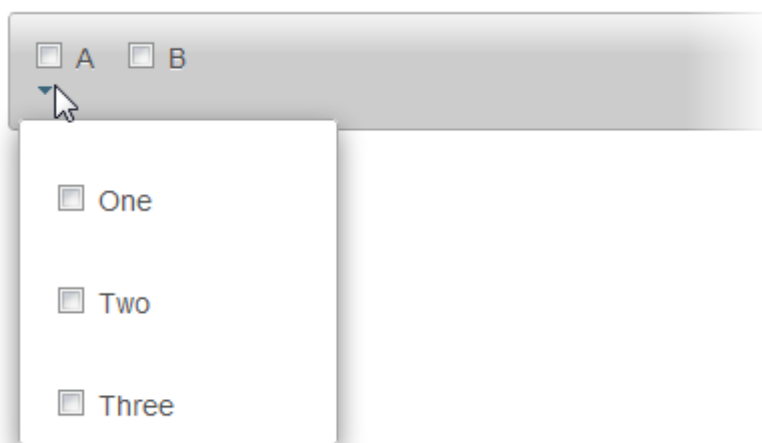
1. Create a **C1Menu** control that has one two top-level menu items named "A" and "B". Give item "A" three child items and name them "One", "Two", and "Three".



2. Click the [C1Menu](#) control's smart tag  and select **Edit Templates** from the **C1Menu Tasks** menu. The **C1Menu Tasks** menu changes to **Template Editing Mode**. The default mode listed in the Description drop-down list is [TopItemsTemplate](#), so there's no need to change it.
3. Navigate to the Toolbox and, using a drag-and-drop operation, add a **Div** element to the **ChildItems** template.
4. Navigate to the Toolbox and, using a drag-and-drop operation, add an **Input (Checkbox)** control and a **Label** control to the **Div** element. The template will look like this:



5. Select the **Label** element, navigate to the **Properties** window, and set the **Text** property to "`<% #DataBinder.Eval(Container.DataItem, 'Text') %>`". This will bind the label's **Text** property to a data binding, which we will specify in a later step.
6. Click **End Template Editing** to close the template.
7. Create a data binding by completing the following steps:
 - a. Click the **C1Menu** control's smart tag  and select **Edit Menu Databindings** from the **C1Menu Tasks** menu.
 - b. Complete the following steps in the **Bindings Collection Editor** dialog box:
 - Click **Add** to add an `<Empty>` databinding to the project.
 - Set the `<Empty>` databinding's **TextField** property to **Text**.
 - Click **OK** to close the **Bindings Collection Editor** dialog box.
8. Run the project and select **A**. Observe that each menu item in the C1Menu control top-level and child items - contains a check box and a label. Also observe that the label text is the same as the text you specified in step 1 of this tutorial because of the data bindings you specified in this tutorial.



Working with CSS Selectors

C1Menu allows full customization through supporting CSS. This section will teach you how to use the built-in CSS selectors to customize the C1Menu control. You can use single selectors or you can combine selectors to make the CSS more specific.

See Also

[Customizing Appearance with CSS Selectors](#)

[Customizing Link Appearance with CSS Selectors](#)

Customizing Appearance with CSS Selectors

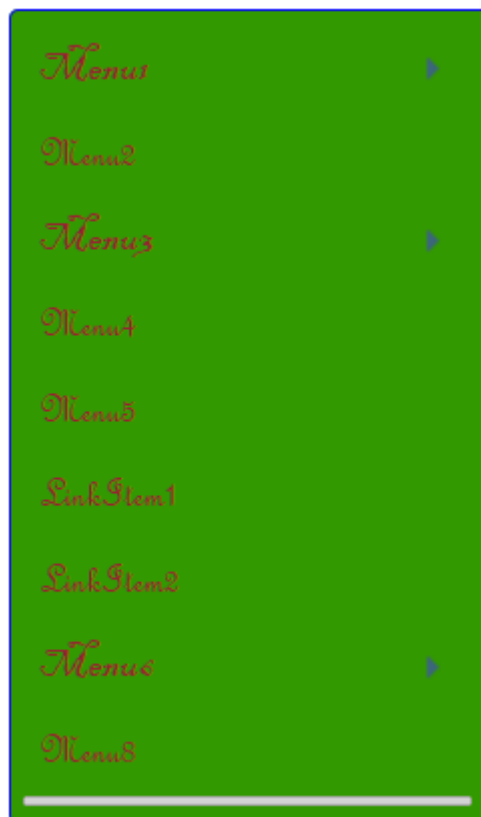
You can use CSS selectors to customize the appearance of the **C1Menu** control in the Source View.

1. In the Source View, locate the first set of `<asp:Content>` tags. Add the following tags between the `<asp:Content>` `</asp:Content>` tags that will allow you to insert CSS styling.

To write code in Source View

```
<style type="text/css"> </style>
```

2. Use the following CSS selector to set the general appearance of the control.
- 3.
4.
 - .wijmo-wijmenu
 - o Insert `{background: #339900; border-color: Blue; }` after the CSS selector. This will set the background to green with a blue border.
 - o Insert `.wijmo-wijmenu-text {color: #992233; }` to set the color of the text for the control.
 - o Insert `.wijmo-wijmenu-item {font-family: French Script MT; }` and `.wijmo-wijmenu-parent {font-family: Blackadder ITC; }` to set the fonts for menu items and parent menu items.
 - o Press F5 to run your program. Note the changes that you have made to the appearance of the **C1Menu** control. It should appear as in the following image. Note that parent menu items and regular menu items appear in different font styles.



Customizing Link Appearance with CSS Selectors

C1Menu allows you to customize link appearance using CSS selectors. For this topic, you will learn how to use the CSS selectors to customize links on mouse hover.

1. In the Source View, locate the first set of `<asp:Content>` tags. Add the following tags between the `<asp:Content>` `</asp:Content>` tags that will allow you to insert CSS styling.

To write code in Source View

```
<style type="text/css"> </style>
```

2. Use the following CSS selector to set the general appearance of the control.

To write code in Source View

```
.wijmo-wijmenu a.wijmo-wijmenu-link:hover  
{  
font-family:Calibri; background: Pink; border-color: green;  
border-style: solid;
```

3. Press F5 to run your program. Hover over one of the menu items; it should appear as in the following image.



Displaying a C1Menu Control as a Context Menu

The C1Menu control can be used as a context menu for most of the other ComponentOne controls. This topic will walk you through displaying a C1Menu control as a context menu for a C1TreeView control.

Complete the following steps:

1. Add a **C1Menu** control and a **C1TreeView** control to your application.
2. Click the **C1Menu** smart tag to open the **C1Menu Tasks** menu and select **Edit Menu**.
3. From the **C1Menu Designer Form**, click the **Add Child Item** button to add a **C1MenuItem** to the **C1Menu** control.
4. Select the **C1MenuItem** you just added and locate the **Text** property in the Properties window. Enter "Add Nodes" in the **Text** property textbox.
5. Click the **Add Child Item** button to add a **C1MenuItem** to the **C1Menu** control.
6. Select the **C1MenuItem** you just added and locate the **Text** property in the Properties window. Enter "Delete Nodes" in the **Text** property textbox.
7. Click **OK** to close the **C1Menu Designer Form** and switch to Source View.
8. Add the following markup to the `<cc1:C1Menu>` tags:

OnClientSelect

```
onclientselect="C1Menu_OnClientSelect" Orientation="Vertical"
    Trigger="#C1TreeView1" TriggerEvent="Rtclick"
```

The complete opening tag markup should resemble the following:

To write code in Source View

```
<cc1:C1Menu ID="C1Menu1" runat="server"

    onclientselect="C1Menu_OnClientSelect"
    Orientation="Vertical"

    Trigger="#C1TreeView1" TriggerEvent="Rtclick">
```

9. Locate the `<cc1:C1TreeView>` tags and add the following markup between the tags:

To write code in Source View

```
<Nodes>
    <cc1:C1TreeViewNode runat="server"
    CheckState="UnChecked" NodeIndex="0"
        Text="Node1">
        <Nodes>
            <cc1:C1TreeViewNode runat="server"
            CheckState="UnChecked" NodeIndex="0"
                Text="SubNode1">
            </cc1:C1TreeViewNode>
```



```
        <ccl:C1TreeNode runat="server"
CheckState="Unchecked" NodeIndex="0"
        Text="SubNode2">
        </ccl:C1TreeNode>
    </Nodes>
</ccl:C1TreeNode>
<ccl:C1TreeNode runat="server"
CheckState="Unchecked" NodeIndex="0"
    Text="Node2">
    <Nodes>
        <ccl:C1TreeNode runat="server"
CheckState="Unchecked" NodeIndex="0"
        Text="SubNode1">
        </ccl:C1TreeNode>
        <ccl:C1TreeNode runat="server"
CheckState="Unchecked" NodeIndex="0"
        Text="SubNode2">
        </ccl:C1TreeNode>
    </Nodes>
    </ccl:C1TreeNode>
</Nodes>
```

10. Add the following script above the `<ccl:C1Menu>` markup.

To write code in Source View

```
<script type="text/javascript">
    $(document).ready(function () {
        $(".wijmo-wijtree-node").mousedown(function (e) {
            if (e.button == 2)
                $(this).arent().c1treeviewnode("option",
"selected", "true");
        });
    });

    function C1Menu_OnClientSelect(sender, args) {
        alert("You have selected " +
args.item[0].innerText);
    }
</script>
```


11. Run your application. When you right-click one of the **C1TreeView** nodes, the menu will appear.

Adding a Top-Level Item to a Menu

This topic illustrates how to add a top-level menu item to a [C1Menu](#) control in Design view, in Source view, and in code.

In Design View

Complete the following steps:

1. Click the smart tag to open the **C1Menu Tasks** menu. Select **Edit Menu**. The **C1Menu Designer Form** dialog box opens.
2. Click the **Add Child Item** button  to add a **C1MenuItem** to the **C1Menu** control.
3. Click **OK** to close the **C1Menu Design Form** dialog box.

In Source View

Add the following markup between the `<cc1:C1Menu>` tags:

To write code in Source View

```
<cc1:C1MenuItem ID="MenuItem1" runat="server" Text="LinkItem1">
</cc1:C1MenuItem>
```

In Code View

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.C1Menu</pre>	

To write code in C#

C#	Copy Code
<pre>using C1.Web.Wijmo.Controls.C1Menu;</pre>	

2. Add the following code to the `Page_Load` event:

To write code in Visual Basic

Visual Basic	Copy Code
<pre>Dim MenuItem1 As New C1MenuItem() C1MenuItem1.Text = "LinkItem1" C1Menu1.Items.Add(C1MenuItem1)</pre>	

To write code in C#

C#	Copy Code
<pre>C1MenuItem MenuItem1 = new C1MenuItem(); C1MenuItem1.Text = "LinkItem1"; C1Menu1.Items.Add(C1MenuItem1);</pre>	

3. Run the program.

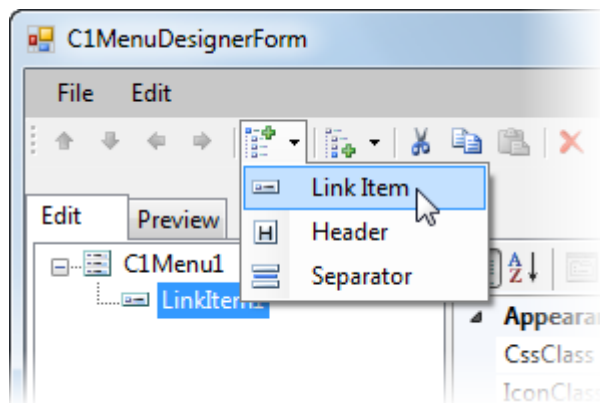
Creating a Drop-Down Menu

This topic illustrates the creation of a drop-down menu using Design view, Source view, and code. To add a drop-down menu, all you have to do is add one (or more) [C1MenuItem](#) as a child of a top-level **C1MenuItem**. This topic assumes that you have completed [Adding a Top-Level Item to a Menu](#).

In Design View

Complete the following steps:

1. Click the smart tag to open the **C1Menu Tasks** menu. Select **Edit Menu**.
2. The **C1Menu Designer Form** dialog box opens.
3. Select the menu item you wish to add the submenu to.
4. Click the **Add Child Item** drop-down arrow and select **Link Item** from the list to add a child item to the selected menu item.



5. Click **OK** to close the **C1Menu Design Form** dialog box.

In Source View

Add the following markup between the `<cc1:C1MenuItem>` tags of the item you wish to add the submenu to:

[To write code in Source View](#)

```
<Nodes>
  <cc1:C1MenuItem ID="Node1" runat="server" Text="LinkItem1">
  </cc1:C1MenuItem>
</Nodes>
```

In Code View

Complete the following steps:

1. Import the following namespace into your project:

[To write code in Visual Basic](#)

Visual Basic	Copy Code
<pre>Imports Cl.Web.Wijmo.Controls.C1Menu</pre>	

[To write code in C#](#)

C#	Copy Code
<pre>using Cl.Web.Wijmo.Controls.C1Menu;</pre>	

2. Add the following code to the **Page_Load** event:

[To write code in Visual Basic](#)

Visual Basic	Copy Code
<pre>' Create first node and add it to the C1Menu. Dim C1MenuItem1 As New C1MenuItem() C1MenuItem1.Text = "LinkItem1" C1Menu1.Items.Add(C1MenuItem1) ' Create the child node and add it to C1MenuItem1 Dim C1MenuItem2 As New C1MenuItem() C1MenuItem1.Text = "LinkItem1" C1MenuItem1.Items.Add(C1MenuItem2)</pre>	

[To write code in C#](#)

C#	Copy Code

```
// Create first node and add it to the C1Menu.
C1MenuItem C1MenuItem1 = new C1MenuItem();
C1MenuItem1.Text = "LinkItem1";
C1Menu1.Items.Add(C1MenuItem1);
// Create the child node and add it to C1MenuItem1
C1MenuItem C1MenuItem2 = new C1MenuItem();
C1MenuItem2.Text = "LinkItem1";
C1MenuItem1.Items.Add(C1MenuItem2);
```

3. Run the program.

✓ **This topic illustrates the following:**

Creating a submenu is as easy as adding a child **C1MenuItem** to a parent **C1MenuItem**.



Creating a Sliding Menu

Menu for ASP.NET Web Forms can be displayed as a traditional flyout menu or as a touch phone-like sliding menu. In this tutorial, you'll create a sliding menu with a breadcrumb header and an elastic sliding animation.

Complete the following steps:

1. In Source view, add the following markup between the `<cc1:C1Menu>` and `</cc1:C1Menu>` tags to add menu items and submenus to the **C1Menu** control:

To write code in Source View

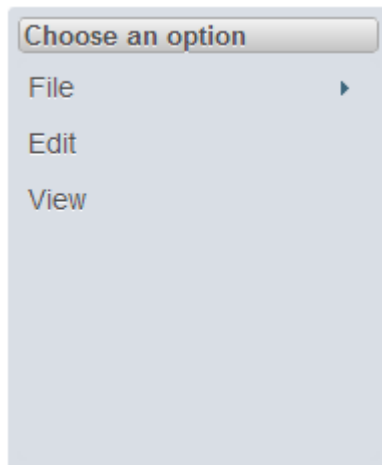
```
<Items>
  <cc1:C1MenuItem ID="C1MenuItem1" runat="server"
    ImagePosition="Left" Text="File">
    <Items>
      <cc1:C1MenuItem runat="server" Header="True" Text="File
        Menu" StaticKey="sk1"
          ImagePosition="Left"
          ID="C1MenuItem2"></cc1:C1MenuItem>
      <cc1:C1MenuItem runat="server"
        ImagePosition="Left"
        Text="Open File">
      </cc1:C1MenuItem>
    <cc1:C1MenuItem ID="C1MenuItem3"
      runat="server"
```

```
        ImagePosition="Left" Text="Save">
            </cc1:C1MenuItem>
            <cc1:C1MenuItem ID="C1MenuItem4"
runat="server"
        ImagePosition="Left" Text="Save As...">
            </cc1:C1MenuItem>
            <cc1:C1MenuItem runat="server"
ImagePosition="Left"
        Separator="True">
            </cc1:C1MenuItem>
            <cc1:C1MenuItem ID="C1MenuItem6"
runat="server"
        ImagePosition="Left" Text="Close Project">
            </cc1:C1MenuItem>
            <cc1:C1MenuItem runat="server"
ImagePosition="Left"
        Separator="True">
            </cc1:C1MenuItem>
            <cc1:C1MenuItem runat="server"
ImagePosition="Left"
        Text="Options">
            <Items>
                <cc1:C1MenuItem
runat="server" ImagePosition="Left" Text="Margins">
            </cc1:C1MenuItem>
                <cc1:C1MenuItem
runat="server" ImagePosition="Left" Text="Settings">
            </cc1:C1MenuItem>
            </Items>
            </cc1:C1MenuItem>
        </Items>
        </cc1:C1MenuItem>
        <cc1:C1MenuItem ID="C1MenuItem7" runat="server"
ImagePosition="Left" Text="Edit">
            </cc1:C1MenuItem>
            <cc1:C1MenuItem ID="C1MenuItem8" runat="server"
ImagePosition="Left"
        Text="View">
            </cc1:C1MenuItem>
        </Items>
```

2. Switch to Design view.
3. Click the **C1Menu** control's smart tag to open the **C1Menu Tasks** menu. Click **Edit Menu**.

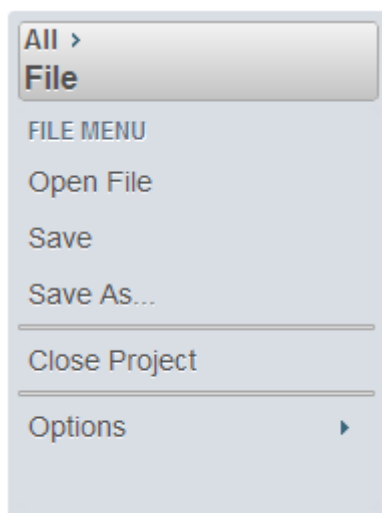
The **C1MenuDesignerForm** dialog box opens.

4. With **C1Menu1** selected in the treeview, set the following properties:
 - Set the **Mode** property to **Sliding**. This is what changes **C1Menu** from a traditional flyout menu to a sliding phone-like sliding menu.
 - Set the **BackLink** property to **False**. This replaces the default "Back" link with breadcrumb links.
 - Expand the **SlidingAnimation** node and set the **Easing** property to **EaseInOutElastic**. This sets the animation for the sliding menu.
5. Click **OK** to exit the **C1MenuDesignerForm** dialog box.
6. Press **F5** to run your project. The project will look as follows:

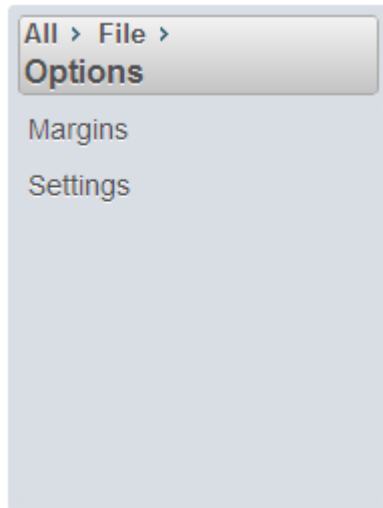


The "Choose an option" text at the top of the menu is a placeholder. It's where the breadcrumb links will appear as you navigate through the menu.

7. Click **File**. Did you notice the elastic animation as the new menu screen slid into view? Also observe that the breadcrumb bar says "All > File"



8. Click Options and observe that the breadcrumbs say "All > File > Options". To return to the File menu, click **File**. To return to the top-level menu, click All.

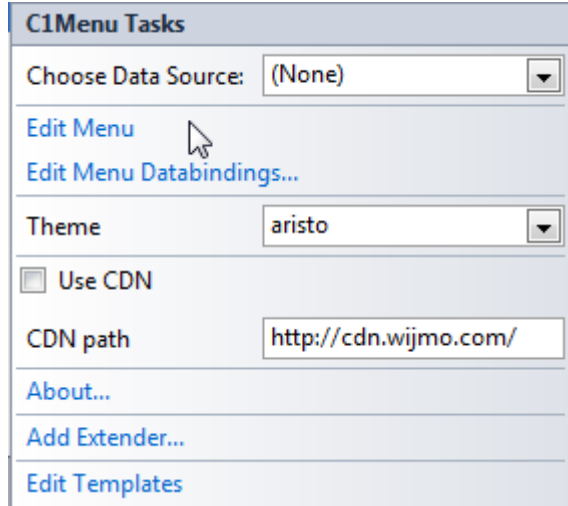


Animating C1Menu

The **C1Menu** control supports animation effects. This topic illustrates changing the animation effects using either the **C1Menu Tasks** menu or the Source View.

In Design View

1. Click the **C1Menu** smart tag to open the **C1Menu Tasks** menu.
2. Click the **Edit Menu** link to open the **C1Menu DesignerForm**.



3. Find the **Hide Animation** and **Show Animation** properties in the list. Use the arrows to expand the property nodes.

HideAnimation	C1.Web.Wijmo.Controls.An
▶ Animated	Effect: blind
Duration	400
Easing	Swing
Option	(Collection)
HideDelay	400
ShowAnimation	C1.Web.Wijmo.Controls.An
▶ Animated	Effect: bounce
Duration	400
Easing	Swing
Option	(Collection)
ShowDelay	400
SkinID	

4. Set the **HideAnimation > Animation > Effect** property to **blind** and the **ShowAnimation > Animation > Effect** property to **bounce**.
5. Click **OK** and then press F5 to run your program. Note the animation effects as you open and close submenus.

In Source View

1. In the Source View of your project, add the following markup between the `<cc1:C1Menu>` tags.

To write code in Source View

```
<ShowAnimation>
  <Animated Effect="bounce" />
</ShowAnimation>
<HideAnimation>
  <Animated Effect="blind"></Animated>
</HideAnimation>
```

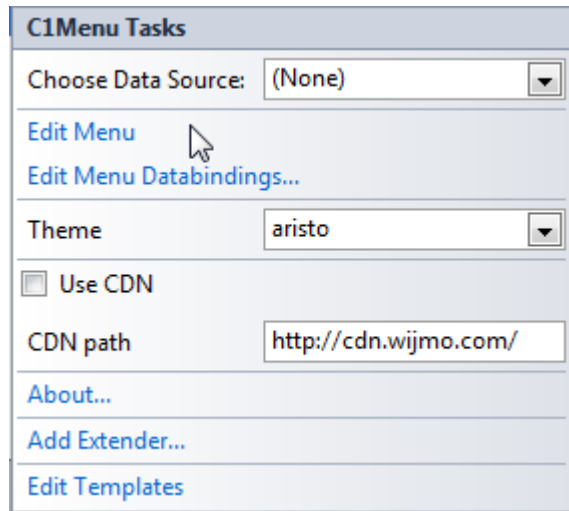
2. Press F5 to run your program. Hover your mouse over the submenus to see the animation effects.

Changing Menu Item Triggers

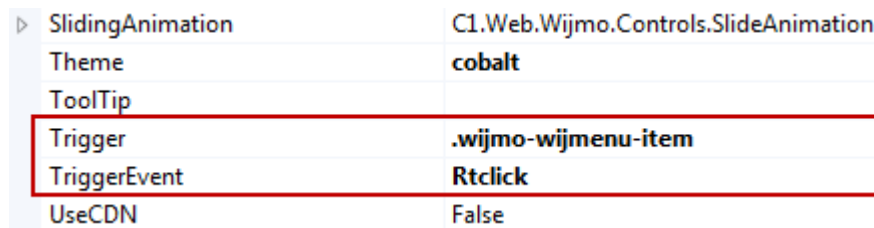
The **C1Menu** control can respond to different trigger events to open a menu item. This topic illustrates how to set the **Trigger** and **TriggerEvent** properties to change the trigger that opens the menu items in either the Design View or the Source View.

In Design View

1. Click the **C1Menu** smart tag to open the **C1Menu Tasks** menu.
2. Click the **Edit Menu** link to open the **C1Menu DesignerForm**.



3. Locate the **Trigger** and **TriggerEvent** properties in the list.
4. Set the **Trigger** property to **.wijmo-wijmenu-item** and then use the **TriggerEvent** drop-down list to choose a **Trigger Event**. For this task, set the **Trigger Event** to **Rtclick** as in the following image.



5. Click **OK** and then press F5 to run your application. The submenus will appear when you right-click on the menu item.

In Source View


In Source View, add `TriggerEvent="Rtclick"` and `Trigger=".wijmo-wijmenu-item"` to the `<cc1:C1Menu>` tags.








Press F5 to run your project. Note that you now need to right-click a menu item to make the submenu drop down.

Adding Icons to C1Menu Items

The `iconClass` property allows you to show icons in the C1Menu control. This topic illustrates adding icons to the control.

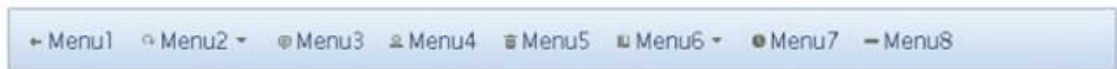
The following icons are shown in this help:

Icon Name	Icon	Icon markup
arrowthick		ui-icon ui-icon-arrowthick-1-w

arrowrefresh		ui-icon ui-icon-arrowrefresh-1-s
comment		ui-icon ui-icon-comment
person		ui-icon ui-icon-person
trash		ui-icon ui-icon-trash
bookmark		ui-icon ui-icon-bookmark"
clock		ui-icon ui-icon-clock
minusthick		ui-icon ui-icon-minusthick

In Source View

1. In Source View, add the following markup to the first `<cc1:C1MenuItem>` tag:
- 2.
3. `IconClass="ui-icon ui--icon-arrowthick-1-w"`
- 4.
5. Continue adding the markup for the icons to the rest of the `c1menuitem`.
6. When you run your application, it will resemble the following image:



Setting the Position of the Context Menu

You can customize where the context menu displays relative to it's trigger, such as a button that causes the menu to display.

1. Add a `C1Menu` control and a general button control to your project.
2. Switch to Source View and edit the markup so that it resembles the following:

To write code in Source View

```
<input type="button" value="click here" id="btn1" />
<cc1:C1Menu runat="server" ID="Menu1" Orientation="Vertical"
Trigger="#btn1" TriggerEvent="Click">
<Animation Option=""></Animation>
<ShowAnimation Option=""></ShowAnimation>
<HideAnimation Option="">
    <Animated Effect="fade"></Animated>
</HideAnimation>
<Items>
    <cc1:C1MenuItem ID="C1MenuItem1" runat="server" Text="Breaking
```

```

News">
    </cc1:C1MenuItem>
    <cc1:C1MenuItem ID="C1MenuItem2" runat="server"
Text="Entertainment">
    </cc1:C1MenuItem>
    <cc1:C1MenuItem ID="C1MenuItem3" runat="server"
Text="Finance">
    </cc1:C1MenuItem>
    <cc1:C1MenuItem ID="C1MenuItem4" runat="server" Text="Food
&#38; Cooking">
    </cc1:C1MenuItem>
    <cc1:C1MenuItem ID="C1MenuItem5" runat="server"
Text="Lifestyle">
    <Items>
        <cc1:C1MenuItem ID="C1MenuItem8" runat="server"
Text="submenu"></cc1:C1MenuItem>
    </Items>
    </cc1:C1MenuItem>
    <cc1:C1MenuItem ID="C1MenuItem6" runat="server" Text="News">
    </cc1:C1MenuItem>
    <cc1:C1MenuItem ID="C1MenuItem7" runat="server"
Text="Politics">
    </cc1:C1MenuItem>
    <cc1:C1MenuItem ID="C1MenuItem9" runat="server" Text="Sports">
    </cc1:C1MenuItem>
</Items>
</cc1:C1Menu>

```

Note that you have set the Click Event and the Trigger properties for the C1Menu control.

- Switch back to Design View and click the C1Menu control to show the control properties in the Properties Window. Scroll down to find the **Position** property and set the options to resemble the following image:

Position	
▲ At	Center, Top
Left	Center
Top	Top
▲ Collision	Flip, Flip
Left	Flip
Top	Flip
▲ My	Right, Top
Left	Right
Top	Top
▲ Offset	0, 0
Left	0
Top	0

- When you run your application, you'll see a button control. When you click the button, the context menu will appear at the position you specified in the properties.

Item Functions

The **C1Menu** control has an extensive client-side API. This topic illustrates how to call the client-side methods using the same patterns you would see in a jQuery UI using the Source View.

1. In Source View, add the following markup between the `<c1:C1Menu>` tags.

To write code in Source View

```

<Items>

    <c1:C1MenuItem Text="MenuItem"
runat="server">

        </c1:C1MenuItem>

        <c1:C1MenuItem ID="C1MenuItem2"
Text="Breaking News" runat="server">

            <Items>

                <c1:C1MenuItem runat="server"
Header="true" Text="header2">

                    </c1:C1MenuItem>

                    <c1:C1MenuItem runat="server"
Separator="true">

                        </c1:C1MenuItem>

                        <c1:C1MenuItem runat="server"
Text="Entertainment">

                            </c1:C1MenuItem>

                            <c1:C1MenuItem
ID="C1MenuItem6" runat="server" Text="Politics">

                                </c1:C1MenuItem>

                                <c1:C1MenuItem
ID="C1MenuItem7" runat="server" Text="A&E">

                                    </c1:C1MenuItem>

                                    <c1:C1MenuItem
ID="C1MenuItem8" runat="server" Text="Sports">

                                        </c1:C1MenuItem>

                                        <c1:C1MenuItem
ID="C1MenuItem9" runat="server" Text="Local">

                                            </c1:C1MenuItem>

```

```

                                <cc1:C1MenuItem
ID="C1MenuItem10" runat="server" Text="Health">

                                </cc1:C1MenuItem>

                                </Items>

                                </cc1:C1MenuItem>

                                <cc1:C1MenuItem ID="C1MenuItem11"
runat="server" Text="Entertainment">

                                <Items>

                                <cc1:C1MenuItem
ID="C1MenuItem12" runat="server" Text="Celebrity news">

                                </cc1:C1MenuItem>

                                <cc1:C1MenuItem
ID="C1MenuItem13" runat="server" Text="Gossip">

                                </cc1:C1MenuItem>

                                <cc1:C1MenuItem
ID="C1MenuItem14" runat="server" Text="Movies">

                                </cc1:C1MenuItem>

                                <cc1:C1MenuItem
ID="C1MenuItem15" runat="server" Text="Music">

                                <Items>

                                <cc1:C1MenuItem
ID="C1MenuItem16" runat="server" Text="Alternative">

                                </cc1:C1MenuItem>

                                <cc1:C1MenuItem
ID="C1MenuItem17" runat="server" Text="Country">

                                </cc1:C1MenuItem>

                                <cc1:C1MenuItem
ID="C1MenuItem18" runat="server" Text="Dance">

                                </cc1:C1MenuItem>

                                <cc1:C1MenuItem
ID="C1MenuItem19" runat="server" Text="Electronica">

                                </cc1:C1MenuItem>

```

```

                                <ccl:C1MenuItem
ID="C1MenuItem20" runat="server" Text="Metal">

</ccl:C1MenuItem>

                                <ccl:C1MenuItem
ID="C1MenuItem21" runat="server" Text="Pop">

</ccl:C1MenuItem>

                                <ccl:C1MenuItem
ID="C1MenuItem22" runat="server" Text="Rock">

                                <Items>

<ccl:C1MenuItem ID="C1MenuItem23" runat="server" Text="Bands">

                                <Items>

                                <ccl:C1MenuItem ID="C1MenuItem24" runat="server"
Text="Dokken">

                                </ccl:C1MenuItem>

                                </Items>

</ccl:C1MenuItem>

<ccl:C1MenuItem ID="C1MenuItem25" runat="server" Text="Fan
Clubs">

</ccl:C1MenuItem>

<ccl:C1MenuItem ID="C1MenuItem26" runat="server" Text="Songs">

</ccl:C1MenuItem>

                                </Items>

</ccl:C1MenuItem>

                                </Items>

                                </ccl:C1MenuItem>
```

```

                                <ccl:C1MenuItem
ID="C1MenuItem27" runat="server" Text="Slide shows">

                                </ccl:C1MenuItem>

                                <ccl:C1MenuItem
ID="C1MenuItem28" runat="server" Text="Red carpet">

                                </ccl:C1MenuItem>

                                </Items>

                                </ccl:C1MenuItem>

                                <ccl:C1MenuItem ID="C1MenuItem29"
Text="Finance" runat="server">

                                <Items>

                                <ccl:C1MenuItem
ID="C1MenuItem30" Text="Personal" runat="server">

                                <Items>

                                <ccl:C1MenuItem
ID="C1MenuItem31" Text="Loans" runat="server">

                                </ccl:C1MenuItem>

                                <ccl:C1MenuItem
ID="C1MenuItem32" Text="Savings" runat="server">

                                </ccl:C1MenuItem>

                                <ccl:C1MenuItem
ID="C1MenuItem33" Text="Mortgage" runat="server">

                                </ccl:C1MenuItem>

                                <ccl:C1MenuItem
ID="C1MenuItem34" Text="Debt" runat="server">

                                </ccl:C1MenuItem>

                                </Items>

                                </ccl:C1MenuItem>

                                <ccl:C1MenuItem
ID="C1MenuItem35" Text="Business" runat="server">

                                </ccl:C1MenuItem>

                                </Items>
```



```
</cc1:C1MenuItem>

    <cc1:C1MenuItem ID="C1MenuItem36"
Text="Food &#38; Cooking" runat="server">

        <Items>

            <cc1:C1MenuItem
ID="C1MenuItem37" Text="Breakfast" runat="server">

                </cc1:C1MenuItem>

            <cc1:C1MenuItem
ID="C1MenuItem38" Text="Lunch" runat="server">

                </cc1:C1MenuItem>

            <cc1:C1MenuItem
ID="C1MenuItem39" Text="Dinner" runat="server">

                </cc1:C1MenuItem>

            <cc1:C1MenuItem
ID="C1MenuItem40" Text="Dessert" runat="server">

                <Items>

                    <cc1:C1MenuItem
ID="C1MenuItem41" Text="Dump Cake" runat="server">

                        </cc1:C1MenuItem>

                    <cc1:C1MenuItem
ID="C1MenuItem42" Text="Doritos" runat="server">

                        </cc1:C1MenuItem>

                    <cc1:C1MenuItem
ID="C1MenuItem43" Text="Both please" runat="server">

                        </cc1:C1MenuItem>

                </Items>

            </cc1:C1MenuItem>

        </Items>

    </cc1:C1MenuItem>

    <cc1:C1MenuItem ID="C1MenuItem44"
Text="Lifestyle" runat="server">

        </cc1:C1MenuItem>
```

```
<ccl:C1MenuItem ID="C1MenuItem45"
Text="News" runat="server">

</ccl:C1MenuItem>

<ccl:C1MenuItem ID="C1MenuItem46"
Text="Politics" runat="server">

</ccl:C1MenuItem>

<ccl:C1MenuItem ID="C1MenuItem47"
Text="Sports" runat="server">

</ccl:C1MenuItem>

<ccl:C1MenuItem ID="C1MenuItem48"
Text="Novels" runat="server">

</ccl:C1MenuItem>

<ccl:C1MenuItem ID="C1MenuItem49"
Text="Magazine" runat="server">

</ccl:C1MenuItem>

<ccl:C1MenuItem ID="C1MenuItem50"
Text="Books" runat="server">

</ccl:C1MenuItem>

<ccl:C1MenuItem ID="C1MenuItem51"
Text="Education" runat="server">

</ccl:C1MenuItem>

</Items>
```

2. After the closing `<ccl:C1Menu>` tag, insert the following markup to create the button controls.

[To write code in Source View](#)

```
<p>
<input type="button" id="previous" value="previous" />

<input type="button" id="next" value="next" />

<input type="button" id="previousPage" value="previousPage" />

<input type="button" id="nextPage" value="nextPage" />

</p>
```

3. Use the following script to call the client-side functions.

[To write code in Source View](#)

```
<script type="text/javascript">

    var count = 0;

    $(document).ready(function () {

        $("#previous").click(function () {

            $("#<%= Menu1.ClientID
%>").focus().clmenu("previous");

            count++;

        });

        $("#next").click(function () {

            $("#<%= Menu1.ClientID
%>").focus().clmenu("next");

            count++;

        });

        $("#previousPage").click(function () {

            if (count === 0) {

                $("#<%= Menu1.ClientID %>").find(".wijmo-
wijmenu-link:first").click();

            }

            $("#<%= Menu1.ClientID
%>").clmenu("previousPage");

            count++;

        });

        $("#nextPage").click(function () {

            if (count === 0) {

                $("#<%= Menu1.ClientID %>").find(".wijmo-
wijmenu-link:first").click();

            }

            $("#<%= Menu1.ClientID
%>").clmenu("nextPage");

            count++;

        });

    });

});
```

4. Press F5 to run your program. The menu should resemble the following image.



Dynamically Adding Items to C1Menu

This topic illustrates how to use the client-side controls to dynamically add items to **C1Menu**.

1. Go to the Source View and insert the following markup between the `<cc1:C1Menu>` tags to populate the menu.

[To write code in Source View](#)

```
<Items>
    <cc1:clmenuitem id="C1MenuItem1" runat="server"
text="Menu item">
    </cc1:clmenuitem>
    <cc1:clmenuitem id="C1MenuItem2" runat="server"
separator="true">
    </cc1:clmenuitem>
    <cc1:clmenuitem id="C1MenuItem3" runat="server"
text="Vertical"
        value="DynamicOrientationItem">
    <Items>
        <cc1:C1MenuItem ID="C1MenuItem4"
runat="server" Text="Menu item">
        </cc1:C1MenuItem>
        <cc1:C1MenuItem ID="C1MenuItem5"
runat="server" Text="Menu item">
        </cc1:C1MenuItem>
        <cc1:C1MenuItem ID="C1MenuItem6"
runat="server" Text="Menu item">
        </cc1:C1MenuItem>
        <cc1:C1MenuItem ID="C1MenuItem7"
runat="server" Text="Menu item">
        </cc1:C1MenuItem>
        <cc1:C1MenuItem ID="C1MenuItem8"
runat="server" Text="Menu item">
        </cc1:C1MenuItem>
    </Items>
    </cc1:clmenuitem>
</Items>
```

```

        </cc1:clmenuitem>
        <cc1:clmenuitem id="ClMenuItem9" runat="server"
text="Menu item">
        </cc1:clmenuitem>
        <cc1:clmenuitem id="ClMenuItem10" runat="server"
text="Menu item">
        </cc1:clmenuitem>

    </Items>

```

2. After the closing </Items> tag, insert the following markup to create the buttons that will control the add and remove functions.

To write code in Source View

```

<fieldset>
    <legend>Remove function</legend>
    <label for="tbSelector">
        Selector</label>

    <input type="text" id="tbSelector" />
    <label for="tbIndex">
        Index</label>
    <input type="text" id="tbIndex" />
    <input type="button" value="Remove"
onclick="remove()" />
</fieldset>

<fieldset>
    <legend>Add function</legend>
    <label for="tbItem">
        item</label>
    <input type="text" id="tbItem" />
    <label for="tbAddSelector">
        Selector</label>
    <input type="text" id="tbAddSelector" />
    <label for="tbAddIndex">
        Index</label>
    <input type="text" id="tbAddIndex" />
    <input type="button" value="Add" onclick="add()" />
</fieldset>

```

3. Use the following script to initialize the buttons.

To write code in Source View

```

<script type="text/javascript">
    function remove() {
        var index, selector;
        if ($("#tbIndex").val() != "" &&
!isNaN($("#tbIndex").val())) {
            index = parseInt($("#tbIndex").val());
        }
        if ($("#tbSelector").val() != "") {
            selector = $("#tbSelector").val();

```

```

    }
    if (!selector && index != undefined) {
        selector = index;
        index = null
    }
    $("#<%= Menu1.ClientID %>").c1menu("remove",
selector, index);
}

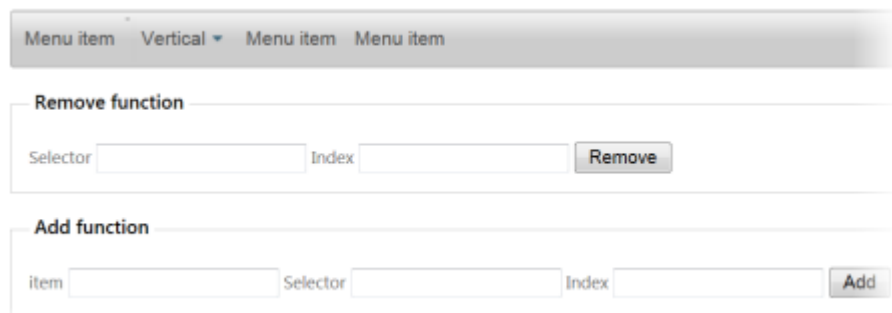
function add() {
    var index, selector, item;
    item = $("#tbItem").val();
    if ($("#tbAddIndex").val() != "" &&
!isNaN($("#tbAddIndex").val())) {
        index = parseInt($("#tbAddIndex").val());
    }

    if ($("#tbAddSelector").val() != "") {
        selector = $("#tbAddSelector").val();
    }

    if (!selecotr && index != undefined) {
        selector = index;
        index = null;
    }
    $("#<%= Menu1.ClientID %>").c1menu("add",
item, selector, index);
}
</script>

```

4. Press F5 to run your program. The **C1Menu** control should resemble the following image.



Populating C1Menu with a Site Map

This lesson illustrates how to populate a [C1Menu](#) with site map data.

To create the Site Map and bind it to the **C1Menu** control, complete the following:

1. In the Solution Explorer, right-click the project's name and select **Add New Item**.

The **Add New Item** dialog box appears.

2. Select **Site Map** from the list of templates, and then click **Add** to add the new **Web.sitemap** page to the project.

The following default source code appears for the **Web.sitemap** file:

[To write code in Source View](#)

```
<?xml version="1.0" encoding="utf-8" ?>
    <sitemap
xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0">

        <siteMapNode url="" title="" description="">
            <siteMapNode url="" title=""
description="" />
            <siteMapNode url="" title=""
description="" />
        </siteMapNode>
    </sitemap>
```

3. Replace the default data with the following data for the Web.sitemap file:

[To write code in Source View](#)

```
<?xml version="1.0" encoding="utf-8" ?>

<siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-
File-1.0" >

    <siteMapNode title="ComponentOne">

        <siteMapNode title="Products">

            <siteMapNode title="Studio Enterprise">

                <siteMapNode title="Studio for WinForms" />
                <siteMapNode title="Studio for ASP.NET" />
                <siteMapNode title="Studio for WPF" />
                <siteMapNode title="Studio for Mobile" />
                <siteMapNode title="Studio for ActiveX" />
                <siteMapNode title="Studio for Silverlight" />
            </siteMapNode>
        </siteMapNode>
    </siteMapNode>
```

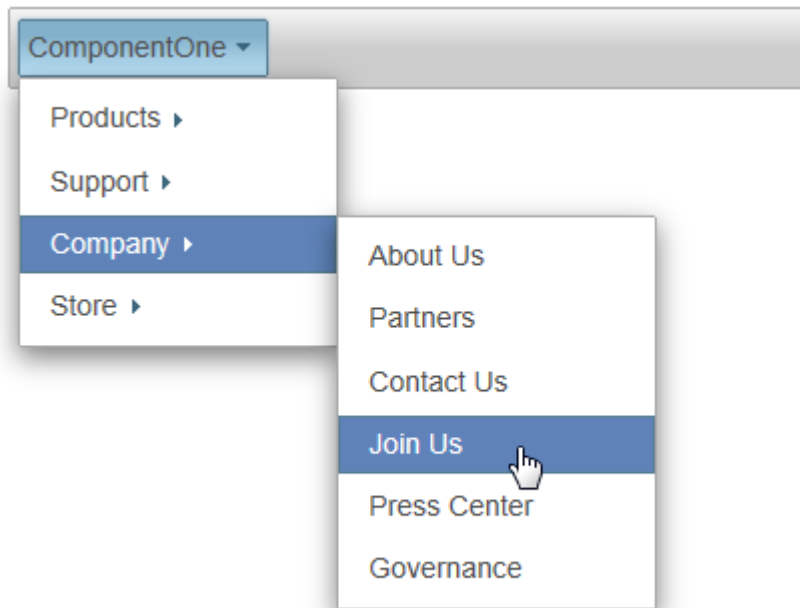
```
<siteMapNode title="Support">
    <siteMapNode title="Support Services" />
    <siteMapNode title="HelpCentral" />
    <siteMapNode title="Product Forums" />
</siteMapNode>
<siteMapNode title="Company">
    <siteMapNode title="About Us" />
    <siteMapNode title="Partners" />
    <siteMapNode title="Contact Us" />
    <siteMapNode title="Join Us" />
    <siteMapNode title="Press Center" />
    <siteMapNode title="Governance" />
</siteMapNode>
<siteMapNode title="Store">
    <siteMapNode title="Buy Now" />
    <siteMapNode title="Resellers" />
</siteMapNode>
</siteMapNode>
</siteMap>
```

4. Open C1Menu control's Tasks menu and click the Choose Data Source drop-down arrow. Select New Data Source to open the Data Source Configuration Wizard.
5. Select **Site Map** and click **OK**.

SiteMapDataSource1 is added to your project.

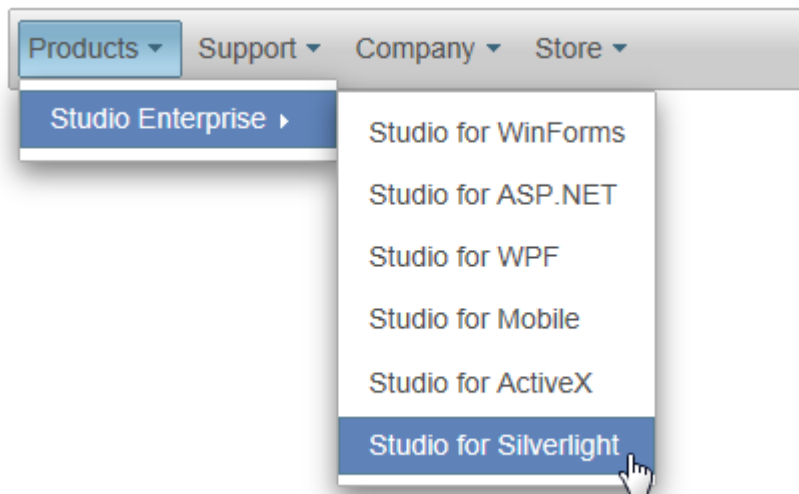
6. Press F5 to run the project and observe the following:

The data from the **Web.sitemap** file is reflected in the **C1Menu** control.



Observe that the control opens with the top-level node, ComponentOne. In the next step, you'll learn how to remove the top-level node so that you just expose the second-level nodes in the [C1Menu](#).

7. Close the browser and return to the project.
8. In Design view, select **SiteMapDataSource** and, in the Properties window, set the **ShowStartingNode** to **False**.
9. Press F5 to run the project and observe that the top-level node has been removed.



Populating C1Menu with XML

This tutorial teaches you how installed templates, add the XML Data Source component to the Web site, assign it to the **C1Menu** control, and then set the binding for the **C1Menu**.

Complete the following steps:

1. From the Toolbox, double-click the [C1Menu](#) icon to add the control to your project.
2. Create and prepare the XML file by completing the following steps:
 - a. Right-click the **App_Data** in the Solution Explorer and select **Add New Item**. The **Add New Item** dialog box appears.
 - b. Select the XML File and rename it "Menu_Hierarchy.xml". The XML file opens.
 - c. In the XML view, add the following data to the **Menu_Hierarchy.xml** document:

[To write code in Source View](#)

```
<?xml version="1.0" encoding="utf-8" ?>

<root>

  <MenuItem Text="CDs">\

    <MenuItem Text="Back to Black"></MenuItem>

    <MenuItem Text="Frank"></MenuItem>

    <MenuItem Text="Nevermind"></MenuItem>

    <MenuItem Text="In Utero"></MenuItem>

  </MenuItem>

  <MenuItem Text="Cassette Tapes">

    <MenuItem Text="Bleach"></MenuItem>

    <MenuItem Text="Cheap Thrills"></MenuItem>

    <MenuItem Text="Dangerous"></MenuItem>

    <MenuItem Text="Bad"></MenuItem>

  </MenuItem>

  <MenuItem Text="Vinyl Records">

    <MenuItem Text="Axis: Bold as Love"></MenuItem>
```

```
<MenuItem Text="Full Circle"></MenuItem>

<MenuItem Text="Off The Wall"></MenuItem>

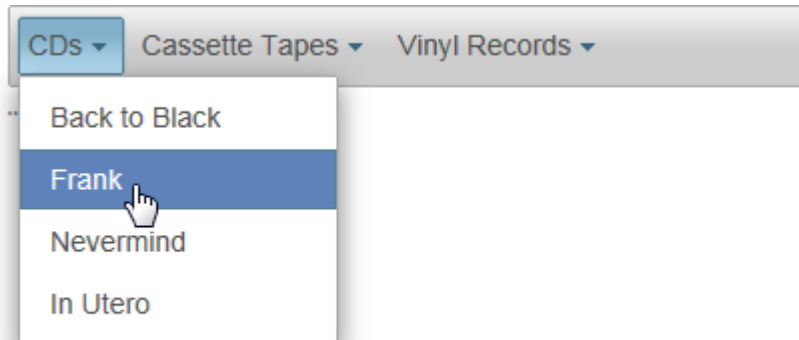
<MenuItem Text="Other Voices"></MenuItem>

</MenuItem>
```

3. Switch back to the Design view of the **.aspx page** and complete the following steps to create a new data source:
 - a. Click **C1Menu**'s smart tag to open the **C1Menu Tasks** menu and then, from the **Choose Data Source** drop-down list, select **New Data Source**.
 - b. The **Data Source Configuration Wizard** dialog box opens.
 - c. Select **XML File** and then click **OK**.
XmlDataSource1 is added to the project.
4. Complete the following steps to configure the data source:
 - a. Click **C1Menu**'s smart tag to open the **C1Menu Tasks** menu; click **Configure Data Source**.
The **Configure Data Source** dialog box opens.
 - b. In the XPath expression text field, enter "root/MenuItem". This will select all MenuItems that are children of root so that the MenuItems are the top-level nodes on the Web page.
 - c. Next to **Data file** text field, click **Browse** to open the **Select XML File** dialog box.
 - d. Select the **App_Data** project folder , and then select **Menu_Hierarchy.xml** from the **Contents of folder** pane
 - e. Click **OK** to close the **Select XML File** dialog box.
 - f. Click **OK** to close the **Configure Data Source** dialog box.
5. Complete the following steps to bind the XML tags to the **C1MenuItems**.
 - a. Click **C1Menu**'s smart tag to open the **C1Menu Tasks** menu and click **Edit Databindings**.
 - b. **The Bindings Collection Editor** dialog box opens.
 - c. Click **Add** to add an empty binding to the project.
 - d. Set the binding's properties as follows:
 - Set the **DataMember** property to "MenuItem".
 - Set the **TextField** property to "Text".
 - e. Click **OK** to close the **Bindings Collection Editor**
6. Press F5 to run the project.

✓ **This topic illustrates the following:**

As the project is running, click through the menu and observe that the data from the **Menu_Hierarchy.xml** file is reflected in the **C1Menu** control.



Saving and Loading a C1Menu from XML

The following tasks show you how to save your **C1Menu** control as an .xml file and then load it into your project using the designer.

Save the C1Menu as XML

To save your tree as an XML file using the designer:

1. Click **C1Menu**'s smart tag and select **Edit Menu** to open the **C1Menu Designer Form**.
2. Navigate to **File | Save as XML**.
3. Name the .xml file for your **C1Menu** and browse to where you would like to save it.
4. Click **OK** to close the **Menu Designer Form** dialog box.

Load an Existing XML C1Menu into your Project

To load the **C1Menu** control you saved as an .xml file into your project:

1. Click **C1Menu**'s smart tag and select **Edit Menu** to open the **C1Menu Designer Form**.
2. Open the **C1Menu Designer Form**.
3. Navigate to **File | Load From XML** and click open to open the existing .xml file.

Load an Existing XML C1Menu in Code

To load the **C1Menu** control you saved as an .xml file into your project:

1. Create an XML file for the **C1Menu** structure.
2. Call the **LoadLayout** method to load the items, passing in the path to the file:

[To write code in Visual Basic](#)

Visual Basic

Copy Code

```
C1Menu1.LoadLayout("c:\\Visual Studio  
2005\\WebSites\\LoadLayoutEX\\App_Data\\C1MenuControl.xml")
```

To write code in C#

C#	Copy Code
<pre>C1Menu1.LoadLayout("c:\\Visual Studio 2005\\WebSites\\LoadLayoutEX\\App_Data\\C1MenuControl.xml");</pre>	

3. Press F5 to run the program.

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 **Maps 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.

- [wijmenu documentation](#)
- [wijmenu 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



Namespaces




C1.Web.Wijmo.Controls Namespace

Overview

[Inheritance Hierarchy](#)

Classes

	Class	Description
	C1BaseStateManager	Base class to support view state management for a server control.
	C1TargetControlBase	Class to be used as base for the wijmo extenders target controls. Class to be used as base for the wijmo target controls.

	C1TargetControlHelper	
	C1TargetControlHelper<T>	
	WijChartNavigator	

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
```

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

See Also

Reference






[C1TargetControlBase Members](#)















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














Members









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

Public Properties

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






	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	Controls	(Inherited from System.Web.UI.Control)
	ControlStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ControlStyleCreated	(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)
	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.

	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	UniqueID	(Inherited from System.Web.UI.Control)
	UseCDN	Determines whether the widget extender must load client resources from CDN (Content Delivery Network) path given by property CDNPath.
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Width	(Inherited from System.Web.UI.WebControls.WebControl)
	WijmoControlMode	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.
	WijmoCssAdapter	Indicates the control applies the theme of JQuery UI or Bootstrap.

[Top](#)







Public Methods

	Name	Description
	ApplyStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ApplyStyleSheetSkin	(Inherited from System.Web.UI.Control)
	CopyBaseAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	DataBind	(Inherited from System.Web.UI.Control)
	Dispose	(Inherited from System.Web.UI.Control)

 FindControl	(Inherited from System.Web.UI.Control)
 Focus	(Inherited from System.Web.UI.Control)
 GetPropertyValues<V>	Gets the property value by property name.
 GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
 GetScriptDescriptors	When overridden in a derived class, registers the WidgetDescriptor objects for the control.
 GetScriptReferences	When overridden in a derived class, registers an additional script libraries for the extender control.
 GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
 HasControls	(Inherited from System.Web.UI.Control)
 MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
 RegisterDesignTimeStyleSheet	Register design time stylesheet
 RenderBeginTag	(Inherited from System.Web.UI.WebControls.WebControl)
 RenderControl	(Inherited from System.Web.UI.Control)
 RenderEndTag	(Inherited from System.Web.UI.WebControls.WebControl)
 ResolveClientUrl	(Inherited from System.Web.UI.Control)
 ResolveUrl	(Inherited from System.Web.UI.Control)
 SetPropertyValues<V>	Sets the property value by property name.
 SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)

[Top](#)

Public Events

	Name	Description
	DataBinding	(Inherited from System.Web.UI.Control)
	Disposed	(Inherited from System.Web.UI.Control)
	Init	(Inherited from System.Web.UI.Control)
	Load	(Inherited from System.Web.UI.Control)
	PreRender	(Inherited from System.Web.UI.Control)
	Unload	(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
	ApplyStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ApplyStyleSheetSkin	(Inherited from System.Web.UI.Control)
	CopyBaseAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	DataBind	(Inherited from System.Web.UI.Control)
	Dispose	(Inherited from System.Web.UI.Control)
	FindControl	(Inherited from System.Web.UI.Control)

 Focus	(Inherited from System.Web.UI.Control)
 GetPropertyValues<V>	Gets the property value by property name.
 GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
 GetScriptDescriptors	When overridden in a derived class, registers the WidgetDescriptor objects for the control.
 GetScriptReferences	When overridden in a derived class, registers an additional script libraries for the extender control.
 GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
 HasControls	(Inherited from System.Web.UI.Control)
 MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
 RegisterDesignTimeStyleSheet	Register design time stylesheet
 RenderBeginTag	(Inherited from System.Web.UI.WebControls.WebControl)
 RenderControl	(Inherited from System.Web.UI.Control)
 RenderEndTag	(Inherited from System.Web.UI.WebControls.WebControl)
 ResolveClientUrl	(Inherited from System.Web.UI.Control)
 ResolveUrl	(Inherited from System.Web.UI.Control)
 SetPropertyValues<V>	Sets the property value by property name.
 SetRenderMethodDelegate	(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)	
<pre>Public Overridable Function GetScriptDescriptors() As System.Collections.Generic.IEnumerable(Of ScriptDescriptor)</pre>	
C#	
<pre>public virtual System.Collections.Generic.IEnumerable<ScriptDescriptor> GetScriptDescriptors()</pre>	

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

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
	AccessKey	(Inherited from System.Web.UI.WebControls.WebControl)
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from

		System.Web.UI.WebControls.WebControl)
	CDNDependencyPaths	A string array value specifies all css/js references that wijmo control depends on.
	CDNPath	Content Delivery Network path.
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	Controls	(Inherited from System.Web.UI.Control)
	ControlStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ControlStyleCreated	(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)
	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.
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	UniqueID	(Inherited from System.Web.UI.Control)
	UseCDN	Determines whether the widget extender must load client resources from CDN (Content Delivery Network) path given by property CDNPath.
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Width	(Inherited from System.Web.UI.WebControls.WebControl)
	WijmoControlMode	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.
	WijmoCssAdapter	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)	
<pre> <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() </pre>	
C#	
<pre> [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;} </pre>	

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(DesignerSeriali
zationVisibility.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)	
<pre><WidgetOptionAttribute()> <System.ComponentModel.DefaultValueAttribute()> <C1DescriptionAttribute(Key="C1Base.Enabled", Description="Indicates whether control is enabled.")> <C1CategoryAttribute("Behavior")> Public Overrides Property Enabled As System.Boolean</pre>	
C#	
<pre>[WidgetOption()] [System.ComponentModel.DefaultValue()] [C1Description(Key="C1Base.Enabled", Description="Indicates whether control is enabled.")] [C1Category("Behavior")] public override System.bool Enabled {get; set;}</pre>	

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)	
<pre> <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 </pre>	
C#	
<pre> [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;} </pre>	

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)	
<pre><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</pre>	
C#	
<pre>[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;}</pre>	

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)	
<pre><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(DesignerSerializationVisibility.Hidden)> Public Overridable Property UseCDN As System.Boolean</pre>	
C#	
<pre>[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(DesignerSerializationVisibility.Hidden)] public virtual System.bool UseCDN {get; set;}</pre>	

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)	
<pre><C1CategoryAttribute("Behavior")></pre>	

```
<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
zationVisibility.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(DesignerSeriali
```



```
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](#)

C1TargetControlHelper

Object Model

C1TargetControlHelper

Syntax

```
Visual Basic (Declaration)
```

```
Friend Class C1TargetControlHelper
```

```
C#
```

```
internal class C1TargetControlHelper
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1TargetControlHelper

[C1.Web.Wijmo.Controls.C1TargetControlHelper<T>](#)

See Also

Reference

[C1TargetControlHelper Members](#)

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

Overview

Object Model

C1TargetControlHelper

Syntax

Visual Basic (Declaration)

```
Friend Class C1TargetControlHelper
```

C#

```
internal class C1TargetControlHelper
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1TargetControlHelper

[C1.Web.Wijmo.Controls.C1TargetControlHelper<T>](#)

See Also

Reference


[C1TargetControlHelper Members](#)

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

Members


The following tables list the members exposed by [C1TargetControlHelper](#).

Public Constructors

	Name	Description
	C1TargetControlHelper Constructor	

[Top](#)

Public Methods

	Name	Description
  S	ResolveEmbeddedResourceUrl	

[Top](#)

See Also

Reference

[C1TargetControlHelper Class](#)

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

C1TargetControlHelper Constructor

Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1TargetControlHelper()	

See Also

Reference


[C1TargetControlHelper Class](#)

[C1TargetControlHelper Members](#)

Methods

For a list of all members of this type, see [C1TargetControlHelper members](#).

Public Methods

	Name	Description
  S	ResolveEmbeddedResourceUrl	

[Top](#)

See Also

Reference

[C1TargetControlHelper Class](#)

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

ResolveEmbeddedResourceUrl Method

Syntax

Visual Basic (Declaration)

```
Public Shared Function ResolveEmbeddedResourceUrl( _  
    ByVal resourceName As System.String, _  
    ByVal page As System.Web.UI.Page, _  
    ByVal combinedScript As System.Boolean _  
) As System.String
```

C#

```
public static System.string ResolveEmbeddedResourceUrl(  
    System.string resourceName,  
    System.Web.UI.Page page,  
    System.bool combinedScript  
)
```

Parameters

resourceName

page

combinedScript

See Also

Reference

[C1TargetControlHelper Class](#)

[C1TargetControlHelper Members](#)

C1TargetControlHelper<T>

Object Model

C1TargetControlHelper<T>

Syntax

Visual Basic (Declaration)

```
Friend Class C1TargetControlHelper(Of T As  
{System.Web.UI.WebControls.WebControl, IC1TargetControl})  
    Inherits C1TargetControlHelper
```

C#

```
internal class C1TargetControlHelper<T> : C1TargetControlHelper  
where T: System.Web.UI.WebControls.WebControl, IC1TargetControl
```

Type Parameters

T

Inheritance Hierarchy

System.Object

[C1.Web.Wijmo.Controls.C1TargetControlHelper](#)

C1.Web.Wijmo.Controls.C1TargetControlHelper<T>

See Also

Reference

[C1TargetControlHelper<T> Members](#)

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

Overview

Object Model

C1TargetControlHelper<T>

Syntax

Visual Basic (Declaration)

```
Friend Class C1TargetControlHelper(Of T As  
{System.Web.UI.WebControls.WebControl, IC1TargetControl})  
    Inherits C1TargetControlHelper
```

C#

```
internal class C1TargetControlHelper<T> : C1TargetControlHelper  
where T: System.Web.UI.WebControls.WebControl, IC1TargetControl
```

Type Parameters

T

Inheritance Hierarchy

System.Object

[C1.Web.Wijmo.Controls.C1TargetControlHelper](#)

C1.Web.Wijmo.Controls.C1TargetControlHelper<T>

See Also

Reference


[C1TargetControlHelper<T> Members](#)

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

Members





The following tables list the members exposed by [C1TargetControlHelper<T>](#).

Public Constructors

	Name	Description
	C1TargetControlHelper<T> Constructor	













[Top](#)

Public Properties

	Name	Description
	Control	
	JsonSerializableHelper	
	Page	
	ScriptManager	

[Top](#)

Public Methods

	Name	Description
	EnsureEnabledState	
	GetScriptDescriptors	Overloaded. When overridden in a derived class, registers the WidgetDescriptor objects for the control.
	RegisterDesignTimeStyleSheet	Register design-time CSS stylesheet.
	RegisterDesignTimeStyleSheets	
	RegisterIncludes	
	RegisterOnSubmitStatement	
	RegisterRunTimeStylesheets	
	RegisterScriptDescriptors	
	RegisterScriptDescriptorsWithoutScriptManager	
	RegisterScriptDescriptorsWithScriptManager	
	RenderJsonDataField	
	ShowAbout	Show about message box.

[Top](#)

See Also

Reference

[C1TargetControlHelper<T> Class](#)

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

C1TargetControlHelper<T> Constructor

Syntax

Visual Basic (Declaration)	
<pre>Public Function New(_ ByVal <i>control</i> As T _)</pre>	
C#	
<pre>public C1TargetControlHelper<T>(T <i>control</i>)</pre>	

Parameters

control

See Also

Reference

[C1TargetControlHelper<T> Class](#)







[C1TargetControlHelper<T> Members](#)

Methods

For a list of all members of this type, see [C1TargetControlHelper<T> members](#).

Public Methods

	Name	Description
⇒	EnsureEnabledState	
⇒	GetScriptDescriptors	Overloaded. When overridden in a derived class, registers the <i>WidgetDescriptor</i> objects for the control.
⇒	RegisterDesignTimeStyleSheet	Register design-time CSS stylesheet.
⇒	RegisterDesignTimeStyleSheets	
⇒	RegisterIncludes	
⇒	RegisterOnSubmitStatement	

	RegisterRunTimeStylesheets	
	RegisterScriptDescriptors	
	RegisterScriptDescriptorsWithoutScriptManager	
	RegisterScriptDescriptorsWithScriptManager	
	RenderJsonDataField	
	ShowAbout	Show about message box.

[Top](#)

See Also

Reference

[C1TargetControlHelper<T> Class](#)

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

EnsureEnabledState Method

Syntax

Visual Basic (Declaration)

```
Public Sub EnsureEnabledState()
```

C#

```
public void EnsureEnabledState()
```

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

GetScriptDescriptors Method

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

Overload List

Overload	Description
GetScriptDescriptors()	When overridden in a derived class, registers the WidgetDescriptor objects for the control.
GetScriptDescriptors(String)	When overridden in a derived class, registers the WidgetDescriptor objects for the control.

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

GetScriptDescriptors() Method

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

Syntax

Visual Basic (Declaration)	
<pre>Public Overloads Function GetScriptDescriptors() As System.Collections.Generic.IEnumerable(Of ScriptDescriptor)</pre>	
C#	
<pre>public System.Collections.Generic.IEnumerable<ScriptDescriptor> GetScriptDescriptors()</pre>	

Return Value

An enumeration of WidgetDescriptor objects.

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

[Overload List](#)

GetScriptDescriptors(String) Method

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

Syntax

Visual Basic (Declaration)

```
Public Overloads Function GetScriptDescriptors( _
    ByVal targetSelector As System.String _
) As System.Collections.Generic.IEnumerable(Of ScriptDescriptor)
```

C#

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

Parameters*targetSelector*

The target selector.

Return Value

An enumeration of WidgetDescriptor objects.

See Also**Reference**

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

[Overload List](#)

RegisterDesignTimeStyleSheet Method

Register design-time CSS 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

cssResourceName

CSS resource name.

Remarks

The *cssResourceName* is the link to an resource element.

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

RegisterDesignTimeStyleSheets Method

Syntax

Visual Basic (Declaration)

```
Public Sub RegisterDesignTimeStyleSheets( _  
    ByVal writer As System.Web.UI.HtmlTextWriter _  
)
```

C#

```
public void RegisterDesignTimeStyleSheets(  
    System.Web.UI.HtmlTextWriter writer  
)
```

Parameters

writer

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

RegisterIncludes Method

Syntax

Visual Basic (Declaration)

```
Public Sub RegisterIncludes()
```

C#

```
public void RegisterIncludes()
```

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

RegisterOnSubmitStatement Method

Syntax

Visual Basic (Declaration)

```
Public Sub RegisterOnSubmitStatement( _  
    Optional ByVal callback As System.String _  
)
```

C#

```
public void RegisterOnSubmitStatement(  
    System.string callback  
)
```

Parameters

callback

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

RegisterRunTimeStylesheets Method

Syntax

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

Public Sub RegisterRunTimeStylesheets()

C#	
----	--

public void RegisterRunTimeStylesheets()
--

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

RegisterScriptDescriptors Method

Syntax

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

Public Sub RegisterScriptDescriptors()
--

C#	
----	--

public void RegisterScriptDescriptors()

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

RegisterScriptDescriptorsWithoutScriptManager Method

Syntax

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

Public Sub RegisterScriptDescriptorsWithoutScriptManager()
--

C#	
----	--

public void RegisterScriptDescriptorsWithoutScriptManager()

See Also

Reference

[C1TargetControlHelper<T> Class](#)[C1TargetControlHelper<T> Members](#)

RegisterScriptDescriptorsWithScriptManager Method

Syntax

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

See Also

Reference

[C1TargetControlHelper<T> Class](#)[C1TargetControlHelper<T> Members](#)

RenderJsonDataField Method

Syntax

Visual Basic (Declaration)	
Public Sub RenderJsonDataField(_ ByVal <i>writer</i> As System.Web.UI.HtmlTextWriter _)	
C#	
public void RenderJsonDataField(System.Web.UI.HtmlTextWriter <i>writer</i>)	

Parameters

writer

See Also

Reference

[C1TargetControlHelper<T> Class](#)[C1TargetControlHelper<T> Members](#)

ShowAbout Method

Show about message box.

Syntax

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

See Also

Reference





[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

Properties

For a list of all members of this type, see [C1TargetControlHelper<T> members](#).

Public Properties

	Name	Description
	Control	
	JsonSerializableHelper	
	Page	
	ScriptManager	

[Top](#)

See Also

Reference

[C1TargetControlHelper<T> Class](#)

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

Control Property

Syntax

Visual Basic (Declaration)	
Public ReadOnly Property Control As T	
C#	
public T Control {get;}	

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

JsonSerializableHelper Property

Syntax

Visual Basic (Declaration)	
Public ReadOnly Property JsonSerializableHelper As JsonSerializableHelper	
C#	
public JsonSerializableHelper JsonSerializableHelper {get;}	

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

Page Property

Syntax

Visual Basic (Declaration)	
Public ReadOnly Property Page As System.Web.UI.Page	
C#	

```
public System.Web.UI.Page Page {get;}
```

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

ScriptManager Property

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property ScriptManager As System.Web.UI.ScriptManager
```

C#

```
public System.Web.UI.ScriptManager ScriptManager {get;}
```

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> 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
	WijChartNavigator Constructor	

[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.C1Menu

Namespace


Overview

Classes


	Class	Description
	C1Menu	Represents a Menu in an ASP.NET Web page.

	C1MenuEventArgs	C1MenuEventArgs is used by the ItemClick event.
	C1MenuItem	
	C1MenuItemBinding	
	C1MenuItemBindingCollection	Represents a collection of C1MenuItemBinding objects that are used by data-bound C1Menu control.
	C1MenuItemCollection	Represents a collection of C1MenuItemBase objects that are used by Items property.
	C1MenuItemCollectionChangedEventArgs	C1MenuItemCollectionChangedEventArgs is used for the C1MenuItemCollectionChange event.
	C1MenuItemTemplateContainer	Represents a container that holds the contents of a templated menu item in a C1Menu control.
	C1MenuSerializer	

Interfaces






	Interface	Description
	IC1MenuItemCollectionOwner	IC1MenuItemCollectionOwner interface.

Delegates

	Delegate	Description
	C1MenuEventHandler	Delegate type for handling events that are related to the items.

Enumerations

	Enumeration	Description
--	-------------	-------------

	ChangeStatus	Specifies IC1MenuItemCollectionOwner change status.
	ImagePosition	Decides which side the image displays on menu item.
	MenuMode	Decides which mode the menu displays.
	SubmenuTriggerEvent	If menu is flyout mode, this value specially decide which event handle to show the submenu.
	TriggerEvent	If menu is flyout mode, this value decide which event handle to show the menu and submenu.

See Also

Reference

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

Classes

C1Menu

Represents a Menu in an ASP.NET Web page.

Object Model

C1Menu

Syntax

Visual Basic (Declaration)

```
<System.Web.UI.ToolboxDataAttribute("<{0}:C1Menu
runat=server></{0}:C1Menu>")>
<System.ComponentModel.DesignerAttribute(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1Menu.C1MenuDesigner,
C1.Web.Wijmo.Controls.Design.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<System.ComponentModel.LicenseProviderAttribute()>
<WidgetDependenciesAttribute(System.Object[])>
<System.Drawing.ToolboxBitmapAttribute()>
```

Public Class C1Menu

Inherits

C1.Web.Wijmo.Controls.Base.C1TargetHierarchicalDataBoundControlBase

Implements [IC1MenuItemCollectionOwner](#)

C#

```
[System.Web.UI.ToolboxData("<{0}:C1Menu runat=server></{0}:C1Menu>")]
[System.ComponentModel.Designer(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1Menu.C1MenuDesigner,
C1.Web.Wijmo.Controls.Design.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[System.ComponentModel.LicenseProvider()]
[WidgetDependencies(System.Object[])]
[System.Drawing.ToolboxBitmap()]
public class C1Menu :
C1.Web.Wijmo.Controls.Base.C1TargetHierarchicalDataBoundControlBase,
IC1MenuItemCollectionOwner
```

Inheritance Hierarchy

System.Object

System.Web.UI.Control

System.Web.UI.WebControls.WebControl

System.Web.UI.WebControls.BaseDataBoundControl

System.Web.UI.WebControls.HierarchicalDataBoundControl

C1.Web.Wijmo.Controls.C1Menu.C1Menu

See Also

Reference

[C1Menu Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

Represents a Menu in an ASP.NET Web page.

Object Model

C1Menu

Syntax

Visual Basic (Declaration)

```

<System.Web.UI.ToolboxDataAttribute("<{0}:C1Menu
runat=server></{0}:C1Menu>")>
<System.ComponentModel.DesignerAttribute(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1Menu.C1MenuDesigner,
C1.Web.Wijmo.Controls.Design.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<System.ComponentModel.LicenseProviderAttribute()>
<WidgetDependenciesAttribute(System.Object[])>
<System.Drawing.ToolboxBitmapAttribute()>
Public Class C1Menu
    Inherits
C1.Web.Wijmo.Controls.Base.C1TargetHierarchicalDataBoundControlBase
    Implements IC1MenuItemCollectionOwner

```

C#

```

[System.Web.UI.ToolboxData("<{0}:C1Menu runat=server></{0}:C1Menu>")]
[System.ComponentModel.Designer(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1Menu.C1MenuDesigner,
C1.Web.Wijmo.Controls.Design.4, Version=4.0.20151.207, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[System.ComponentModel.LicenseProvider()]
[WidgetDependencies(System.Object[])]
[System.Drawing.ToolboxBitmap()]
public class C1Menu :
C1.Web.Wijmo.Controls.Base.C1TargetHierarchicalDataBoundControlBase,
IC1MenuItemCollectionOwner

```

Inheritance Hierarchy

```

System.Object
    System.Web.UI.Control
        System.Web.UI.WebControls.WebControl
            System.Web.UI.WebControls.BaseDataBoundControl
                System.Web.UI.WebControls.HierarchicalDataBoundControl
                    C1.Web.Wijmo.Controls.C1Menu.C1Menu

```

See Also

Reference


[C1Menu Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members








The following tables list the members exposed by [C1Menu](#).
















Public Constructors















	Name	Description
	C1Menu Constructor	Initializes a new instance of the C1Menu class.
















[Top](#)















Public Properties


	Name	Description
	AccessKey	(Inherited from System.Web.UI.WebControls.WebControl)
	Animation	Defines the animation to show or hide the submenu in flyout mode. If showAnimation or hideAnimation is not specified use this property.
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from System.Web.UI.WebControls.WebControl)
	AutoPostBack	Sets or retrieves a value that indicates whether or not the control posts back to the server each time a user interacts with the control.
	BackLink	Determines whether the iPod menu shows a back link or a breadcrumb header in the menu.
	BackLinkText	Gets and sets the text of the back link.

	Checkable	Gets or sets the item can be checked
	ChildItemsTemplate	If specified, this template will be applied for all menu items instead of top level menu items.
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	Controls	(Inherited from System.Web.UI.Control)
	ControlStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ControlStyleCreated	(Inherited from System.Web.UI.WebControls.WebControl)
	CrumbDefaultText	Gets and sets the top breadcrumb's default text.
	CssClass	(Inherited from System.Web.UI.WebControls.WebControl)
	DataBindings	
	DataSource	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	DataSourceID	(Inherited from System.Web.UI.WebControls.HierarchicalDataBoundControl)
	Direction	A value that indicates menu's direction.
	EnableTheming	(Inherited from System.Web.UI.WebControls.WebControl)
	EnableViewState	(Inherited from System.Web.UI.Control)

	EnsureSubmenuOnBody	A value indicating the submenu will be append to the body or menu container. If the value is true, the submenu will be appended to body element, else it will append to the menu container. Default: false. Type: Boolean.
	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	Height	Overridden. Gets or sets the height of the menu.
	HideAnimation	Defines the animation to hide submenu in flyout mode.
	HideDelay	Gets or sets delay time to hide submenu.
	ID	(Inherited from System.Web.UI.Control)
	IsDirty	Gets or sets a value indicates whether this instance is dirty.
	Items	Gets the menu items.
	ItemsTemplate	If specified, this template will be applied for all menu items that does not have other defined templates.
	MaxHeight	Determines the iPod-style menu's maximum height.
	Mode	Defines the behavior of the submenu whether it is a popup menu or an iPod-style navigation list.
	NamingContainer	(Inherited from System.Web.UI.Control)
	OnClientBlur	Triggered when a menu item loses focus.
	OnClientFocus	Triggered when a menu item gets the focus, either when the mouse is used to hover over it (on hover) or when the cursor keys are used on the keyboard















		(navigation with cursor key) focus.
	OnClientHidden	Triggered once a menu or submenu has hidden.
	OnClientHidding	Triggered before hidding a menu or submenu.
	OnClientSelect	Triggered when a menu item is selected.
	OnClientShowing	Triggered before showing a menu or submenu.
	OnClientShown	Triggered once a menu or submenu has shown.
	Orientation	Controls the root menu's orientation. All submenus are vertical regardless of the orientation of the root menu.
	Owner	Owner of the menu. Always returns null.
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	Position	Specifies the location and orientation of the menu relative to the button or link used to open it. Configuration for the Position Utility Of option is excluded; it is always configured by the widget. Collision also controls collision detection automatically.
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	ShowAnimation	Defines the show animation to show submenu in flyout mode.
	ShowDelay	Gets or sets delay time to show submenu.
	Site	(Inherited from System.Web.UI.Control)
	SkinID	(Inherited from







		System.Web.UI.WebControls.WebControl)
	SlidingAnimation	Defines the sliding animation in slide mode.
	Style	(Inherited from System.Web.UI.WebControls.WebControl)
	SubmenuTriggerEvent	Specifies the event used to show the submenu.
	SupportsDisabledAttribute	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	TabIndex	(Inherited from System.Web.UI.WebControls.WebControl)
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	TopItemsTemplate	If specified, this template will be applied for top level menu items.
	TopLinkText	Gets and sets the text of the top link.
	Trigger	Specifies the event used to show the menu.
	TriggerEvent	Set the event used to show the menu and submenu(If "SubmenuTriggerEvent" is "Default").
	UniqueID	(Inherited from System.Web.UI.Control)
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)

	Width	Overridden. Gets or sets the width of the menu.
---	-------	---

[Top](#)









Public Methods

	Name	Description
	ApplyStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ApplyStyleSheetSkin	(Inherited from System.Web.UI.Control)
	CopyBaseAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	DataBind	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	Dispose	(Inherited from System.Web.UI.Control)
	FindControl	(Inherited from System.Web.UI.Control)
	Focus	(Inherited from System.Web.UI.Control)
	GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
	GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
	HasControls	(Inherited from System.Web.UI.Control)
	LoadLayout	Overloaded. Loads control layout properties from the file.
	MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	OnItemClick	Raise ItemClick event.
	RenderBeginTag	Overridden. Renders the HTML opening tag of the control to the specified writer. This method is used primarily by control developers.

	RenderControl	(Inherited from System.Web.UI.Control)
	RenderEndTag	Overridden. Renders the HTML closing tag of the control into the specified writer. This method is used primarily by control developers.
	ResolveClientUrl	(Inherited from System.Web.UI.Control)
	ResolveUrl	(Inherited from System.Web.UI.Control)
	SaveLayout	Overloaded. Saves the control layout properties to the file.
	SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)

[Top](#)

Public Events

	Name	Description
	DataBinding	(Inherited from System.Web.UI.Control)
	DataBound	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	Disposed	(Inherited from System.Web.UI.Control)
	Init	(Inherited from System.Web.UI.Control)
	ItemClick	Fires when an item is clicked
	Load	(Inherited from System.Web.UI.Control)
	PreRender	(Inherited from System.Web.UI.Control)
	Unload	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1Menu Class](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1Menu Constructor

Initializes a new instance of the [C1Menu](#) class.

Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1Menu()	

See Also

Reference















[C1Menu Class](#)[C1Menu Members](#)

Methods

For a list of all members of this type, see [C1Menu members](#).

Public Methods

	Name	Description
≡	ApplyStyle	(Inherited from System.Web.UI.WebControls.WebControl)
≡	ApplyStyleSheetSkin	(Inherited from System.Web.UI.Control)
≡	CopyBaseAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
≡	DataBind	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
≡	Dispose	(Inherited from System.Web.UI.Control)
≡	FindControl	(Inherited from System.Web.UI.Control)

 Focus	(Inherited from System.Web.UI.Control)
 GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
 GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
 HasControls	(Inherited from System.Web.UI.Control)
 LoadLayout	Overloaded. Loads control layout properties from the file.
 MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
 OnItemClick	Raise ItemClick event.
 RenderBeginTag	Overridden. Renders the HTML opening tag of the control to the specified writer. This method is used primarily by control developers.
 RenderControl	(Inherited from System.Web.UI.Control)
 RenderEndTag	Overridden. Renders the HTML closing tag of the control into the specified writer. This method is used primarily by control developers.
 ResolveClientUrl	(Inherited from System.Web.UI.Control)
 ResolveUrl	(Inherited from System.Web.UI.Control)
 SaveLayout	Overloaded. Saves the control layout properties to the file.
 SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1Menu Class](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

LoadLayout Method

Loads control layout properties from the file.

Overload List

Overload	Description
LoadLayout(String)	Loads control layout properties from the file.
LoadLayout(Stream)	Load control layout properties from the stream.
LoadLayout(String,LayoutType)	Loads control layout properties with specified types from the file.
LoadLayout(Stream,LayoutType)	Loads the control layout properties with specified types from the stream.

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

LoadLayout(String) Method

Loads control layout properties from the file.

Syntax

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

Parameters

filename

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

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

[Overload List](#)

LoadLayout(Stream) Method

Load control layout properties from the stream.

Syntax

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

Parameters

stream

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

See Also

Reference

[C1Menu Class](#)

[C1Menu 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

[C1Menu Class](#)

[C1Menu 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

[C1Menu Class](#)
[C1Menu Members](#)
[Overload List](#)

OnItemClick Method

Raise ItemClick event.

Syntax

Visual Basic (Declaration)

```
Public Overridable Sub OnItemClick( _  
    ByVal e As C1MenuEventArgs _  
)
```

C#

```
public virtual void OnItemClick(  
    C1MenuEventArgs e  
)
```

Parameters

e

A C1MenuEventArgs object that contains the event data.

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

RenderBeginTag Method

A System.Web.UI.HtmlTextWriter that represents the output stream to render HTML content on the client.

Renders the HTML opening tag of the control to the specified writer. This method is used primarily by control developers.

Syntax

Visual Basic (Declaration)	
<pre>Public Overrides Sub RenderBeginTag(_ ByVal <i>writer</i> As System.Web.UI.HtmlTextWriter _)</pre>	
C#	
<pre>public override void RenderBeginTag(System.Web.UI.HtmlTextWriter <i>writer</i>)</pre>	

Parameters

writer

A System.Web.UI.HtmlTextWriter that represents the output stream to render HTML content on the client.

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

RenderEndTag Method

A System.Web.UI.HtmlTextWriter that represents the output stream to render HTML content on the client.

Renders the HTML closing tag of the control into the specified writer. This method is used primarily by control developers.

Syntax

Visual Basic (Declaration)	
<pre>Public Overrides Sub RenderEndTag(_ ByVal writer As System.Web.UI.HtmlTextWriter _)</pre>	
C#	
<pre>public override void RenderEndTag(System.Web.UI.HtmlTextWriter writer)</pre>	

Parameters

writer

A `System.Web.UI.HtmlTextWriter` that represents the output stream to render HTML content on the client.

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

SaveLayout Method

Saves the control layout properties to the file.

Overload List

Overload	Description
SaveLayout(String)	Saves the control layout properties to the file.
SaveLayout(Stream)	Saves control layout properties to the stream.

See Also

Reference

[C1Menu Class](#)[C1Menu Members](#)

SaveLayout(String) Method

Saves the control layout properties to the file.

Syntax

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

Parameters

filename

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

See Also

Reference

[C1Menu Class](#)[C1Menu Members](#)[Overload List](#)

SaveLayout(Stream) Method

Saves control layout properties to the stream.

Syntax

Visual Basic (Declaration)	
Public Overloads Sub SaveLayout(_ ByVal <i>stream</i> 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

[C1Menu Class](#)






[C1Menu Members](#)
















[Overload List](#)















Properties















For a list of all members of this type, see [C1Menu members](#).
















Public Properties




	Name	Description
	AccessKey	(Inherited from System.Web.UI.WebControls.WebControl)
	Animation	Defines the animation to show or hide the submenu in flyout mode. If showAnimation or hideAnimation is not specified use this property.
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from System.Web.UI.WebControls.WebControl)
	AutoPostBack	Sets or retrieves a value that indicates whether or not the control posts back to the server each time a user interacts with the control.

	BackLink	Determines whether the iPod menu shows a back link or a breadcrumb header in the menu.
	BackLinkText	Gets and sets the text of the back link.
	Checkable	Gets or sets the item can be checked
	ChildItemsTemplate	If specified, this template will be applied for all menu items instead of top level menu items.
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	Controls	(Inherited from System.Web.UI.Control)
	ControlStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ControlStyleCreated	(Inherited from System.Web.UI.WebControls.WebControl)
	CrumbDefaultText	Gets and sets the top breadcrumb's default text.
	CssClass	(Inherited from System.Web.UI.WebControls.WebControl)
	DataBindings	
	DataSource	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	DataSourceID	(Inherited from System.Web.UI.WebControls.HierarchicalDataBoundControl)
	Direction	A value that indicates menu's direction.

	EnableTheming	(Inherited from System.Web.UI.WebControls.WebControl)
	EnableViewState	(Inherited from System.Web.UI.Control)
	EnsureSubmenuOnBody	A value indicating the submenu will be append to the body or menu container. If the value is true, the submenu will be appended to body element, else it will append to the menu container. Default: false. Type: Boolean.
	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	Height	Overridden. Gets or sets the height of the menu.
	HideAnimation	Defines the animation to hide submenu in flyout mode.
	HideDelay	Gets or sets delay time to hide submenu.
	ID	(Inherited from System.Web.UI.Control)
	IsDirty	Gets or sets a value indicates whether this instance is dirty.
	Items	Gets the menu items.
	ItemsTemplate	If specified, this template will be applied for all menu items that does not have other defined templates.
	MaxHeight	Determines the iPod-style menu's maximum height.
	Mode	Defines the behavior of the submenu whether it is a popup menu or an iPod-style navigation list.
	NamingContainer	(Inherited from System.Web.UI.Control)

	OnClientBlur	Triggered when a menu item loses focus.
	OnClientFocus	Triggered when a menu item gets the focus, either when the mouse is used to hover over it (on hover) or when the cursor keys are used on the keyboard (navigation with cursor key) focus.
	OnClientHidden	Triggered once a menu or submenu has hidden.
	OnClientHidding	Triggered before hidding a menu or submenu.
	OnClientSelect	Triggered when a menu item is selected.
	OnClientShowing	Triggered before showing a menu or submenu.
	OnClientShown	Triggered once a menu or submenu has shown.
	Orientation	Controls the root menu's orientation. All submenus are vertical regardless of the orientation of the root menu.
	Owner	Owner of the menu. Always returns null.
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	Position	Specifies the location and orientation of the menu relative to the button or link used to open it. Configuration for the Position Utility Of option is excluded; it is always configured by the widget. Collision also controls collision detection automatically.
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	ShowAnimation	Defines the show animation to show submenu in flyout mode.

	ShowDelay	Gets or sets delay time to show submenu.
	Site	(Inherited from System.Web.UI.Control)
	SkinID	(Inherited from System.Web.UI.WebControls.WebControl)
	SlidingAnimation	Defines the sliding animation in slide mode.
	Style	(Inherited from System.Web.UI.WebControls.WebControl)
	SubmenuTriggerEvent	Specifies the event used to show the submenu.
	SupportsDisabledAttribute	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	TabIndex	(Inherited from System.Web.UI.WebControls.WebControl)
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	TopItemsTemplate	If specified, this template will be applied for top level menu items.
	TopLinkText	Gets and sets the text of the top link.
	Trigger	Specifies the event used to show the menu.
	TriggerEvent	Set the event used to show the menu and submenu(If "SubmenuTriggerEvent" is "Default").
	UniqueID	(Inherited from System.Web.UI.Control)

	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Width	Overridden. Gets or sets the width of the menu.

[Top](#)

See Also

Reference

[C1Menu Class](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Animation Property

Defines the animation to show or hide the submenu in flyout mode. If showAnimation or hideAnimation is not specified use this property.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<C1DescriptionAttribute(Key="C1Menu.Animation", Description="Defines the
animation to show or hide the submenu in flyout mode")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Content)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property Animation As Animation
```

C#

```
[System.ComponentModel.NotifyParentProperty(true)]
[C1Description(Key="C1Menu.Animation", Description="Defines the animation to
show or hide the submenu in flyout mode")]
[WidgetOption()]
[C1Category("Behavior")]
```

```
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[Layout(LayoutType.Behavior)]
public Animation Animation {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

AutoPostBack Property

Sets or retrieves a value that indicates whether or not the control posts back to the server each time a user interacts with the control.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1Menu.AutoPostBack", Description="Sets or
retrieves a value that indicates whether or not the control posts back to the
server each time a user interacts with the control")>
Public Property AutoPostBack As System.Boolean
```

C#

```
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1Menu.AutoPostBack", Description="Sets or retrieves a
value that indicates whether or not the control posts back to the server each
time a user interacts with the control")]
public System.bool AutoPostBack {get; set;}
```

See Also

Reference

[C1Menu Class](#)[C1Menu Members](#)

BackLink Property

Determines whether the iPod menu shows a back link or a breadcrumb header in the menu.

Syntax

Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1Menu.BackLink", Description="Determines
whether the iPod menu shows a back link or a breadcrumb header in the menu")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Property BackLink As System.Boolean

```

C#

```

[C1Description(Key="C1Menu.BackLink", Description="Determines whether the
iPod menu shows a back link or a breadcrumb header in the menu")]
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
public System.bool BackLink {get; set;}

```

See Also

Reference

[C1Menu Class](#)[C1Menu Members](#)

BackLinkText Property

Gets and sets the text of the back link.

Syntax

Visual Basic (Declaration)

```

<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1Menu.BackLinkText", Description="Gets and sets
the text of the back link")>

```



```
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property BackLinkText As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1Menu.BackLinkText", Description="Gets and sets the text
of the back link")]
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
public System.string BackLinkText {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

Checkable Property

Gets or sets the item can be checked

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.Checkable", Description="Gets or sets the
item can be checked")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
Public Property Checkable As System.Boolean
```

C#

```
[C1Description(Key="C1Menu.Checkable", Description="Gets or sets the item can
be checked")]
[WidgetOption()]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Behavior)]
```

```
public System.bool Checkable {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

ChildItemsTemplate Property

If specified, this template will be applied for all menu items instead of top level menu items.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.ChildItemsTemplate", Description="If
specified, this template will be applied for all menu items instead of top
level menu items.")>
<System.Web.UI.TemplateInstanceAttribute(TemplateInstance.Single)>
<C1CategoryAttribute("Behavior")>
<System.Web.UI.TemplateContainerAttribute(BindingDirection=BindingDirection.O
neWay,
ContainerType=C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer)>
<System.ComponentModel.BrowsableAttribute(False)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Content)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
Public Overridable Property ChildItemsTemplate As System.Web.UI.ITemplate
```

C#

```
[C1Description(Key="C1Menu.ChildItemsTemplate", Description="If specified,
this template will be applied for all menu items instead of top level menu
items.")]
[System.Web.UI.TemplateInstance(TemplateInstance.Single)]
[C1Category("Behavior")]
[System.Web.UI.TemplateContainer(BindingDirection=BindingDirection.OneWay,
ContainerType=C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer)]
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
```

```
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
public virtual System.Web.UI.ITemplate ChildItemsTemplate {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

CrumbDefaultText Property

Gets and sets the top breadcrumb's default text.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1Menu.CrumbDefaultText", Description="Gets and
sets the top breadcrumb's default text")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
Public Property CrumbDefaultText As System.String
```

C#

```
[C1Category("Appearance")]
[C1Description(Key="C1Menu.CrumbDefaultText", Description="Gets and sets the
top breadcrumb's default text")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
public System.string CrumbDefaultText {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

DataBindings Property

Visual Basic (Declaration)

```
<C1CategoryAttribute("Data")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Content)>
<C1DescriptionAttribute(Key="C1Menu.DataBindings", Description="Specifies whether the menu is context.")>
<System.ComponentModel.DefaultValueAttribute()>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
Public ReadOnly Property DataBindings As C1MenuItemBindingCollection
```

C#

```
[C1Category("Data")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Content)]
[C1Description(Key="C1Menu.DataBindings", Description="Specifies whether the menu is context.")]
[System.ComponentModel.DefaultValue()]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
public C1MenuItemBindingCollection DataBindings {get;}
```

Direction Property

A value that indicates menu's direction.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.Direction", Description="A value that indicates menu's direction.")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Direction As Direction
```

C#

```
[C1Description(Key="C1Menu.Direction", Description="A value that indicates menu's direction.")]
[WidgetOption()]
[C1Category("Appearance")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
public Direction Direction {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

EnsureSubmenuOnBody Property

A value indicating the submenu will be append to the body or menu container. If the value is true, the submenu will be appended to body element, else it will append to the menu container. Default: false. Type: Boolean.

Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1Menu.EnsureSubmenuOnBody", Description="A
value indicating the submenu will be append to the body or menu container.")>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
Public Property EnsureSubmenuOnBody As System.Boolean
```

C#

```
[WidgetOption()]
[C1Description(Key="C1Menu.EnsureSubmenuOnBody", Description="A value
indicating the submenu will be append to the body or menu container.")]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Behavior)]
public System.bool EnsureSubmenuOnBody {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

Height Property

Gets or sets the height of the menu.

Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Sizes)>
Public Overrides Property Height As System.Web.UI.WebControls.Unit
```

C#

```
[WidgetOption()]
[Layout(LayoutType.Sizes)]
public override System.Web.UI.WebControls.Unit Height {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

HideAnimation Property

Defines the animation to hide submenu in flyout mode.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.HideAnimation", Description="Defines the
animation to hide submenu in flyout mode")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.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")>
Public Property HideAnimation As Animation
```

C#

```
[C1Description(Key="C1Menu.HideAnimation", Description="Defines the animation
to hide submenu in flyout mode")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
```

```
[C1Category("Behavior")]
[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")]
public Animation HideAnimation {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

HideDelay Property

Gets or sets delay time to hide submenu.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1Menu.HideDelay", Description="Gets or sets
delay time to hide submenu.")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property HideDelay As System.Integer
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1Menu.HideDelay", Description="Gets or sets delay time
to hide submenu.")]
[WidgetOption()]
[C1Category("Behavior")]
[Layout(LayoutType.Behavior)]
public System.int HideDelay {get; set;}
```

See Also

Reference

[C1Menu Class](#)[C1Menu Members](#)

IsDirty Property

Gets or sets a value indicates whether this instance is dirty.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(False)>
<System.ComponentModel.DesignOnlyAttribute(True)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Hidden)>
Public Property IsDirty As System.Boolean
```

C#

```
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.DesignOnly(true)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Hidden)]
public System.bool IsDirty {get; set;}
```

See Also

Reference

[C1Menu Class](#)[C1Menu Members](#)

Items Property

Gets the menu items.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Content)>
<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1Menu.Items", Description="Gets the menu items.")>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
```



```
<System.ComponentModel.BrowsableAttribute(False)>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.Web.UI.PersistencyModeAttribute(PersistencyMode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<WidgetOptionAttribute()>
Public ReadOnly Property Items As C1MenuItemCollection
```

C#

```
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1Menu.Items", Description="Gets the menu items.")]
[C1Category("Behavior")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.Web.UI.PersistencyMode(PersistencyMode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[WidgetOption()]
public C1MenuItemCollection Items {get;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

ItemsTemplate Property

If specified, this template will be applied for all menu items that does not have other defined templates.

Syntax

Visual Basic (Declaration)

```
<System.Web.UI.PersistencyModeAttribute(PersistencyMode.InnerProperty)>
<C1DescriptionAttribute(Key="C1Menu.ItemsTemplate", Description="If
specified, this template will be applied for all menu items that does not
have other defined templates.")>
<C1CategoryAttribute("Behavior")>
```

```
<System.Web.UI.TemplateContainerAttribute(BindingDirection=BindingDirection.O
newWay,
ContainerType=C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer)>
<System.ComponentModel.BrowsableAttribute(False)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Content)>
<System.Web.UI.TemplateInstanceAttribute(TemplateInstance.Single)>
Public Overridable Property ItemsTemplate As System.Web.UI.ITemplate
```

C#

```
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[C1Description(Key="C1Menu.ItemsTemplate", Description="If specified, this
template will be applied for all menu items that does not have other defined
templates.")]
[C1Category("Behavior")]
[System.Web.UI.TemplateContainer(BindingDirection=BindingDirection.OneWay,
ContainerType=C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer)]
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.Web.UI.TemplateInstance(TemplateInstance.Single)]
public virtual System.Web.UI.ITemplate ItemsTemplate {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

MaxHeight Property

Determines the iPod-style menu's maximum height.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1Menu.MaxHeight", Description="Determines the
```

```
iPod-style menu's maximum height">
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Property MaxHeight As System.Integer
```

C#

```
[C1Category("Appearance")]
[C1Description(Key="C1Menu.MaxHeight", Description="Determines the iPod-style
menu's maximum height")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
public System.int MaxHeight {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

Mode Property

Defines the behavior of the submenu whether it is a popup menu or an iPod-style navigation list.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1Menu.Mode", Description="Defines the behavior
of the submenu whether it is a popup menu or an iPod-style navigation list")>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Mode As MenuMode
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Appearance")]
[WidgetOption()]
[C1Description(Key="C1Menu.Mode", Description="Defines the behavior of the
```

```
submenu whether it is a popup menu or an iPod-style navigation list"))]
[Layout(LayoutType.Appearance)]
public MenuMode Mode {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

OnClientBlur Property

Triggered when a menu item loses focus.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Client-Side Events")>
<C1DescriptionAttribute(Key="C1Menu.OnClientBlur", Description="Triggered
when a menu item loses focus")>
<WidgetEventAttribute("e")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionNameAttribute("blur")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property OnClientBlur As System.String
```

C#

```
[C1Category("Client-Side Events")]
[C1Description(Key="C1Menu.OnClientBlur", Description="Triggered when a menu
item loses focus")]
[WidgetEvent("e")]
[System.ComponentModel.DefaultValue()]
[WidgetOptionName("blur")]
[Layout(LayoutType.Behavior)]
public System.string OnClientBlur {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

OnClientFocus Property

Triggered when a menu item gets the focus, either when the mouse is used to hover over it (on hover) or when the cursor keys are used on the keyboard (navigation with cursor key) focus.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.OnClientFocus", Description="Triggered
when a menu item gets the focus")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionNameAttribute("focus")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property OnClientFocus As System.String
```

C#

```
[C1Description(Key="C1Menu.OnClientFocus", Description="Triggered when a menu
item gets the focus")]
[C1Category("Client-Side Events")]
[WidgetEvent("e")]
[System.ComponentModel.DefaultValue()]
[WidgetOptionName("focus")]
[Layout(LayoutType.Behavior)]
public System.string OnClientFocus {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

OnClientHidden Property

Triggered once a menu or submenu has hidden.

Syntax

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("hidden")>
<C1CategoryAttribute("Client-Side Events")>
```

```
<WidgetEventAttribute("e, item")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1Menu.OnClientHidden", Description="Triggered
once a menu or submenu has hidden.")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property OnClientHidden As System.String
```

C#

```
[WidgetOptionName("hidden")]
[C1Category("Client-Side Events")]
[WidgetEvent("e, item")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1Menu.OnClientHidden", Description="Triggered once a
menu or submenu has hidden.")]
[Layout(LayoutType.Behavior)]
public System.string OnClientHidden {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

OnClientHidding Property

Triggered before hidding a menu or submenu.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1Menu.OnClientHidding", Description="Triggered
before hidding a menu or submenu.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, item")>
<WidgetOptionNameAttribute("hidding")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property OnClientHidding As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1Menu.OnClientHidding", Description="Triggered before
```

```

hidding a menu or submenu."))
[C1Category("Client-Side Events")]
[WidgetEvent("e, item")]
[WidgetOptionName("hidding")]
[Layout(LayoutType.Behavior)]
public System.string OnClientHidding {get; set;}

```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

OnClientSelect Property

Triggered when a menu item is selected.

Syntax

Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1Menu.OnClientSelect", Description="Triggered
when a menu item is selected")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionNameAttribute("select")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property OnClientSelect As System.String

```

C#

```

[C1Description(Key="C1Menu.OnClientSelect", Description="Triggered when a
menu item is selected")]
[C1Category("Client-Side Events")]
[WidgetEvent("e")]
[System.ComponentModel.DefaultValue()]
[WidgetOptionName("select")]
[Layout(LayoutType.Behavior)]
public System.string OnClientSelect {get; set;}

```

See Also

Reference

[C1Menu Class](#)[C1Menu Members](#)

OnClientShowing Property

Triggered before showing a menu or submenu.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.OnClientShowing", Description="Triggered
before showing a menu or submenu.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, item")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionNameAttribute("showing")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property OnClientShowing As System.String
```

C#

```
[C1Description(Key="C1Menu.OnClientShowing", Description="Triggered before
showing a menu or submenu.")]
[C1Category("Client-Side Events")]
[WidgetEvent("e, item")]
[System.ComponentModel.DefaultValue()]
[WidgetOptionName("showing")]
[Layout(LayoutType.Behavior)]
public System.string OnClientShowing {get; set;}
```

See Also

Reference

[C1Menu Class](#)[C1Menu Members](#)

OnClientShown Property

Triggered once a menu or submenu has shown.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.OnClientShown", Description="Triggered
```



```

once a menu or submenu has shown.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("e, item")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionNameAttribute("shown")>
<LayoutAttribute(LayoutType.Behavior)>
Public Property OnClientShown As System.String

```

C#

```

[C1Description(Key="C1Menu.OnClientShown", Description="Triggered once a menu
or submenu has shown.")]
[C1Category("Client-Side Events")]
[WidgetEvent("e, item")]
[System.ComponentModel.DefaultValue()]
[WidgetOptionName("shown")]
[Layout(LayoutType.Behavior)]
public System.string OnClientShown {get; set;}

```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

Orientation Property

Controls the root menu's orientation. All submenus are vertical regardless of the orientation of the root menu.

Syntax

Visual Basic (Declaration)

```

<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1Menu.Orientation", Description="Controls the
root menu's orientation")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Orientation As Orientation

```

C#

```

[C1Category("Appearance")]

```

```
[WidgetOption()]
[C1Description(Key="C1Menu.Orientation", Description="Controls the root
menu's orientation")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
public Orientation Orientation {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

Owner Property

Owner of the menu. Always returns null.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(False)>
Public ReadOnly Property Owner As IC1MenuItemCollectionOwner
```

C#

```
[System.ComponentModel.Browsable(false)]
public IC1MenuItemCollectionOwner Owner {get;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

Position Property

Specifies the location and orientation of the menu relative to the button or link used to open it. Configuration for the Position Utility Of option is excluded; it is always configured by the widget. Collision also controls collision detection automatically.

Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
```

```
<C1DescriptionAttribute(Key="C1Menu.Position", Description="Specifies the
location and orientation of the menu relative to the button or link used to
open it")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Appearance")>
<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")>
Public Property Position As PositionSettings
```

C#

```
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1Menu.Position", Description="Specifies the location and
orientation of the menu relative to the button or link used to open it")]
[WidgetOption()]
[C1Category("Appearance")]
[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")]
public PositionSettings Position {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

ShowAnimation Property

Defines the show animation to show submenu in flyout mode.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Content)>
```

```
<C1DescriptionAttribute(Key="C1Menu.ShowAnimation", Description="Defines the
show animation to show submenu in flyout mode")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
<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.Behavior)>
Public Property ShowAnimation As Animation
```

C#

```
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[C1Description(Key="C1Menu.ShowAnimation", Description="Defines the show
animation to show submenu in flyout mode")]
[WidgetOption()]
[C1Category("Behavior")]
[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.Behavior)]
public Animation ShowAnimation {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

ShowDelay Property

Gets or sets delay time to show submenu.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.ShowDelay", Description="Gets or sets
delay time to show submenu.")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
```

```
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property ShowDelay As System.Integer
```

C#

```
[C1Description(Key="C1Menu.ShowDelay", Description="Gets or sets delay time
to show submenu.")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
public System.int ShowDelay {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

SlidingAnimation Property

Defines the sliding animation in slide mode.

Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1Menu.SlidingAnimation", Description="Defines
the sliding animation in slide mode")>
<C1CategoryAttribute("Behavior")>
<WidgetOptionAttribute()>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.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")>
Public Property SlidingAnimation As SlideAnimation
```

C#

```
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1Menu.SlidingAnimation", Description="Defines the
```

```

sliding animation in slide mode""))
[C1Category("Behavior")]
[WidgetOption()]
[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")]
public SlideAnimation SlidingAnimation {get; set;}

```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

SubmenuTriggerEvent Property

Specifies the event used to show the submenu.

Syntax

Visual Basic (Declaration)

```

<C1DescriptionAttribute(Key="C1Menu.SubmenuTriggerEvent",
Description="Specifies the event used to show the submenu.")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property SubmenuTriggerEvent As SubmenuTriggerEvent

```

C#

```

[C1Description(Key="C1Menu.SubmenuTriggerEvent", Description="Specifies the
event used to show the submenu.")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
public SubmenuTriggerEvent SubmenuTriggerEvent {get; set;}

```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

TopItemsTemplate Property

If specified, this template will be applied for top level menu items.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<System.Web.UI.TemplateInstanceAttribute(TemplateInstance.Single)>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<C1CategoryAttribute("Behavior")>
<System.Web.UI.TemplateContainerAttribute(BindingDirection=BindingDirection.O
neway,
ContainerType=C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer)>
<System.ComponentModel.BrowsableAttribute(False)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Content)>
<C1DescriptionAttribute(Key="C1Menu.TopItemsTemplate", Description="If
specified, this template will be applied for top level menu items.")>
Public Overridable Property TopItemsTemplate As System.Web.UI.ITemplate
```

C#

```
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[System.Web.UI.TemplateInstance(TemplateInstance.Single)]
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[C1Category("Behavior")]
[System.Web.UI.TemplateContainer(BindingDirection=BindingDirection.OneWay,
ContainerType=C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer)]
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[C1Description(Key="C1Menu.TopItemsTemplate", Description="If specified, this
template will be applied for top level menu items.")]
public virtual System.Web.UI.ITemplate TopItemsTemplate {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

TopLinkText Property

Gets and sets the text of the top link.

Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>  
<C1DescriptionAttribute(Key="C1Menu.TopLinkText", Description="Gets and sets  
the text of the top link")>  
<WidgetOptionAttribute()>  
<C1CategoryAttribute("Appearance")>  
<System.ComponentModel.DefaultValueAttribute()>  
Public Property TopLinkText As System.String
```

C#

```
[Layout(LayoutType.Appearance)]  
[C1Description(Key="C1Menu.TopLinkText", Description="Gets and sets the text  
of the top link")]  
[WidgetOption()]  
[C1Category("Appearance")]  
[System.ComponentModel.DefaultValue()]  
public System.string TopLinkText {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

Trigger Property

Specifies the event used to show the menu.

Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
```



```
<C1DescriptionAttribute(Key="C1Menu.Trigger", Description="Specifies the
event used to show the menu")>
<LayoutAttribute(LayoutType.Behavior)>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Behavior")>
Public Property Trigger As System.String
```

C#

```
[WidgetOption()]
[C1Description(Key="C1Menu.Trigger", Description="Specifies the event used to
show the menu")]
[Layout(LayoutType.Behavior)]
[System.ComponentModel.DefaultValue()]
[C1Category("Behavior")]
public System.string Trigger {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

TriggerEvent Property

Set the event used to show the menu and submenu(If "SubmenuTriggerEvent" is "Default").

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Menu.TriggerEvent", Description="Specifies the
event used to show the menu")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property TriggerEvent As TriggerEvent
```

C#

```
[C1Description(Key="C1Menu.TriggerEvent", Description="Specifies the event
used to show the menu")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
```

```
[C1Category("Behavior")]  
[System.ComponentModel.DefaultValue()]  
public TriggerEvent TriggerEvent {get; set;}
```

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

Width Property

Gets or sets the width of the menu.

Syntax

Visual Basic (Declaration)	
<pre><WidgetOptionAttribute()> <LayoutAttribute(LayoutType.Sizes)> Public Overrides Property Width As System.Web.UI.WebControls.Unit</pre>	
C#	
<pre>[WidgetOption()] [Layout(LayoutType.Sizes)] public override System.Web.UI.WebControls.Unit Width {get; set;}</pre>	

See Also

Reference


[C1Menu Class](#)








[C1Menu Members](#)

Events

For a list of all members of this type, see [C1Menu members](#).

Public Events

	Name	Description
	DataBinding	(Inherited from System.Web.UI.Control)

	DataBound	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	Disposed	(Inherited from System.Web.UI.Control)
	Init	(Inherited from System.Web.UI.Control)
	ItemClick	Fires when an item is clicked
	Load	(Inherited from System.Web.UI.Control)
	PreRender	(Inherited from System.Web.UI.Control)
	Unload	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1Menu Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

ItemClick Event

Fires when an item is clicked

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(True)>
<C1CategoryAttribute("Events")>
<C1DescriptionAttribute(Key="C1Menu.ItemClick", Description="Fires when an
item is clicked.")>
Public Event ItemClick As C1MenuEventHandler
```

C#

```
[System.ComponentModel.Browsable(true)]
[C1Category("Events")]
[C1Description(Key="C1Menu.ItemClick", Description="Fires when an item is
clicked.")]
public event C1MenuEventHandler ItemClick
```

Event Data

The event handler receives an argument of type [C1MenuEventArgs](#) containing data related to this event. The following **C1MenuEventArgs** properties provide information specific to this event.

Property	Description
CommandArgument	(Inherited from System.Web.UI.WebControls.CommandEventArgs)
CommandName	(Inherited from System.Web.UI.WebControls.CommandEventArgs)
CommandSource	Gets the Object that raised the event.
Item	Gets the menu item that raised the event.

See Also

Reference

[C1Menu Class](#)

[C1Menu Members](#)

C1MenuEventArgs

C1MenuEventArgs is used by the ItemClick event.

Object Model

C1MenuEventArgs

Syntax

Visual Basic (Declaration)	
<pre>Public NotInheritable Class C1MenuEventArgs Inherits System.Web.UI.WebControls.CommandEventArgs</pre>	
C#	
<pre>public sealed class C1MenuEventArgs : System.Web.UI.WebControls.CommandEventArgs</pre>	

Inheritance Hierarchy

System.Object
 System.EventArgs
 System.Web.UI.WebControls.CommandEventArgs
 C1.Web.Wijmo.Controls.C1Menu.C1MenuEventArgs

See Also

Reference

[C1MenuEventArgs Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

C1MenuEventArgs is used by the ItemClick event.

Object Model

C1MenuEventArgs

Syntax

Visual Basic (Declaration)	
Public NotInheritable Class C1MenuEventArgs Inherits System.Web.UI.WebControls.CommandEventArgs	
C#	
public sealed class C1MenuEventArgs : System.Web.UI.WebControls.CommandEventArgs	

Inheritance Hierarchy

System.Object
 System.EventArgs
 System.Web.UI.WebControls.CommandEventArgs
 C1.Web.Wijmo.Controls.C1Menu.C1MenuEventArgs

See Also

Reference


[C1MenuEventArgs Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members





The following tables list the members exposed by [C1MenuEventArgs](#).

Public Constructors

	Name	Description
	C1MenuEventArgs Constructor	Overloaded.

[Top](#)

Public Properties

	Name	Description
	CommandArgument	(Inherited from System.Web.UI.WebControls.CommandEventArgs)
	CommandName	(Inherited from System.Web.UI.WebControls.CommandEventArgs)
	CommandSource	Gets the Object that raised the event.
	Item	Gets the menu item that raised the event.

[Top](#)

See Also

Reference

[C1MenuEventArgs Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1MenuEventArgs Constructor

Overload List

Overload	Description
C1MenuEventArgs Constructor(C1MenuItem)	Initializes a new instance of the C1MenuEventArgs class.

C1MenuEventArgs Constructor(C1MenuItem, Object, CommandEventArgs)	Initializes a new instance of the C1MenuEventArgs class..
---	---

See Also

Reference

[C1MenuEventArgs Class](#)

[C1MenuEventArgs Members](#)

C1MenuEventArgs Constructor(C1MenuItem)

Initializes a new instance of the [C1MenuEventArgs](#) class.

Syntax

Visual Basic (Declaration)	
<pre>Public Function New(_ ByVal item As C1MenuItem _)</pre>	
C#	
<pre>public C1MenuEventArgs(C1MenuItem item)</pre>	

Parameters

item

Menu item that raised the event.

See Also

Reference

[C1MenuEventArgs Class](#)

[C1MenuEventArgs Members](#)

[Overload List](#)

C1MenuEventArgs

Constructor(C1MenuItem, Object, CommandEventArgs)

Menu item that raised the event.

Object that raised the event.

A `System.Web.UI.WebControls.CommandEventArgs` that contains the event data.

Initializes a new instance of the [C1MenuEventArgs](#) class..

Syntax

Visual Basic (Declaration)

```
Public Function New( _  
    ByVal item As C1MenuItem, _  
    ByVal commandSource As System.Object, _  
    ByVal originalArgs As System.Web.UI.WebControls.CommandEventArgs _  
)
```

C#

```
public C1MenuEventArgs(  
    C1MenuItem item,  
    System.Object commandSource,  
    System.Web.UI.WebControls.CommandEventArgs originalArgs  
)
```

Parameters

item

Menu item that raised the event.

commandSource

Object that raised the event.

originalArgs

A `System.Web.UI.WebControls.CommandEventArgs` that contains the event data.

See Also

Reference

[C1MenuEventArgs Class](#)





[C1MenuEventArgs Members](#)

[Overload List](#)

Properties

For a list of all members of this type, see [C1MenuEventArgs members](#).

Public Properties

	Name	Description
	CommandArgument	(Inherited from System.Web.UI.WebControls.CommandEventArgs)
	CommandName	(Inherited from System.Web.UI.WebControls.CommandEventArgs)
	CommandSource	Gets the Object that raised the event.
	Item	Gets the menu item that raised the event.

[Top](#)

See Also

Reference

[C1MenuEventArgs Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

CommandSource Property

Gets the Object that raised the event.

Syntax

Visual Basic (Declaration)	
Public ReadOnly Property CommandSource As System.Object	
C#	
public System.object CommandSource {get;}	

See Also

Reference

[C1MenuEventArgs Class](#)

[C1MenuEventArgs Members](#)

Item Property

Gets the menu item that raised the event.

Syntax

Visual Basic (Declaration)	
Public ReadOnly Property Item As C1MenuItem	
C#	
public C1MenuItem Item {get;}	

See Also

Reference

[C1MenuEventArgs Class](#)

[C1MenuEventArgs Members](#)

C1MenuItem

Object Model

C1MenuItem

Syntax

Visual Basic (Declaration)	
<pre><System.Web.UI.ParseChildrenAttribute(True)> <System.ComponentModel.ToolboxItemAttribute(ToolboxItemType=, ToolboxItemTypeName="")> Public Class C1MenuItem Inherits C1.Web.Wijmo.Controls.Base.UIElement Implements IC1MenuItemCollectionOwner</pre>	
C#	
<pre>[System.Web.UI.ParseChildren(true)] [System.ComponentModel.ToolboxItem(ToolboxItemType=, ToolboxItemTypeName="")] public class C1MenuItem : C1.Web.Wijmo.Controls.Base.UIElement, IC1MenuItemCollectionOwner</pre>	

Inheritance Hierarchy

System.Object

System.Web.UI.Control

System.Web.UI.WebControls.WebControl

C1.Web.Wijmo.Controls.C1Menu.C1MenuItem

See Also

Reference

[C1MenuItem Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

Object Model

C1MenuItem

Syntax

Visual Basic (Declaration)

```
<System.Web.UI.ParseChildrenAttribute(True)>
<System.ComponentModel.ToolboxItemAttribute(ToolboxItemType=,
ToolboxItemTypeName="")>
Public Class C1MenuItem
    Inherits C1.Web.Wijmo.Controls.Base.UIElement
    Implements IC1MenuItemCollectionOwner
```

C#

```
[System.Web.UI.ParseChildren(true)]
[System.ComponentModel.ToolboxItem(ToolboxItemType=, ToolboxItemTypeName="")]
public class C1MenuItem : C1.Web.Wijmo.Controls.Base.UIElement,
IC1MenuItemCollectionOwner
```

Inheritance Hierarchy

System.Object

System.Web.UI.Control

System.Web.UI.WebControls.WebControl

C1.Web.Wijmo.Controls.C1Menu.C1MenuItem

See Also

Reference

[C1MenuItem Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members











The following tables list the members exposed by [C1MenuItem](#).
















Public Constructors















	Name	Description
	C1MenuItem Constructor	

[Top](#)

Public Properties

	Name	Description
	AccessKey	(Inherited from System.Web.UI.WebControls.WebControl)
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from System.Web.UI.WebControls.WebControl)
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	Controls	(Inherited from System.Web.UI.Control)
	ControlStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ControlStyleCreated	(Inherited from System.Web.UI.WebControls.WebControl)
	Enabled	Overridden.
	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)

	Header	A value that determines whether the menu item is a header item.
	IconClass	Gets or sets the icon css class to the menuitem that is displayed next to the text in the menu item.
	ID	(Inherited from System.Web.UI.Control)
	ImagePosition	Gets or sets which side, left or right, the image will be rendered from the menu item.
	Items	Gets a C1MenuItemCollection object that includes the submenu items of the current menu item.
	NamingContainer	(Inherited from System.Web.UI.Control)
	NavigateUrl	The menu's navigate url.
	Owner	Gets the owner object which contains the item. It might be an instance of the C1Menu class or the C1MenuItem class depending on the current hierarchy level.
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	Selected	Gets or sets the selected of the menu item.
	Separator	A value that determines whether the item is a separator.
	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)
	Target	Gets or sets the target window or frame in which to display the Web page content associated with a menu item when the menu item is clicked or the enter key is pressed.
	Template	Menu item template.
	TemplateContainer	Gets or sets the template container.
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Text	The menu item's text.
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	UniqueID	(Inherited from System.Web.UI.Control)
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	Value	Gets or sets the value of the menu item.
	ViewStateMode	(Inherited from System.Web.UI.Control)

[Top](#)







Public Methods

	Name	Description
≡	ApplyStyle	(Inherited from System.Web.UI.WebControls.WebControl)
≡	ApplyStyleSheetSkin	(Inherited from System.Web.UI.Control)
≡	C1DeserializeItems	Avoid to deserialize the items property.
≡	CopyBaseAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
≡	DataBind	(Inherited from System.Web.UI.Control)
≡	Dispose	(Inherited from System.Web.UI.Control)
≡	FindControl	(Inherited from System.Web.UI.Control)
≡	Focus	(Inherited from System.Web.UI.Control)
≡	GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
≡	GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
≡	HasControls	(Inherited from System.Web.UI.Control)
≡	MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
≡	RenderBeginTag	Overridden. Renders the HTML opening tag of the control to the specified writer. This method is used primarily by control developers.
≡	RenderControl	(Inherited from System.Web.UI.Control)
≡	RenderEndTag	(Inherited from System.Web.UI.WebControls.WebControl)
≡	ResolveClientUrl	(Inherited from System.Web.UI.Control)
≡	ResolveUrl	(Inherited from System.Web.UI.Control)

	SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)
---	---	--

[Top](#)

Public Events

	Name	Description
	DataBinding	(Inherited from System.Web.UI.Control)
	Disposed	(Inherited from System.Web.UI.Control)
	Init	(Inherited from System.Web.UI.Control)
	Load	(Inherited from System.Web.UI.Control)
	PreRender	(Inherited from System.Web.UI.Control)
	Unload	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1MenuItem Class](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1MenuItem Constructor

Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1MenuItem()	

See Also

Reference





[C1MenuItem Class](#)[C1MenuItem Members](#)

Methods

For a list of all members of this type, see [C1MenuItem members](#).

Public Methods

	Name	Description
⇒	ApplyStyle	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	ApplyStyleSheetSkin	(Inherited from System.Web.UI.Control)
⇒	C1DeserializeItems	Avoid to deserialize the items property.
⇒	CopyBaseAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	DataBind	(Inherited from System.Web.UI.Control)
⇒	Dispose	(Inherited from System.Web.UI.Control)
⇒	FindControl	(Inherited from System.Web.UI.Control)
⇒	Focus	(Inherited from System.Web.UI.Control)
⇒	GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
⇒	GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
⇒	HasControls	(Inherited from System.Web.UI.Control)
⇒	MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	RenderBeginTag	Overridden. Renders the HTML opening tag of the control to the specified writer. This method is used primarily by control developers.
⇒	RenderControl	(Inherited from System.Web.UI.Control)

	RenderEndTag	(Inherited from System.Web.UI.WebControls.WebControl)
	ResolveClientUrl	(Inherited from System.Web.UI.Control)
	ResolveUrl	(Inherited from System.Web.UI.Control)
	SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1MenuItem Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1DeserializeItems Method

object to deserialize.

Avoid to deserialize the items property.

Syntax

Visual Basic (Declaration)	
<pre>Public Sub C1DeserializeItems(_ ByVal obj As System.Object _)</pre>	
C#	
<pre>public void C1DeserializeItems(System.object obj)</pre>	

Parameters

obj

object to deserialize.

See Also

Reference

[C1MenuItem Class](#)[C1MenuItem Members](#)

RenderBeginTag Method

A *System.Web.UI.HtmlTextWriter* that represents the output stream to render HTML content on the client.

Renders the HTML opening tag of the control to the specified writer. This method is used primarily by control developers.

Syntax

Visual Basic (Declaration)

```
Public Overrides Sub RenderBeginTag( _  
    ByVal writer As System.Web.UI.HtmlTextWriter _  
)
```

C#

```
public override void RenderBeginTag(  
    System.Web.UI.HtmlTextWriter writer  
)
```

Parameters

writer

A *System.Web.UI.HtmlTextWriter* that represents the output stream to render HTML content on the client.

See Also


Reference















[C1MenuItem Class](#)[C1MenuItem Members](#)















Properties











For a list of all members of this type, see [C1MenuItem members](#).

Public Properties

	Name	Description
	AccessKey	(Inherited from

		System.Web.UI.WebControls.WebControl)
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from System.Web.UI.WebControls.WebControl)
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	Controls	(Inherited from System.Web.UI.Control)
	ControlStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ControlStyleCreated	(Inherited from System.Web.UI.WebControls.WebControl)
	Enabled	Overridden.
	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	Header	A value that determines whether the menu item is a header item.
	IconClass	Gets or sets the icon css class to the menuitem that is displayed next to the text in the menu item.
	ID	(Inherited from System.Web.UI.Control)
	ImagePosition	Gets or sets which side, left or right, the image will be rendered from the menu item.
	Items	Gets a C1MenuItemCollection object that includes the submenu items of the current menu item.

	NamingContainer	(Inherited from System.Web.UI.Control)
	NavigateUrl	The menu's navigate url.
	Owner	Gets the owner object which contains the item. It might be an instance of the C1Menu class or the C1MenuItem class depending on the current hierarchy level.
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	Selected	Gets or sets the selected of the menu item.
	Separator	A value that determines whether the item is a separator.
	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)
	Target	Gets or sets the target window or frame in which to display the Web page content associated with a menu item when the menu item is clicked or the

		enter key is pressed.
	Template	Menu item template.
	TemplateContainer	Gets or sets the template container.
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Text	The menu item's text.
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	UniqueID	(Inherited from System.Web.UI.Control)
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	Value	Gets or sets the value of the menu item.
	ViewStateMode	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1MenuItem Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Enabled Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Appearance")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Overrides Property Enabled As System.Boolean
```

C#

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

```
[C1Category("Appearance")]
[WidgetOption()]
[Layout(LayoutType.Appearance)]
public override System.bool Enabled {get; set;}
```

Header Property

A value that determines whether the menu item is a header item.

Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Behavior")>
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1MenuItem.Header", Description="A value that
determines whether the menu item is a header item.")>
Public Property Header As System.Boolean
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[C1Category("Behavior")]
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1MenuItem.Header", Description="A value that determines
whether the menu item is a header item.")]
public System.bool Header {get; set;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

IconClass Property

Gets or sets the icon css class to the menuitem that is displayed next to the text in the menu item.

Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1MenuItem.IconClass", Description="Gets or sets
the icon css class to the menuitem that is displayed next to the text in the
menu item.")>
```

```
Public Property IconClass As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[C1Category("Appearance")]
[C1Description(Key="C1MenuItem.IconClass", Description="Gets or sets the icon
css class to the menuitem that is displayed next to the text in the menu
item.")]
public System.string IconClass {get; set;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

ImagePosition Property

Gets or sets which side, left or right, the image will be rendered from the menu item.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1MenuItem.ImagePosition", Description="Gets or
sets which side, left or right, the image will be rendered from the menu
item.")>
Public Property ImagePosition As ImagePosition
```

C#

```
[C1Category("Appearance")]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1MenuItem.ImagePosition", Description="Gets or sets
which side, left or right, the image will be rendered from the menu item.")]
```



```
public ImagePosition ImagePosition {get; set;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

Items Property

Gets a C1MenuItemCollection object that includes the submenu items of the current menu item.

Syntax

Visual Basic (Declaration)	
<pre><System.ComponentModel.NotifyParentPropertyAttribute(True)> <LayoutAttribute(LayoutType.Behavior)> <System.ComponentModel.BrowsableAttribute(False)> <C1DescriptionAttribute(Key="C1MenuItemBase.Items", Description="C1MenuItemBase.Items")> <System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)> <System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Content)> <System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)> <System.ComponentModel.MergablePropertyAttribute(False)> <C1CategoryAttribute("Default")> <WidgetOptionAttribute()> Public Overridable ReadOnly Property Items As C1MenuItemCollection</pre>	
C#	
<pre>[System.ComponentModel.NotifyParentProperty(true)] [Layout(LayoutType.Behavior)] [System.ComponentModel.Browsable(false)] [C1Description(Key="C1MenuItemBase.Items", Description="C1MenuItemBase.Items")] [System.ComponentModel.RefreshProperties(RefreshProperties.All)] [System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Content)] [System.Web.UI.Persistencemode(Persistencemode.InnerProperty)] [System.ComponentModel.MergableProperty(false)] [C1Category("Default")]</pre>	

```
[WidgetOption()]
public virtual C1MenuItemCollection Items {get;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

NavigateUrl Property

The menu's navigate url.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute(>
<C1DescriptionAttribute(Key="C1MenuItem.NavigateUrl", Description="The menu's
navigate url.")>
<C1CategoryAttribute("Behavior")>
<LayoutAttribute(LayoutType.Behavior)>
<System.Web.UI.UrlPropertyAttribute(>
<WidgetOptionAttribute(>
Public Property NavigateUrl As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1MenuItem.NavigateUrl", Description="The menu's navigate
url.")]
[C1Category("Behavior")]
[Layout(LayoutType.Behavior)]
[System.Web.UI.UrlProperty()]
[WidgetOption()]
public System.string NavigateUrl {get; set;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

Owner Property

Gets the owner object which contains the item. It might be an instance of the C1Menu class or the C1MenuItem class depending on the current hierarchy level.

Syntax

Visual Basic (Declaration)	
<pre><System.ComponentModel.BrowsableAttribute(False)> Public ReadOnly Property Owner As IC1MenuItemCollectionOwner</pre>	
C#	
<pre>[System.ComponentModel.Browsable(false)] public IC1MenuItemCollectionOwner Owner {get;}</pre>	

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

Selected Property

Gets or sets the selected of the menu item.

Syntax

Visual Basic (Declaration)	
<pre><C1DescriptionAttribute(Key="C1MenuItem.Selected", Description="A value that indicates whether the menu item is selected")> <System.ComponentModel.DefaultValueAttribute()> <C1CategoryAttribute("Behavior")> <LayoutAttribute(LayoutType.Behavior)> <WidgetOptionAttribute()> Public Property Selected As System.Boolean</pre>	
C#	
<pre>[C1Description(Key="C1MenuItem.Selected", Description="A value that indicates whether the menu item is selected")] [System.ComponentModel.DefaultValue()] [C1Category("Behavior")] [Layout(LayoutType.Behavior)]</pre>	

```
[WidgetOption()]
public System.bool Selected {get; set;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

Separator Property

A value that determines whether the item is a separator.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1MenuItem.Separator", Description="A value that
determines whether the item is a separator.")>
Public Property Separator As System.Boolean
```

C#

```
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1MenuItem.Separator", Description="A value that
determines whether the item is a separator.")]
public System.bool Separator {get; set;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

Target Property

Gets or sets the target window or frame in which to display the Web page content associated with a menu item when the menu item is clicked or the enter key is pressed.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1MenuItem.Target", Description="Gets or sets
the target window or frame in which to display the Web page content
associated with a menu item when the menu item is clicked or the enter key is
pressed.")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
Public Overridable Property Target As System.String
```

C#

```
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1MenuItem.Target", Description="Gets or sets the target
window or frame in which to display the Web page content associated with a
menu item when the menu item is clicked or the enter key is pressed.")]
[WidgetOption()]
[Layout(LayoutType.Behavior)]
public virtual System.string Target {get; set;}
```

Remarks

The default value is an empty string.

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

Template Property

Menu item template.

Syntax

Visual Basic (Declaration)

```
<System.Web.UI.TemplateContainerAttribute(BindingDirection=BindingDirection.O
```

```

newWay,
ContainerType=C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer)>
<C1DescriptionAttribute(Key="C1MenuItem.Template", Description="Gets or sets
the template that will be used for this menu item.")>
<System.ComponentModel.BrowsableAttribute(False)>
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Content)>
<System.Web.UI.PersistencyModeAttribute(PersistencyMode.InnerProperty)>
<System.Web.UI.TemplateInstanceAttribute(TemplateInstance.Single)>
<C1CategoryAttribute("Behavior")>
Public Property Template As System.Web.UI.ITemplate

```

C#

```

[System.Web.UI.TemplateContainer(BindingDirection=BindingDirection.OneWay,
ContainerType=C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer)]
[C1Description(Key="C1MenuItem.Template", Description="Gets or sets the
template that will be used for this menu item.")]
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.Bindable(Bindable=false,
Direction=BindingDirection.OneWay)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.Web.UI.PersistencyMode(PersistencyMode.InnerProperty)]
[System.Web.UI.TemplateInstance(TemplateInstance.Single)]
[C1Category("Behavior")]
public System.Web.UI.ITemplate Template {get; set;}

```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

TemplateContainer Property

Gets or sets the template container.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(False)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Hidden)>
Public Property TemplateContainer As C1MenuItemTemplateContainer
```

C#

```
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Hidden)]
public C1MenuItemTemplateContainer TemplateContainer {get; set;}
```

Property Value

The template container.

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

Text Property

The menu item's text.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Appearance")>
<C1DescriptionAttribute(Key="C1MenuItem.Text", Description="Gets or sets the text caption for the menu item.")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute()>
Public Property Text As System.String
```

C#

```
[C1Category("Appearance")]
[C1Description(Key="C1MenuItem.Text", Description="Gets or sets the text caption for the menu item.")]
[WidgetOption()]
[Layout(LayoutType.Appearance)]
```

```
[System.ComponentModel.DefaultValue()]
public System.string Text {get; set;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

Value Property

Gets or sets the value of the menu item.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1MenuItem.Value", Description="Gets or sets the
value of the menu item.")>
<C1CategoryAttribute("Behavior")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
Public Property Value As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1MenuItem.Value", Description="Gets or sets the value of
the menu item.")]
[C1Category("Behavior")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
public System.string Value {get; set;}
```

See Also

Reference

[C1MenuItem Class](#)

[C1MenuItem Members](#)

C1MenuItemBinding

Object Model

C1MenuItemBinding

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultPropertyAttribute("TextField")>  
Public NotInheritable Class C1MenuItemBinding
```

C#

```
[System.ComponentModel.DefaultProperty("TextField")]  
public sealed class C1MenuItemBinding
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemBinding

See Also

Reference

[C1MenuItemBinding Members](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

Object Model

C1MenuItemBinding

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultPropertyAttribute("TextField")>  
Public NotInheritable Class C1MenuItemBinding
```

C#

```
[System.ComponentModel.DefaultProperty("TextField")]  
public sealed class C1MenuItemBinding
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemBinding

See Also

Reference


[C1MenuItemBinding Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members







The following tables list the members exposed by [C1MenuItemBinding](#).




Public Constructors

	Name	Description
	C1MenuItemBinding Constructor	Overloaded.

[Top](#)

Public Properties

	Name	Description
	DataMember	Gets or sets the data member to bind a menu item.
	Depth	Gets or sets the menu depth to which the C1MenuItemBinding object is applied.
	FormatString	Gets or sets the string that specifies the display format for the text of a C1MenuItem to which the C1MenuItemBinding object is applied
	HeaderField	Gets or sets the name of the field from the data source that indicates C1Menu header item.
	ImageUrlField	Gets or sets the name of the field from the data source to bind the imageUrl property of a C1MenuItem object to which the C1MenuBinding object is applied.
	NavigateUrlField	Gets or sets the name of the field from the data source to bind to the NavigateUrl property of a C1MenuItem object to which the

		C1MenuBinding object is applied.
	SeparatorField	Gets or sets the name of the field from the data source that indicates C1Menu separator item.
	TextField	Gets or sets the name of the field from the data source to bind to the Text property of a C1menuItem object to which the C1MenuBinding object is applied.
	ValueField	Gets or sets the name of the field from the data source to bind the value property of a C1MenuItem object to which the C1MenuBinding object is applied.

[Top](#)

See Also

Reference

[C1MenuItemBinding Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1MenuItemBinding Constructor

Overload List

Overload	Description
C1MenuItemBinding Constructor()	Create new instance of the C1MenuItemBinding class.
C1MenuItemBinding Constructor(String)	Create new instance of the C1MenuItemBinding class.

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

C1MenuItemBinding Constructor()

Create new instance of the C1MenuItemBinding class.

Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1MenuItemBinding()	

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

[Overload List](#)

C1MenuItemBinding Constructor(String)

Create new instance of the C1MenuItemBinding class.

Syntax

Visual Basic (Declaration)	
Public Function New(_ ByVal <i>dataMember</i> As System.String _)	
C#	
public C1MenuItemBinding(System.string <i>dataMember</i>)	

Parameters

dataMember

Data member to bind a menu item.

See Also

Reference

[C1MenuItemBinding Class](#)






[C1MenuItemBinding Members](#)

[Overload List](#)

Properties

For a list of all members of this type, see [C1MenuItemBinding members](#).

Public Properties

	Name	Description
	DataMember	Gets or sets the data member to bind a menu item.
	Depth	Gets or sets the menu depth to which the C1MenuItemBinding object is applied.
	FormatString	Gets or sets the string that specifies the display format for the text of a C1MenuItem to which the C1MenuItemBinding object is applied
	HeaderField	Gets or sets the name of the field from the data source that indicates C1Menu header item.
	ImageUrlField	Gets or sets the name of the field from the data source to bind the imageUrl property of a C1MenuItem object to which the C1MenuBinding object is applied.
	NavigateUrlField	Gets or sets the name of the field from the data source to bind to the NavigateUrl property of a C1MenuItem object to which the C1MenuBinding object is applied.
	SeparatorField	Gets or sets the name of the field from the data source that indicates C1Menu separator item.
	TextField	Gets or sets the name of the field from the data source to bind to the Text property of a C1menuItem object to which the C1MenuBinding object is applied.
	ValueField	Gets or sets the name of the field from the data source to bind the value property of a C1MenuItem object to which the C1MenuBinding object is applied.

[Top](#)

See Also

Reference

[C1MenuItemBinding Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

DataMember Property

Gets or sets the data member to bind a menu item.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Data")>
<C1DescriptionAttribute(Key="C1MenuItemBinding.DataMember", Description="Gets
or sets the data member to bind a menu item.")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property DataMember As System.String
```

C#

```
[C1Category("Data")]
[C1Description(Key="C1MenuItemBinding.DataMember", Description="Gets or sets
the data member to bind a menu item.")]
[System.ComponentModel.DefaultValue()]
public System.string DataMember {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

Depth Property

Gets or sets the menu depth to which the C1MenuItemBinding object is applied.

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Data")>
<C1DescriptionAttribute(Key="C1MenuItemBinding.Depth", Description="Gets or
```

```
sets the menu depth to which the C1MenuItemBinding object is applied.")>
Public Property Depth As System.Integer
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Data")]
[C1Description(Key="C1MenuItemBinding.Depth", Description="Gets or sets the
menu depth to which the C1MenuItemBinding object is applied.")]
public System.int Depth {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

FormatString Property

Gets or sets the string that specifies the display format for the text of a C1MenuItem to which the C1MenuItemBinding object is applied

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1MenuItemBinding.FormatString",
Description="Gets or sets the string that specifies the display format for
the text of a C1MenuItem to which the C1MenuItemBinding object is applied")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.LocalizableAttribute(True)>
<C1CategoryAttribute("Data")>
Public Property FormatString As System.String
```

C#

```
[C1Description(Key="C1MenuItemBinding.FormatString", Description="Gets or
sets the string that specifies the display format for the text of a
C1MenuItem to which the C1MenuItemBinding object is applied")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.Localizable(true)]
[C1Category("Data")]
public System.string FormatString {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

HeaderField Property

Gets or sets the name of the field from the data source that indicates C1Menu header item.

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Databindings")>
<System.ComponentModel.DefaultValueAttribute()>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<C1DescriptionAttribute(Key="C1MenuItemBinding.HeaderField",
Description="Gets or sets the name of the field from the data source that
indicates C1Menu header item.")>
Public Property HeaderField As System.String
```

C#

```
[C1Category("Databindings")]
[System.ComponentModel.DefaultValue()]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[C1Description(Key="C1MenuItemBinding.HeaderField", Description="Gets or sets
the name of the field from the data source that indicates C1Menu header
item.")]
public System.string HeaderField {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

ImageUrlField Property

Gets or sets the name of the field from the data source to bind the imageUrl property of a C1MenuItem object to which the C1MenuBinding object is applied.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1MenuItemBinding.ImageUrlField",
Description="Gets or sets the name of the field from the data source to bind
the imageUrl property of a C1MenuItem object to which the C1MenuBinding
object is applied")>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<C1CategoryAttribute("Databindings")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property ImageUrlField As System.String
```

C#

```
[C1Description(Key="C1MenuItemBinding.ImageUrlField", Description="Gets or
sets the name of the field from the data source to bind the imageUrl property
of a C1MenuItem object to which the C1MenuBinding object is applied")]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[C1Category("Databindings")]
[System.ComponentModel.DefaultValue()]
public System.string ImageUrlField {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

NavigateUrlField Property

Gets or sets the name of the field from the data source to bind to the NavigateUrl property of a C1MenuItem object to which the C1MenuBinding object is applied.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1MenuItemBinding.NavigateUrlField",
Description="Gets or sets the name of the field from the data source to bind
```

```
to the NavigateUrl property of a C1MenuItem object to which the C1MenuBinding
object is applied.")>
<C1CategoryAttribute("Databindings")>
<System.ComponentModel.DefaultValueAttribute(>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
Public Property NavigateUrlField As System.String
```

C#

```
[C1Description(Key="C1MenuItemBinding.NavigateUrlField", Description="Gets or
sets the name of the field from the data source to bind to the NavigateUrl
property of a C1MenuItem object to which the C1MenuBinding object is
applied.")]
[C1Category("Databindings")]
[System.ComponentModel.DefaultValue()]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
public System.string NavigateUrlField {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

SeparatorField Property

Gets or sets the name of the field from the data source that indicates C1Menu separator item.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1MenuItemBinding.SeparatorField",
Description="Gets or sets the name of the field from the data source that
indicates C1Menu separator item.")>
<System.ComponentModel.DefaultValueAttribute(>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<C1CategoryAttribute("Databindings")>
```

```
Public Property SeparatorField As System.String
```

```
C#
```

```
[C1Description(Key="C1MenuItemBinding.SeparatorField", Description="Gets or
sets the name of the field from the data source that indicates C1Menu
separator item.")]
[System.ComponentModel.DefaultValue()]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[C1Category("Databindings")]
public System.string SeparatorField {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

TextField Property

Gets or sets the name of the field from the data source to bind to the Text property of a C1menuItem object to which the C1MenuBinding object is applied.

Syntax

```
Visual Basic (Declaration)
```

```
<C1DescriptionAttribute(Key="C1MenuItemBinding.TextField", Description="Gets
or sets the name of the field from the data source to bind to the Text
property of a C1menuItem object to which the C1MenuBinding object is
applied")>
<C1CategoryAttribute("Databindings")>
<System.ComponentModel.DefaultValueAttribute()>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
Public Property TextField As System.String
```

```
C#
```

```
[C1Description(Key="C1MenuItemBinding.TextField", Description="Gets or sets
the name of the field from the data source to bind to the Text property of a
C1menuItem object to which the C1MenuBinding object is applied")]
```

```
[C1Category("Databindings")]
[System.ComponentModel.DefaultValue()]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
public System.string TextField {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

ValueField Property

Gets or sets the name of the field from the data source to bind the value property of a C1MenuItem object to which the C1MenuBinding object is applied.

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1MenuItemBinding.ValueField", Description="Gets or sets the name of the field from the data source to bind the value property of a C1MenuItem object to which the C1MenuBinding object is applied")>
<C1CategoryAttribute("Databindings")>
<System.ComponentModel.DefaultValueAttribute()>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
Public Property ValueField As System.String
```

C#

```
[C1Description(Key="C1MenuItemBinding.ValueField", Description="Gets or sets the name of the field from the data source to bind the value property of a C1MenuItem object to which the C1MenuBinding object is applied")]
[C1Category("Databindings")]
[System.ComponentModel.DefaultValue()]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
public System.string ValueField {get; set;}
```

See Also

Reference

[C1MenuItemBinding Class](#)

[C1MenuItemBinding Members](#)

C1MenuItemBindingCollection

Represents a collection of C1MenuItemBinding objects that are used by data-bound C1Menu control.

Object Model

C1MenuItemBindingCollection

Syntax

Visual Basic (Declaration)

```
Public Class C1MenuItemBindingCollection
    Inherits
    C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection(Of
    C1Menu,C1MenuItemBinding)
```

C#

```
public class C1MenuItemBindingCollection :
    C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection<C1Menu,C1MenuItemBinding>
```

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.Collection<T>

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemBindingCollection

See Also

Reference

[C1MenuItemBindingCollection Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

Represents a collection of C1MenuItemBinding objects that are used by data-bound C1Menu control.

Object Model

C1MenuItemBindingCollection

Syntax

Visual Basic (Declaration)

```
Public Class C1MenuItemBindingCollection
    Inherits
    C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection(Of
    C1Menu,C1MenuItemBinding)
```

C#

```
public class C1MenuItemBindingCollection :
    C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection<C1Menu,C1Me
    nuItemBinding>
```

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.Collection<T>

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemBindingCollection

See Also


Reference

[C1MenuItemBindingCollection Members](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members









The following tables list the members exposed by [C1MenuItemBindingCollection](#).

Public Properties

	Name	Description
	Item	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)

[Top](#)

Public Methods

	Name	Description
	Add	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)
	Clear	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)
	Contains	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)
	CopyTo	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)
	IndexOf	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)
	Insert	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)
	Remove	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)
	RemoveAt	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItemBinding>)

[Top](#)

See Also

Reference

[C1MenuItemBindingCollection Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1MenuItemCollection

Represents a collection of C1MenuItemBase objects that are used by Items property.

Object Model

C1MenuItemCollection

Syntax

Visual Basic (Declaration)	
<pre>Public Class C1MenuItemCollection Inherits C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection(Of IC1MenuItemCollectionOwner,C1MenuItem)</pre>	
C#	
<pre>public class C1MenuItemCollection : C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection<IC1MenuItem CollectionOwner,C1MenuItem></pre>	

Inheritance Hierarchy

System.Object
 System.Collections.ObjectModel.Collection<T>
 C1.Web.Wijmo.Controls.C1Menu.C1MenuItemCollection

See Also

Reference

[C1MenuItemCollection Members](#)
[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

Represents a collection of C1MenuItemBase objects that are used by Items property.

Object Model

C1MenuItemCollection

Syntax

Visual Basic (Declaration)	
<pre>Public Class C1MenuItemCollection Inherits C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection(Of IC1MenuItemCollectionOwner,C1MenuItem)</pre>	

C#

```
public class C1MenuItemCollection :  
C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection<IC1MenuItem  
CollectionOwner, C1MenuItem>
```

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.Collection<T>

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemCollection

See Also


Reference

[C1MenuItemCollection Members](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members



The following tables list the members exposed by [C1MenuItemCollection](#).

Public Constructors

	Name	Description
	C1MenuItemCollection Constructor	Initializes a new instance of the C1MenuItemCollection class.









[Top](#)

Public Properties

	Name	Description
	Events	Events of C1MenuItemCollection Change.
	Item	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItem>)


[Top](#)

Public Methods

	Name	Description
	Add	Adds new menu item to the end of the list.
	Clear	Removes all items from collection.
	Contains	(Inherited from <code>System.Collections.ObjectModel.Collection<C1MenuItem></code>)
	CopyTo	(Inherited from <code>System.Collections.ObjectModel.Collection<C1MenuItem></code>)
	IndexOf	(Inherited from <code>System.Collections.ObjectModel.Collection<C1MenuItem></code>)
	Insert	Insert C1MenuItem item to specific position into the collection.
	Remove	Removes item from list.
	RemoveAt	Removes item from list by index.

[Top](#)

Public Events

	Name	Description
	OnCollectionChanged	Occurs after C1MenuItemCollection collection was changed.

[Top](#)

See Also

Reference

[C1MenuItemCollection Class](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1MenuItemCollection Constructor

Initializes a new instance of the [C1MenuItemCollection](#) class.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
    ByVal owner As IC1MenuItemCollectionOwner _
)
```

C#

```
public C1MenuItemCollection(
    IC1MenuItemCollectionOwner owner
)
```

Parameters*owner*

The owner object of the collection.

See Also**Reference**






[C1MenuItemCollection Class](#)

[C1MenuItemCollection Members](#)

Methods

For a list of all members of this type, see [C1MenuItemCollection members](#).

Public Methods

	Name	Description
⇒ 	Add	Adds new menu item to the end of the list.
⇒ 	Clear	Removes all items from collection.
⇒ 	Contains	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItem>)
⇒ 	CopyTo	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItem>)
⇒ 	IndexOf	(Inherited from

		System.Collections.ObjectModel.Collection<C1MenuItem>)
⇒	Insert	Insert C1MenuItem item to specific position into the collection.
⇒	Remove	Removes item from list.
⇒	RemoveAt	Removes item from list by index.

[Top](#)

See Also

Reference

[C1MenuItemCollection Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Add Method

Child menu item

Adds new menu item to the end of the list.

Syntax

Visual Basic (Declaration)

```
Public Shadows Sub Add( _  
    ByVal child As C1MenuItem _  
)
```

C#

```
public new void Add(  
    C1MenuItem child  
)
```

Parameters

child

Child menu item

See Also

Reference

[C1MenuItemCollection Class](#)
[C1MenuItemCollection Members](#)

Clear Method

Removes all items from collection.

Syntax

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

See Also

Reference

[C1MenuItemCollection Class](#)
[C1MenuItemCollection Members](#)

Insert Method

Insert [C1MenuItem](#) item to specific position into the collection.

Syntax

Visual Basic (Declaration)	
Public Shadows Sub Insert(_ ByVal <i>index</i> As System.Integer, _ ByVal <i>child</i> As C1MenuItem _)	
C#	
public new void Insert(System.int <i>index</i> , C1MenuItem <i>child</i>)	

Parameters

index

Position, value should be mayor or equal to 0

child

Child menu item

See Also

Reference

[C1MenuItemCollection Class](#)

[C1MenuItemCollection Members](#)

Remove Method

Child menu item.

Removes item from list.

Syntax

Visual Basic (Declaration)	
<pre>Public Shadows Sub Remove(_ ByVal <i>child</i> As C1MenuItem _)</pre>	
C#	
<pre>public new void Remove(C1MenuItem <i>child</i>)</pre>	

Parameters

child

Child menu item.

See Also

Reference

[C1MenuItemCollection Class](#)

[C1MenuItemCollection Members](#)

RemoveAt Method

Removes item from list by index.

Syntax

Visual Basic (Declaration)	
<pre>Public Shadows Sub RemoveAt(_ ByVal index As System.Integer _)</pre>	
C#	
<pre>public new void RemoveAt(System.int index)</pre>	

Parameters

index

An index that indicates which submenu to remove.

See Also



Reference

[C1MenuItemCollection Class](#)
[C1MenuItemCollection Members](#)

Properties

For a list of all members of this type, see [C1MenuItemCollection members](#).

Public Properties

	Name	Description
	Events	Events of C1MenuItemCollection Change.
	Item	(Inherited from System.Collections.ObjectModel.Collection<C1MenuItem>)

[Top](#)

See Also

Reference

[C1MenuItemCollection Class](#)
[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Events Property

Events of C1MenuItemCollection Change.

Syntax

Visual Basic (Declaration)	
Public Shadows ReadOnly Property Events As System.ComponentModel.EventHandlerList	
C#	
public new System.ComponentModel.EventHandlerList Events {get;}	

See Also

Reference

[C1MenuItemCollection Class](#)

[C1MenuItemCollection Members](#)

Events

For a list of all members of this type, see [C1MenuItemCollection members](#).

Public Events

	Name	Description
	OnCollectionChanged	Occurs after C1MenuItemCollection collection was changed.

[Top](#)

See Also

Reference

[C1MenuItemCollection Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

OnCollectionChanged Event

Occurs after [C1MenuItemCollection](#) collection was changed.

Syntax

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

Public Event OnCollectionChanged As System.EventHandler

C#

public event System.EventHandler OnCollectionChanged
--

See Also

Reference

[C1MenuItemCollection Class](#)

[C1MenuItemCollection Members](#)

C1MenuItemCollectionChangedEventArgs

C1MenuItemCollectionChangedEventArgs is used for the C1MenuItemCollectionChange event.

Object Model

C1MenuItemCollectionChangedEventArgs

Syntax

Visual Basic (Declaration)

Public Class C1MenuItemCollectionChangedEventArgs Inherits System.EventArgs
--

C#

public class C1MenuItemCollectionChangedEventArgs : System.EventArgs
--

Inheritance Hierarchy

System.Object

System.EventArgs

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemCollectionChangedEventArgs

See Also

Reference

[C1MenuItemCollectionChangedEventArgs Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

C1MenuItemCollectionChangedEventArgs is used for the C1MenuItemCollectionChange event.

Object Model

C1MenuItemCollectionChangedEventArgs

Syntax

Visual Basic (Declaration)	
<pre>Public Class C1MenuItemCollectionChangedEventArgs Inherits System.EventArgs</pre>	
C#	
<pre>public class C1MenuItemCollectionChangedEventArgs : System.EventArgs</pre>	

Inheritance Hierarchy

System.Object

System.EventArgs

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemCollectionChangedEventArgs

See Also

Reference


[C1MenuItemCollectionChangedEventArgs Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members

The following tables list the members exposed by [C1MenuItemCollectionChangedEventArgs](#).



Public Constructors

	Name	Description
	C1MenuItemCollectionChangedEventArgs Constructor	Initializes a new instance of the C1MenuItemCollectionChangedEventArgs class.

[Top](#)

Public Properties

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

	Item	indicate the item which to be add or remove from the C1ToolBarItemCollection.
	Status	indicate the ChangeStatus of the C1ToolBarItemCollection.

[Top](#)

See Also

Reference

[C1MenuItemCollectionChangedEventArgs Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1MenuItemCollectionChangedEventArgs Constructor

Initializes a new instance of the [C1MenuItemCollectionChangedEventArgs](#) class.

Syntax

Visual Basic (Declaration)	
<pre>Public Function New(_ ByVal item As IC1MenuItemCollectionOwner _)</pre>	
C#	
<pre>public C1MenuItemCollectionChangedEventArgs(IC1MenuItemCollectionOwner item)</pre>	

Parameters

item

The owner menu item

See Also

Reference



[C1MenuItemCollectionChangedEventArgs Class](#)

[C1MenuItemCollectionChangedEventArgs Members](#)

Properties

For a list of all members of this type, see [C1MenuItemCollectionChangedEventArgs members](#).

Public Properties

	Name	Description
	Item	indicate the item which to be add or remove from the C1ToolBarItemCollection.
	Status	indicate the ChangeStatus of the C1ToolBarItemCollection.

[Top](#)

See Also

Reference

[C1MenuItemCollectionChangedEventArgs Class](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Item Property

indicate the item which to be add or remove from the C1ToolBarItemCollection.

Syntax

Visual Basic (Declaration)	
Public ReadOnly Property Item As IC1MenuItemCollectionOwner	
C#	
public IC1MenuItemCollectionOwner Item {get;}	

See Also

Reference

[C1MenuItemCollectionChangedEventArgs Class](#)

[C1MenuItemCollectionChangedEventArgs Members](#)

Status Property

indicate the ChangeStatus of the C1ToolBarItemCollection.

Syntax

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

Public ReadOnly Property Status As [ChangeStatus](#)

C#

```
public ChangeStatus Status {get;}
```

See Also

Reference

[C1MenuItemCollectionChangedEventArgs Class](#)

[C1MenuItemCollectionChangedEventArgs Members](#)

C1MenuItemTemplateContainer

Represents a container that holds the contents of a templated menu item in a C1Menu control.

Object Model

C1MenuItemTemplateContainer

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.ToolboxItemAttribute(ToolboxItemType=,
ToolboxItemTypeName="")>
<System.Web.AspNetHostingPermissionAttribute(Level=AspNetHostingPermissionLevel.Minimal,
    Action=SecurityAction.Demand Or _
    SecurityAction.Deny Or _
    SecurityAction.LinkDemand,
    Unrestricted=False)>
Public NotInheritable Class C1MenuItemTemplateContainer
    Inherits System.Web.UI.Control
```

C#

```
[System.ComponentModel.ToolboxItem(ToolboxItemType=, ToolboxItemTypeName="")]
[System.Web.AspNetHostingPermission(Level=AspNetHostingPermissionLevel.Minimal,
    Action=SecurityAction.Demand |
    SecurityAction.Deny |
    SecurityAction.LinkDemand,
    Unrestricted=false)]
```

```
public sealed class C1MenuItemTemplateContainer : System.Web.UI.Control
```

Inheritance Hierarchy

System.Object

System.Web.UI.Control

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer

See Also

Reference

[C1MenuItemTemplateContainer Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

Represents a container that holds the contents of a templated menu item in a C1Menu control.

Object Model

C1MenuItemTemplateContainer

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.ToolboxItemAttribute(ToolboxItemType=,
ToolboxItemTypeName="")>
<System.Web.AspNetHostingPermissionAttribute(Level=AspNetHostingPermissionLevel.Minimal,
    Action=SecurityAction.Demand Or _
    SecurityAction.Deny Or _
    SecurityAction.LinkDemand,
    Unrestricted=False)>
Public NotInheritable Class C1MenuItemTemplateContainer
    Inherits System.Web.UI.Control
```

C#

```
[System.ComponentModel.ToolboxItem(ToolboxItemType=, ToolboxItemTypeName="")]
[System.Web.AspNetHostingPermission(Level=AspNetHostingPermissionLevel.Minimal,
    Action=SecurityAction.Demand |
    SecurityAction.Deny |
```

```

        SecurityAction.LinkDemand,
        Unrestricted=false)]
public sealed class C1MenuItemTemplateContainer : System.Web.UI.Control

```

Inheritance Hierarchy

System.Object

System.Web.UI.Control

C1.Web.Wijmo.Controls.C1Menu.C1MenuItemTemplateContainer

See Also

Reference


[C1MenuItemTemplateContainer Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members





The following tables list the members exposed by [C1MenuItemTemplateContainer](#).


















Public Constructors

	Name	Description
	C1MenuItemTemplateContainer Constructor	Initializes a new instance of the C1MenuItemTemplateContainer class using the specified menu item index and menu item.

[Top](#)













Public Properties

	Name	Description
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	Controls	(Inherited from System.Web.UI.Control)

	DataItem	Gets or sets the menu item associated with the container.
	EnableTheming	(Inherited from System.Web.UI.Control)
	EnableViewState	(Inherited from System.Web.UI.Control)
	ID	(Inherited from System.Web.UI.Control)
	ItemIndex	Gets the index of the menu item associated with the container.
	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.Control)
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	UniqueID	(Inherited from System.Web.UI.Control)
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Visible	(Inherited from System.Web.UI.Control)





[Top](#)



Public Methods

	Name	Description
	ApplyStyleSheetSkin	(Inherited from System.Web.UI.Control)
	DataBind	(Inherited from System.Web.UI.Control)
	Dispose	(Inherited from System.Web.UI.Control)
	FindControl	(Inherited from System.Web.UI.Control)
	Focus	(Inherited from System.Web.UI.Control)
	GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
	GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
	HasControls	(Inherited from System.Web.UI.Control)
	RenderControl	(Inherited from System.Web.UI.Control)
	ResolveClientUrl	(Inherited from System.Web.UI.Control)
	ResolveUrl	(Inherited from System.Web.UI.Control)
	SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)

[Top](#)

Public Events

	Name	Description
	DataBinding	(Inherited from System.Web.UI.Control)
	Disposed	(Inherited from System.Web.UI.Control)
	Init	(Inherited from System.Web.UI.Control)
	Load	(Inherited from System.Web.UI.Control)

	PreRender	(Inherited from System.Web.UI.Control)
	Unload	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1MenuItemTemplateContainer Class](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1MenuItemTemplateContainer Constructor

Initializes a new instance of the C1MenuItemTemplateContainer class using the specified menu item index and menu item.

Syntax

Visual Basic (Declaration)	
<pre>Public Function New(_ ByVal <i>itemIndex</i> As System.Integer, _ ByVal <i>dataItem</i> As C1MenuItem _)</pre>	
C#	
<pre>public C1MenuItemTemplateContainer(System.int <i>itemIndex</i>, C1MenuItem <i>dataItem</i>)</pre>	

Parameters

itemIndex

The index of the item

dataItem

An menuitem object provide data to the template

See Also

Reference















[C1MenuItemTemplateContainer Class](#)








[C1MenuItemTemplateContainer Members](#)

Properties

For a list of all members of this type, see [C1MenuItemTemplateContainer members](#).

Public Properties

	Name	Description
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	Controls	(Inherited from System.Web.UI.Control)
	DataItem	Gets or sets the menu item associated with the container.
	EnableTheming	(Inherited from System.Web.UI.Control)
	EnableViewState	(Inherited from System.Web.UI.Control)
	ID	(Inherited from System.Web.UI.Control)
	ItemIndex	Gets the index of the menu item associated with the container.
	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.Control)
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	UniqueID	(Inherited from System.Web.UI.Control)
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Visible	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1MenuItemTemplateContainer Class](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

DataItem Property

Gets or sets the menu item associated with the container.

Syntax

Visual Basic (Declaration)

```
Public Property DataItem As System.Object
```

C#

```
public System.object DataItem {get; set;}
```

See Also

Reference

[C1MenuItemTemplateContainer Class](#)[C1MenuItemTemplateContainer Members](#)

ItemIndex Property

Gets the index of the menu item associated with the container.

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property ItemIndex As System.Integer
```

C#

```
public System.int ItemIndex {get;}
```

See Also

Reference

[C1MenuItemTemplateContainer Class](#)

[C1MenuItemTemplateContainer Members](#)

C1MenuSerializer

Object Model

C1MenuSerializer

Syntax

Visual Basic (Declaration)

```
Public Class C1MenuSerializer
    Inherits C1.Web.Wijmo.Controls.Base.C1BaseSerializer(Of
C1Menu,C1MenuItem,IC1MenuItemCollectionOwner)
```

C#

```
public class C1MenuSerializer :
C1.Web.Wijmo.Controls.Base.C1BaseSerializer<C1Menu,C1MenuItem,IC1MenuItemColl
ectionOwner>
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1Menu.C1MenuSerializer

See Also

Reference

[C1MenuSerializer Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

Object Model

C1MenuSerializer

Syntax

Visual Basic (Declaration)

```
Public Class C1MenuSerializer
    Inherits C1.Web.Wijmo.Controls.Base.C1BaseSerializer(Of
C1Menu, C1MenuItem, IC1MenuItemCollectionOwner)
```

C#

```
public class C1MenuSerializer :
C1.Web.Wijmo.Controls.Base.C1BaseSerializer<C1Menu, C1MenuItem, IC1MenuItemColl
ectionOwner>
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1Menu.C1MenuSerializer

See Also

Reference


[C1MenuSerializer Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members

The following tables list the members exposed by [C1MenuSerializer](#).

Public Constructors

	Name	Description
	C1MenuSerializer Constructor	Initializes a new instance of the C1MenuSerializer class.

[Top](#)

See Also

Reference

[C1MenuSerializer Class](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

C1MenuSerializer Constructor

Initializes a new instance of the [C1MenuSerializer](#) class.

Syntax

Visual Basic (Declaration)	
<pre>Public Function New(_ ByVal obj As System.Object _)</pre>	
C#	
<pre>public C1MenuSerializer(System.Object obj)</pre>	

Parameters

obj

See Also

Reference

[C1MenuSerializer Class](#)[C1MenuSerializer Members](#)

Enumerations

ChangeStatus

Specifies [IC1MenuItemCollectionOwner](#) change status.

Syntax

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

```
Public Enum ChangeStatus
    Inherits System.Enum
```

C#

```
public enum ChangeStatus : System.Enum
```

Members

Member	Description
ItemAdd	IC1MenuItemCollectionOwner add.
ItemRemove	IC1MenuItemCollectionOwner remove.

Inheritance Hierarchy

```
System.Object
  System.ValueType
    System.Enum
      C1.Web.Wijmo.Controls.C1Menu.ChangeStatus
```

See Also

Reference

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

ImagePosition

Decides which side the image displays on menu item.

Syntax

Visual Basic (Declaration)

```
Public Enum ImagePosition
    Inherits System.Enum
```

C#

```
public enum ImagePosition : System.Enum
```

Members

Member	Description
Left	Show the image on the left side of the menu item.
Right	Show the image on the right side of the menu item.

Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

C1.Web.Wijmo.Controls.C1Menu.ImagePosition

See Also

Reference

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

MenuMode

Decides which mode the menu displays.

Syntax

Visual Basic (Declaration)	
<pre>Public Enum MenuMode Inherits System.Enum</pre>	
C#	
<pre>public enum MenuMode : System.Enum</pre>	

Members

Member	Description
Flyout	Show the menu as flyout style.
Sliding	Show the meun as i-pod style

Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

C1.Web.Wijmo.Controls.C1Menu.MenuMode

See Also

Reference

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

SubmenuTriggerEvent

If menu is flyout mode, this value specially decide which event handle to show the submenu.

Syntax

Visual Basic (Declaration)	
<pre>Public Enum SubmenuTriggerEvent Inherits System.Enum</pre>	
C#	
<pre>public enum SubmenuTriggerEvent : System.Enum</pre>	

Members

Member	Description
Click	Click event
Dbclick	Double click event
Default	Follow the TriggerEvent property.
Mouseenter	Mouseover and mouseout event
Rtclick	Right click event.

Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

C1.Web.Wijmo.Controls.C1Menu.SubmenuTriggerEvent

See Also

Reference

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

TriggerEvent

If menu is flyout mode, this value decide which event handle to show the menu and submenu.

Syntax

Visual Basic (Declaration)	
<pre>Public Enum TriggerEvent Inherits System.Enum</pre>	
C#	
<pre>public enum TriggerEvent : System.Enum</pre>	

Members

Member	Description
Click	Click event
Dbclick	Double click event
Mouseenter	Mouseover and mouseout event
Rtclick	Right click event.

Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

C1.Web.Wijmo.Controls.C1Menu.TriggerEvent

See Also

Reference

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Interfaces

IC1MenuItemCollectionOwner

IC1MenuItemCollectionOwner interface.

Object Model

IC1MenuItemCollectionOwner

Syntax

Visual Basic (Declaration)	
Public Interface IC1MenuItemCollectionOwner	
C#	
public interface IC1MenuItemCollectionOwner	

See Also

Reference

[IC1MenuItemCollectionOwner Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Overview

IC1MenuItemCollectionOwner interface.

Object Model

IC1MenuItemCollectionOwner

Syntax

Visual Basic (Declaration)	
Public Interface IC1MenuItemCollectionOwner	
C#	

```
public interface IC1MenuItemCollectionOwner
```

See Also

Reference



[IC1MenuItemCollectionOwner Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Members

The following tables list the members exposed by [IC1MenuItemCollectionOwner](#).

Public Properties

	Name	Description
	Items	Menu items collection
	Owner	Menu item owner.

[Top](#)

See Also

Reference



[IC1MenuItemCollectionOwner Interface](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Properties

For a list of all members of this type, see [IC1MenuItemCollectionOwner members](#).

Public Properties

	Name	Description
	Items	Menu items collection
	Owner	Menu item owner.

[Top](#)

See Also

Reference

[IC1MenuItemCollectionOwner Interface](#)[C1.Web.Wijmo.Controls.C1Menu Namespace](#)

Items Property

Menu items collection

Syntax

Visual Basic (Declaration)	
ReadOnly Property Items As C1MenuItemCollection	
C#	
C1MenuItemCollection Items {get;}	

See Also

Reference

[IC1MenuItemCollectionOwner Interface](#)[IC1MenuItemCollectionOwner Members](#)

Owner Property

Menu item owner.

Syntax

Visual Basic (Declaration)	
ReadOnly Property Owner As IC1MenuItemCollectionOwner	
C#	
IC1MenuItemCollectionOwner Owner {get;}	

See Also

Reference

[IC1MenuItemCollectionOwner Interface](#)[IC1MenuItemCollectionOwner Members](#)

Delegates

C1MenuEventHandler

The source of the event

C1MenuItemArgs object that contains the event data

Delegate type for handling events that are related to the items.

Syntax

Visual Basic (Declaration)

```
Public Delegate Sub C1MenuEventHandler( _  
    ByVal sender As System.Object, _  
    ByVal e As C1MenuItemArgs _  
)
```

C#

```
public delegate void C1MenuEventHandler(  
    System.Object sender,  
    C1MenuItemArgs e  
)
```

Parameters

sender

The source of the event

e

C1MenuItemArgs object that contains the event data

See Also

Reference

[C1MenuEventHandler Members](#)

[C1.Web.Wijmo.Controls.C1Menu Namespace](#)