
ComponentOne

TreeView 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
Quick Start	12
Step 1 of 3: Adding C1TreeView to the Page	12
Step 2 of 3: Creating a TreeView Using the Designer	12
Step 3 of 3: Running the Project	14
Design-Time Support	15
Smart Tag	15
NodeBinding Collection Editor	18
Designer Form.....	19
Exploring the Designer Form	19
Designer Form Menu	21
Designer Form Toolbar	22
How to Use the Designer	23
Deleting a C1TreeNode	24
Renaming the TreeViewNode in the Designer.....	24
Adding a Child Node.....	24
Inserting a Node.....	25
TreeView Structure and Elements	25
TreeView Creation	27
Static Creation.....	27
Dynamic Creation.....	29
Appearance and Behavior.....	33
Themes.....	33
CSS Selectors	34
Check Boxes	35
Drag and Drop Nodes.....	37
Load on Demand.....	38
Node Selection	38
Node Navigation	38
Task-Based Help.....	39

Creating and Configuring Check Box Nodes	40
Creating Node Check Boxes	40
Preventing Child Nodes from Being Automatically Checked	41
Working with Themes	43
Using a Built-In Theme	43
Using a Custom Theme	45
Working with CSS Selectors	47
Adding a Top-Level Node	47
Adding a Child Node	49
Adding a Child Node with a ButtonClick Event	51
Populating C1TreeView with a Site Map	53
Populating C1TreeView with XML	55
Populating C1TreeView Dynamically	57
Saving and Loading a C1TreeView from XML	60
Setting the Auto Collapse Property	61
Setting C1TreeView to Open on Hover	64
Setting C1TreeView Node Icons.....	65
Setting C1TreeView Properties to Allow Drag-and-Drop Behaviors.....	68
Drag-and-drop Behaviors Within One Tree Structure	68
Drag-and-drop Behaviors Between Two Tree Structures	69
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	78
Members.....	79
C1TargetControlBase	79
Overview	80
Members.....	81
Methods.....	85

GetProperty<V> Method	87
GetScriptDescriptors Method	87
GetScriptReferences Method	88
RegisterDesignTimeStyleSheet Method	88
SetProperty<V> Method	89
Properties.....	89
CDNDependencyPaths Property	92
CDNPath Property	93
DisplayVisible Property	94
EnableCombinedJavaScripts Property	94
Enabled Property	95
Theme Property	96
ThemeSwatch Property	97
UseCDN Property	98
WijmoControlMode Property	98
WijmoCssAdapter Property	99
C1TargetControlHelper	100
Overview	101
Members.....	101
C1TargetControlHelper Constructor.....	102
Methods.....	102
ResolveEmbeddedResourceUrl Method.....	103
C1TargetControlHelper<T>	103
Overview	104
Members.....	105
C1TargetControlHelper<T> Constructor	106
Methods.....	107
EnsureEnabledState Method	108
GetScriptDescriptors Method	108
GetScriptDescriptors() Method.....	109
GetScriptDescriptors(String) Method	109
RegisterDesignTimeStyleSheet Method	110
RegisterDesignTimeStyleSheets Method.....	110
RegisterIncludes Method.....	111

RegisterOnSubmitStatement Method	111
RegisterRunTimeStylesheets Method.....	112
RegisterScriptDescriptors Method.....	112
RegisterScriptDescriptorsWithoutScriptManager Method	113
RegisterScriptDescriptorsWithScriptManager Method	113
RenderJsonDataField Method	113
ShowAbout Method.....	114
Properties.....	114
Control Property	115
JsonSerializableHelper Property	115
Page Property.....	116
ScriptManager Property.....	116
WijChartNavigator	117
Overview	117
Members.....	118
WijChartNavigator Constructor	118
C1.Web.Wijmo.Controls.C1TreeView Namespace	119
Overview	119
Classes.....	120
C1TreeView	120
Overview	121
Members.....	122
C1TreeView Constructor	129
Methods.....	130
CreateTreeViewNode Method.....	131
DataBind Method.....	132
ExpandAllNode Method.....	132
ExpandAllNode() Method	132
ExpandAllNode(IC1TreeViewNodeCollectionOwner) Method	133
FindNodeByNavigateUrl Method.....	133
FindNodeByText Method.....	134
FindNodeByValue Method.....	134
LoadLayout Method.....	135
LoadLayout(String) Method	135

LoadLayout(Stream) Method	136
LoadLayout(String,LayoutType) Method	137
LoadLayout(Stream,LayoutType) Method	137
SaveLayout Method	138
SaveLayout(String) Method	138
SaveLayout(Stream) Method	139
Properties.....	139
AllowDrag Property.....	144
AllowDrop Property	144
AllowEdit Property	145
AllowSorting Property.....	145
AllowTriState Property.....	145
AutoCheckNodes Property.....	146
AutoCollapse Property.....	146
AutoPostBack Property	147
CollapseAnimation Property	147
CollapseDelay Property.....	148
DataBindings Property	148
DataBindStartLevel Property	148
ExpandAnimation Property.....	149
ExpandCollapseHoverUsed Property	149
ExpandDelay Property	150
LoadOnDemand Property	150
Nodes Property	150
NodesTemplate Property.....	151
OnClientNodeCheckChanged Property.....	152
OnClientNodeCheckChanging Property.....	152
OnClientNodeClick Property	153
OnClientNodeCollapsed Property.....	153
OnClientNodeDragging Property	153
OnClientNodeDragStarted Property	154
OnClientNodeDropped Property	154
OnClientNodeExpanded Property.....	155
OnClientNodeMouseOut Property	155

OnClientNodeMouseOver Property.....	156
OnClientNodeTextChanged Property	156
OnClientSelectedNodeChanged Property.....	157
Owner Property	157
SelectedNodes Property	158
ShowCheckBoxes Property	158
ShowExpandCollapse Property	159
UniqueID Property	159
Events.....	159
NodeCheckChanged Event.....	160
NodeClicked Event	161
NodeCollapsed Event.....	162
NodeDataBound Event	163
NodeDropped Event	164
NodeExpanded Event.....	164
NodeTextChanged Event	165
SelectedNodesChanged Event	166
C1TreeViewEventArgs.....	167
Overview	167
Members.....	168
C1TreeViewEventArgs Constructor.....	169
Properties.....	169
Node Property.....	170
C1TreeViewNode	170
Overview	171
Members.....	171
C1TreeViewNode Constructor	176
Methods.....	176
IsSearchedUrl Method	178
Properties.....	178
AllowDrag Property.....	181
AllowDrop Property	182
Checked Property.....	182
CheckState Property	183

CollapsedIconClass Property	184
Expanded Property	184
ExpandedIconClass Property	185
ItemIconClass Property	186
Nodes Property	186
Owner Property	187
Selected Property	187
Template Property	188
Text Property	189
TreeView Property	190
Url Property	190
Value Property	191
C1TreeViewNodeBinding	191
Overview	192
Members	193
C1TreeViewNodeBinding Constructor	194
C1TreeViewNodeBinding Constructor()	195
C1TreeViewNodeBinding Constructor(String)	195
Properties	196
CollapsedIconClassField Property	196
DataMember Property	197
Depth Property	198
EnabledField Property	199
ExpandedField Property	200
ExpandedIconClassField Property	200
FormatString Property	201
ItemIconClassField Property	202
NavigateUrlField Property	203
TextField Property	203
ValueField Property	204
C1TreeViewNodeBindingCollection	205
Overview	206
Members	207
Methods	208

Add Method	209
Contains Method	210
CopyTo Method	210
CopyTo(C1TreeViewNodeBinding[],Int32) Method	211
IndexOf Method	211
Insert Method	212
Remove Method	212
RemoveAt Method	213
Properties	214
Item Property	214
C1TreeViewNodeCollection	215
Overview	215
Members	216
C1TreeViewNodeCollection Constructor	218
Methods	218
Add Method	219
CheckNodes Method	220
FindNode Method	220
FindNodeByText Method	221
FindNodeByValue Method	221
Insert Method	222
Remove Method	223
C1TreeViewNodeDroppedEventArgs	223
Overview	224
Members	225
C1TreeViewNodeDroppedEventArgs Constructor	225
Properties	226
CopyUsed Property	227
Handled Property	227
C1TreeViewSerializer	227
Overview	228
Members	229
C1TreeViewSerializer Constructor	229
Enumerations	230

C1TreeNodeCheckState	230
Interfaces	231
IC1TreeNodeCollectionOwner	231
Overview	231
Members.....	232
Properties.....	232
Nodes Property	233
Owner Property	233
Delegates	233
C1TreeViewEventHandler	233

Overview

Present items in a hierarchical tree structure with **TreeView for ASP.NET Web Forms**. The **C1TreeView** control supports expand/collapse animations, stylish themes, and the ever popular drag-and-drop functionality.

In this section

This section is the User Guide, with a quick start guide, conceptual topics, and task-based help.

[Key Features](#)

[Quick Start](#)

[C1TreeView Concepts](#)

[Samples](#)

[Task-Based Help](#)

In other sections

The API reference section contains a list of all classes, properties, methods, and events for members of C1.Web.Wijmo.Controls.C1TreeView in addition to a list of the client-side options, methods, and events.

[C1TreeView Class](#) (Server-side reference)

[Client-Side Reference](#)

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 [C1TreeView](#) control contains the following key features:

- **Data Binding Support**

Bind the [C1TreeView](#) control to a data source; you can bind to an XML or SiteMap data source, or you can even read data from an Access data source and create the [C1TreeView](#) hierarchy dynamically.

- **Drag-and-drop Nodes**

You can drag-and-drop [C1TreeViewNodes](#) on nodes, in between nodes, or from one tree to another tree. Visual cues, such as a vertical gray line, are used to show you where the [C1TreeViewNode](#) is going to be dropped.

- **Check Box Support**

Node items can be implemented as regular check boxes. This enables end-users to check or uncheck the boxes to select or unselect the corresponding nodes. When the check boxes are enabled for the [C1TreeView](#) you can create an action when the status of a check box changes between posts.

- **Animation**

[C1TreeView](#) supports expand and collapse animation effects. Typically collapsing treeview items use animations that scroll in, fade in, fold in, close or drop in and expanding treeview items use animations that scroll out, fade out, fold out, open, or drop out. You also have the flexibility to specify transition effects and how long the animation lasts.

- **Templates Support**

Change the tree view's appearance with the built-in template editor. Add your own elements, including text, images, and controls such as buttons, to various nodes.

- **Keyboard Support**

Add access key support to give the [C1TreeView](#) control focus with a chosen key combination. This enables end-users to use the keyboard arrow keys to navigate through the treeview items.

- **Theming**

With just a click of the SmartTag, change the treeview's look by selecting one of the 5 premium themes (Midnight, Aristo, Rocket, Cobalt, and Sterling). Optionally, useThemeRoller from 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 treeview to your organization's standards.

See Also

[Quick Start](#)

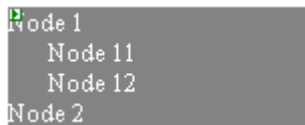
Quick Start

Step 1 of 3: Adding C1TreeView to the Page

In this step, you will learn how to create a new ASP.NET Web site and add a [C1TreeView](#) control to your project.

To begin the Quick Start, complete the following steps:

1. Begin by creating a new ASP.NET Web Site.
2. While in Design view navigate to the Visual Studio Toolbox and double-click the [C1TreeView](#) icon to add the [C1TreeView](#) control to your page.



```
Node 1
  Node 11
  Node 12
Node 2
```

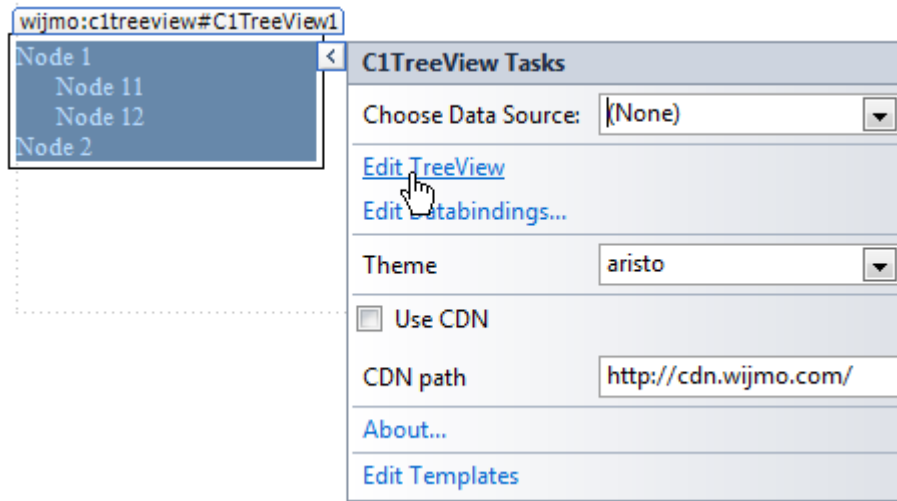
See Also

[Step 2 of 3: Creating a TreeView Using the Designer](#)

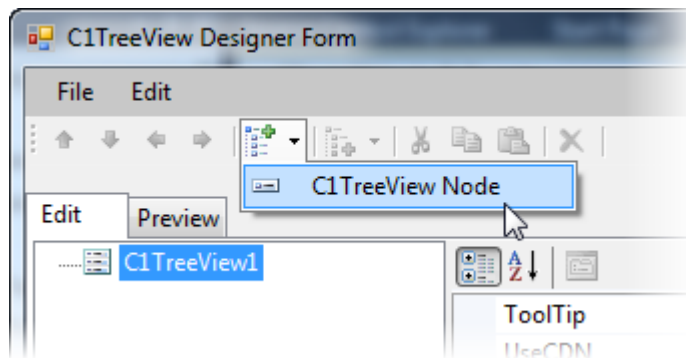
Step 2 of 3: Creating a TreeView Using the Designer

This step will show you how to create root and child nodes, apply a visual style, and display check boxes next to the nodes.

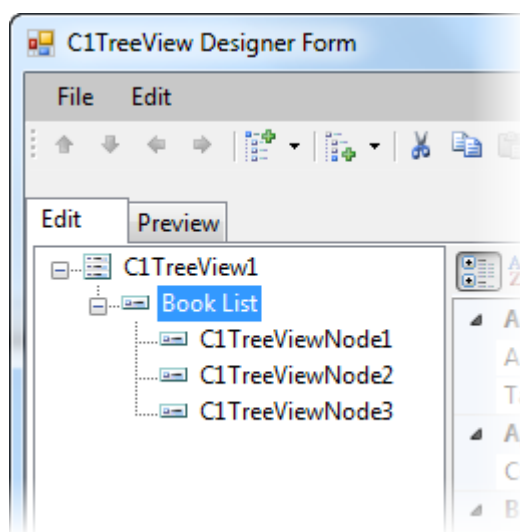
1. Select the [C1TreeView](#) control and click on the smart tag to open its **Tasks** menu.
2. Select **Edit TreeView** from **C1TreeView Tasks** menu to open the designer.



- Right-click on the **C1TreeView** item and select **Add Child | C1TreeView Node** to add the root to the **C1TreeView** control. Set the **C1TreeViewNode1**'s **Text** property to "Book List".



- Right-click on **Book List** and select **Add Child** to create a child for the root node. Repeat this two more times. Three child nodes will exist under the Book List.



- Select the first node under the Book List and set its **Text** property to "Language Books".

6. Select the second node under the Book List and set its **Text** property to "Security Books".
7. Select the third node under the Book List and set its **Text** property to "Classic Books".
8. Right-click on the **Classic Books** node and select **Add Child** to create a child for the Classic Books node. Repeat this to create two **C1TreeViewNodes** under the **Classic Books** node.
9. Select the first node under the **Classic Books** node and set its **Text** property to "The Great Gatsby".
10. Select the second node under the **Classic Books** node and set its **Text** property to "Catch-22".
11. Right-click on **Book List** and select **Add Child** to add to add a child node.

In this step, you nodes and child nodes to the **C1TreeView** control. In the next step, you'll run the project and see the results of this quick start.

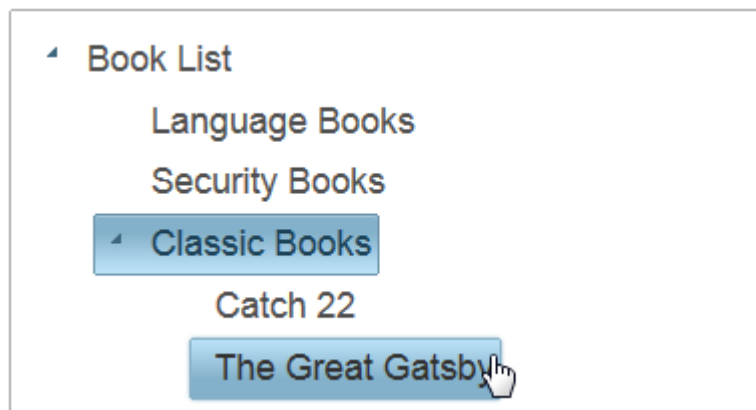
See Also

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

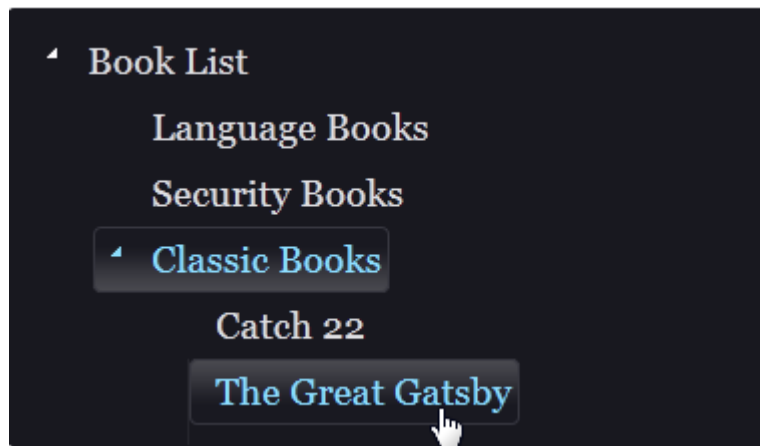
Step 3 of 3: Running the Project

In this step, you'll run the project and see the results of this quick start.

1. Save and run your project and observe the following:
 - The root node, Book List, isn't expanded since the **Expanded** property was set to **False** by default.
 - Expand the **Book List** node and notice the child nodes you created.
 - Notice the theme is the default theme, Aristo.



2. Go back to your project in the design view and click the **C1TreeView**'s smart tag to open the **C1TreeView Tasks** list. Click the **Theme** drop-down arrow and select **midnight** from the list.
3. Save and run your project and notice the new theme, midnight, is applied to **C1TreeView**.



4. Go back to your project in the Design view and open the **TreeView Designer Form**.
5. Select the **C1TreeView** item and in the Properties window, set its behavior properties to the following:
 - [AllowDrag](#) to **True**
 - [AllowDrop](#) to **True**.
 - [ShowCheckBoxes](#) to **True**
6. Click **OK** to save and close the designer.
7. Run the project and observe the following:
 - Expand the Book List node and notice the check boxes next to each treeview node.
 - Select any of the treeview nodes and drag it to a new location.

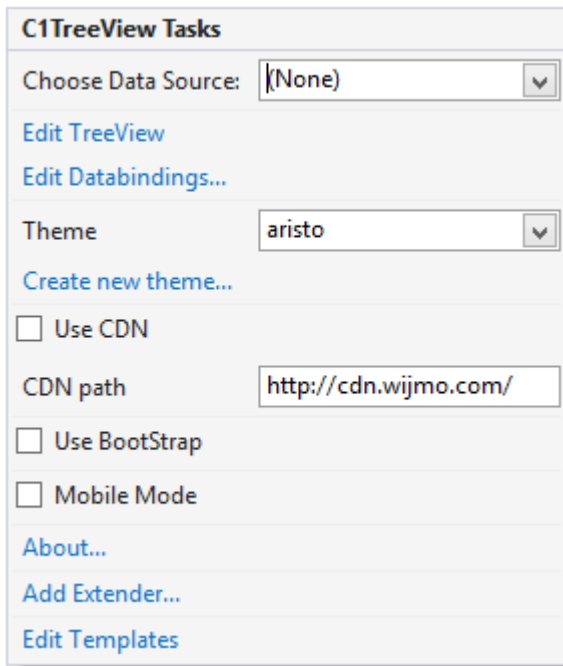
Design-Time Support

Smart Tag

The [C1TreeView](#) control includes a smart tag in Visual Studio. A smart tag represents a short-cut tasks menu that provides the most commonly used properties in [C1TreeView](#).

The [C1TreeView](#) control provides quick and easy access to the **TreeView Designer Form** and common properties through its smart tag.

To access the **C1TreeView Tasks** menu, click on the smart tag in the upper-right corner of the [C1TreeView](#) control. This will open the **C1TreeView Tasks** menu.



The **C1TreeView 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 TreeView**

Clicking on the **Edit TreeView** item opens the **TreeView Designer Form** where you can quickly configure **C1TreeView**'s elements without having to scroll through its Properties window. Here you can add, remove, and re-order **C1TreeViewNodes** as well as set a variety of properties defining their appearance, behavior, and more. For more information on the **TreeView Designer Form**, see [TreeView Designer Form](#).

- **Edit 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 visual styles. See [C1TreeView Themes](#) for more information.

- **Create new theme...**

The **Create new theme...** option opens **ThemeRoller for Visual Studio**. This allows you to customize a theme without leaving your development environment. To find more information on using **ThemeRoller** in your application, see [ThemeRoller for Visual Studio](#).

- **Use CDN**

Determines whether the control is using the CDN for the client-side reference.

- **CDN Path**

The path to the CDN library you are using.

- **Use Bootstrap**

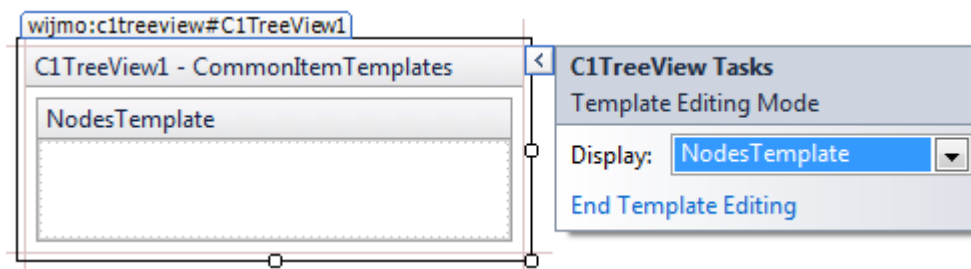
Selecting the **Use Bootstrap** option applies Bootstrap theming to your control. To find more information on using **Bootstrap** theming in your application, see [Bootstrap Theming](#).

- **About**

Clicking 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**

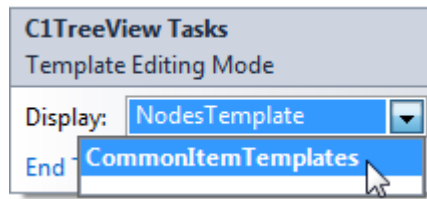
Clicking on the **Edit Templates** item switches the **C1TreeView** control to Template Editing Mode:



In Template Editing Mode, the **C1TreeView 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 **C1TreeView Tasks** menu.

See Also

[NodeBinding Collection Editor](#)

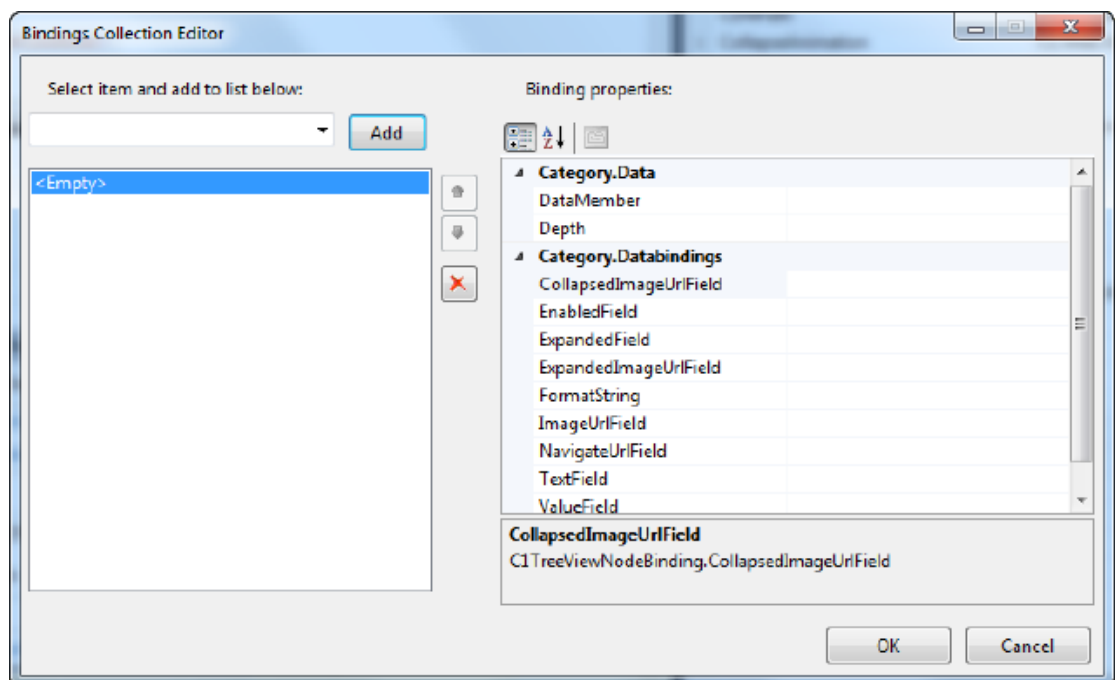
NodeBinding Collection Editor

The [C1TreeView](#) control includes a collection editor that allows you to add or remove databindings from the [C1TreeViewNodeCollection](#), as well as specify binding properties.

There are two ways to access the **C1TreeViewNodeBinding Collection Editor**:

From the C1TreeView Tasks menu:

1. Click the smart tag in the upper-right corner of the C1TreeView control to open the **C1TreeView Tasks** menu.
2. Select **Edit databindings**. The **Bindings Collection Editor** appears.



From the TreeView Designer Form:

1. Click the smart tag in the upper-right corner of the C1TreeView control to open the **C1TreeView Tasks** menu.
2. Select **Edit TreeView**. The **TreeView Designer Form** appears.
3. With the [C1TreeView](#) control selected, click the **ellipsis** button next to the [DataBindings](#) property. The **C1TreeViewNodeBinding Collection Editor** appears. This dialog box, although it appears slightly different, is essentially the same as and contains the same properties as the **Bindings Collection Editor**.

See Also

[Designer Form](#)

Designer Form

The **TreeView Designer Form** is **C1TreeView**'s designer for editing its properties, as well as the [C1TreeViewNode](#) properties. The **TreeView Designer Form** is similar to the Properties window as it allows programmers to modify the control visually. However, it allows you to select a [C1TreeViewNode](#), set its properties, manipulate the nodes, and then preview the appearance of the [C1TreeView](#) control, all within the form.

In this topic you will become familiar with the **TreeView Designer Form's** design interface so you can use the commands within it to edit C1TreeView with minimal effort and time.

To open the **TreeView Designer Form**, click the [C1TreeView](#) smart tag and select the **Edit TreeView** link from the **C1TreeView Tasks** menu.

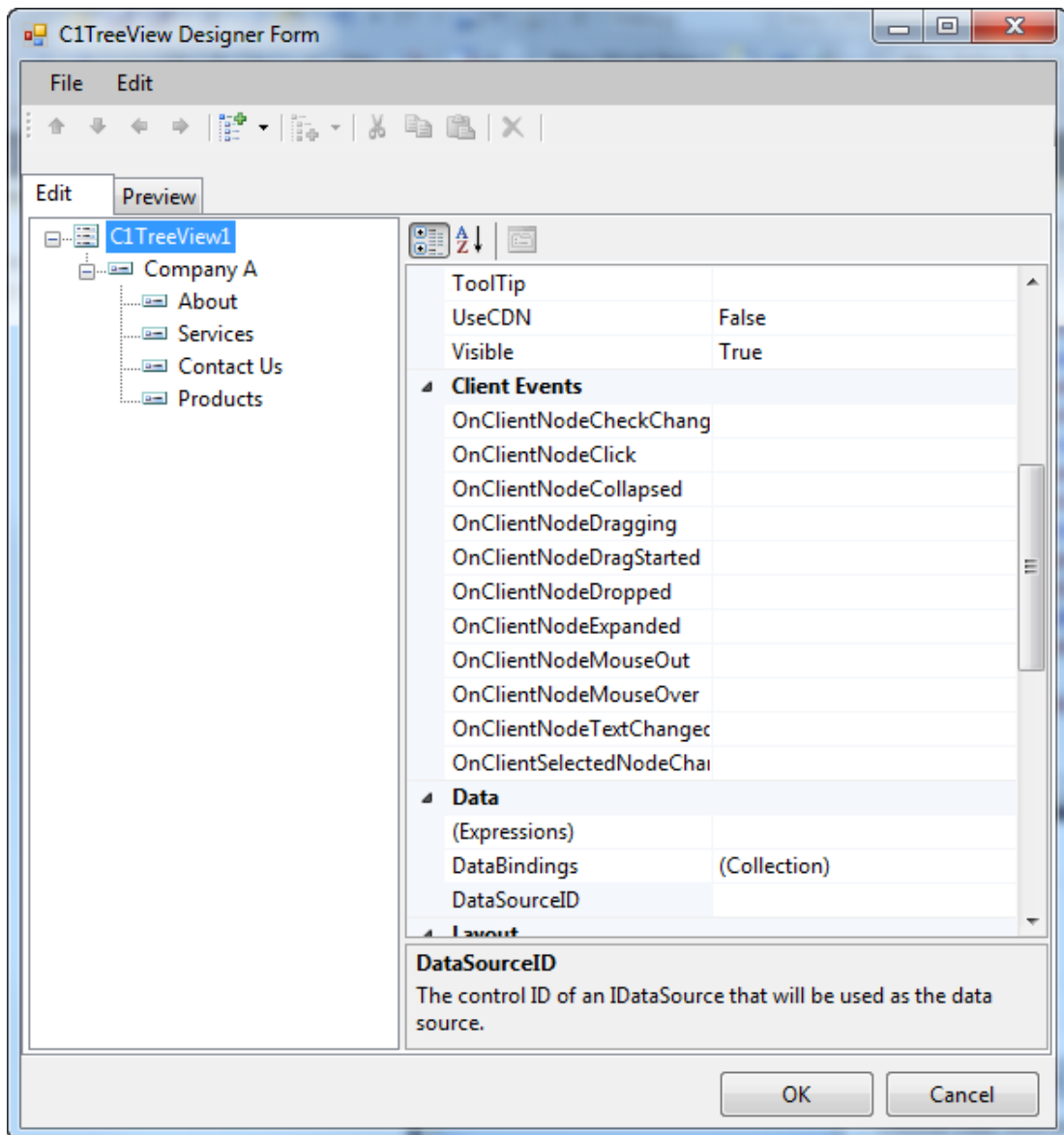
See Also

[Exploring the Designer Form](#)

[How to Use the Designer](#)

Exploring the Designer Form

The **TreeView Designer Form** contains a menu, toolbar, **Edit** tab, **Preview** tab, and properties pane.



- **Edit Tab**

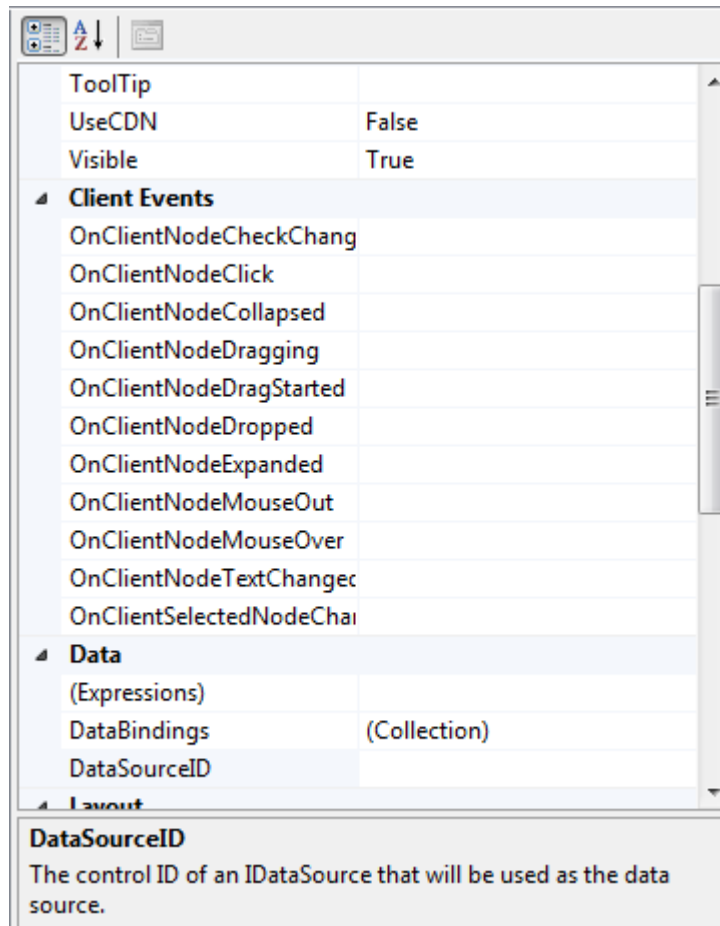
Click the **Edit** tab and select the [C1TreeView](#) control or the desired [C1TreeViewNode](#) for which you would like to manipulate or adjust the properties.

- **Preview Tab**

Click the **Preview** tab for a WYSIWYG preview of what the [C1TreeView](#) control will look like.

- **Properties Pane**

The **TreeView Designer Form** properties pane is almost identical to the Visual Studio Properties window. Simply select a [C1TreeViewNode](#) or the [C1TreeView](#) control and set the desired properties here.



- **Command Buttons**

The command buttons are summarized in the following table:

Button	Description
OK	Clicking OK applies the new settings to the C1TreeView control.
Cancel	Clicking Cancel closes the TreeView Designer Form , cancelling the new settings and applying the default settings to the C1TreeView control.

See Also

[Designer Form Menu](#)

[Designer Form Toolbar](#)

Designer Form Menu

The **TreeView Designer Form** menu contains the following menu items and subitems:

Menu Item	Submenu Item	Description
File	Load from XML	Load the formatting for a C1TreeView control from an .xml file.
	Save as XML	Save the current formatting of the C1TreeView control to an .xml file.
	Exit	Closes the TreeView Designer Form .
Edit	Insert Item	Inserts a new C1TreeNode at the specified place in the list of nodes.
	Add Child	Adds a new C1TreeNode as a child of the C1TreeView or of another C1TreeNode .
	Cut	Cuts the selected C1TreeNode to be moved in the list of nodes.
	Copy	Copies the selected C1TreeNode .
	Paste	Pastes a C1TreeNode at the specified location in the list of nodes.
	Delete	Removes the selected C1TreeNode .
	Rename	Allows you to change the name of the C1TreeNode .

See Also











[Designer Form Toolbar](#)

Designer Form Toolbar

The toolbar for the TreeView Designer Form appears like the following:



The table below describes each button in the toolbar:

Button	Name	Description
	Move Item Up	Moves the selected C1TreeViewNode up in the list of nodes.
	Move Item Down	Moves the selected C1TreeViewNode down in the list nodes.
	Move Item Left	Moves the selected C1TreeViewNode to the left in the hierarchy.
	Move Item Right	Moves the selected C1TreeViewNode to the right in the hierarchy.
	Add Child Item	Inserts a C1TreeViewNode as a child of the C1TreeView control or of another C1TreeViewNode .
	Insert Item	Inserts a C1TreeViewNode at the specified location in the list of nodes.
	Cut	Cuts the selected C1TreeViewNode to be moved in the list of nodes.
	Copy	Copies the selected C1TreeViewNode .
	Paste	Pastes a C1TreeViewNode at the specified location in the list of nodes.
	Delete	Removes the selected C1TreeViewNode .

See Also

[How to Use the Designer](#)

How to Use the Designer

The following topics illustrate how to use the **TreeView Designer Form** for several tasks.

See Also

[Deleting a C1TreeViewNode](#)

[Renaming the TreeViewNode in the Designer](#)

[Adding a Child Node](#)

[Inserting a Node](#)

Deleting a C1TreeNode

You can use one of the following three methods when deleting a [C1TreeNode](#) in the designer:

- **Deleting a child node through the shortcut menu**

Right-click the [C1TreeNode](#) you wish to delete and select **Delete**.

- **Deleting a child node by pressing on the delete button**

Select the node you wish to delete and click on the **Delete**.

- **Deleting a child node through the Edit menu.**

Click the Edit menu and select **Delete**.

See Also

[Renaming the TreeNode in the Designer](#)

Renaming the TreeNode in the Designer

You can use one of the following three methods when renaming a [C1TreeNode](#) in the designer:

- **Pressing F2**
 - a. Select the [C1TreeNode](#) you wish to rename.
 - b. Press the F2 key and type the new name.
- **Selecting rename from the shortcut menu**
 - a. Right-click the [C1TreeNode](#) you wish to rename.
 - b. Select **Rename** from the context menu and type the new name.
- **Selecting rename from the Edit menu**
 - a. Select the [C1TreeNode](#) you wish to rename.
 - b. Click the Edit menu, select **Rename**, and enter the new name.

See Also

[Adding a Child Node](#)

Adding a Child Node

You can use one of the following three methods when adding a child [C1TreeNode](#) in the designer:

- **Adding a child node through the shortcut menu**

Right-click on the node you wish to add a child node and select **Add Child | C1TreeViewNode**.

- **Adding a child node by pressing on the add child button**

Select the node you wish to add a child node and click on the **Add Child** button's drop-down arrow and select [C1TreeViewNode](#).

- **Adding a child node through the Edit menu**

Select the node you wish to add a child node and click on the **Edit** menu and select **Add Child | C1TreeViewNode**.

See Also

[Inserting a Node](#)

Inserting a Node

You can use one of the following three methods when inserting a [C1TreeViewNode](#) in the designer:

- **Inserting a node through the shortcut menu**

Right-click on the [C1TreeViewNode](#) you wish to add a child node and select **Insert Item | C1TreeViewNode**.

- **Inserting a node by pressing on the Insert Node button**

Select the [C1TreeViewNode](#) you wish to add a child node and click on the **Insert Item** button's drop-down arrow and select [C1TreeViewNode](#).

- **Inserting a node using the Edit menu**

Select the [C1TreeViewNode](#) you wish to add a child node and click on the **Edit** menu and select **Insert Item | C1TreeViewNode**.

See Also

[TreeView Structure and Elements](#)

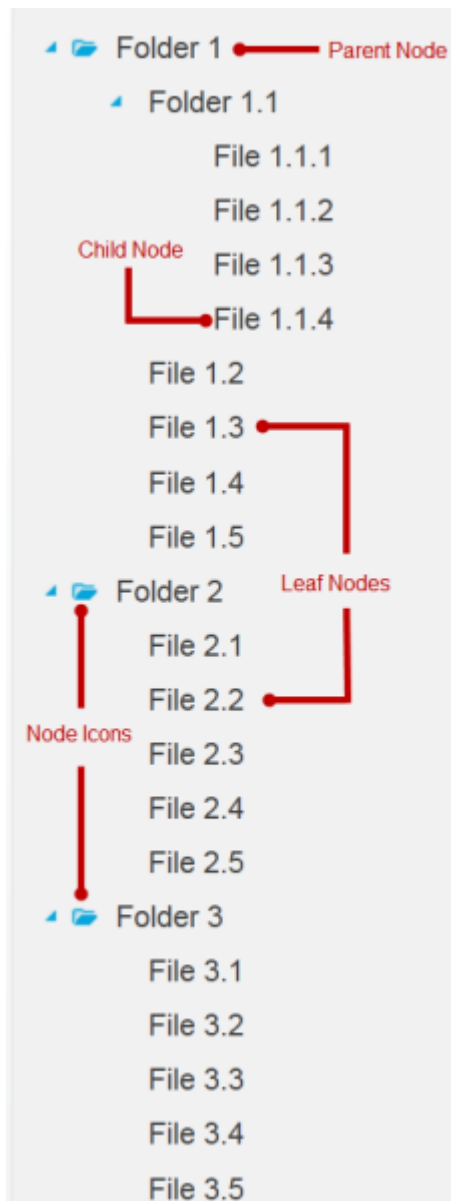
TreeView Structure and Elements

[C1TreeView](#) is a tree type Web control that displays a hierarchical tree structure. A tree structure is used to represent hierarchical data into a graphical form. This section will provide a visual and written overview of the structure and elements of [C1TreeView](#).

A tree contains one or more elements where each element is a node. A node can be a parent, child, or leaf node. The description for each type of node is as follows:

- Parent node is a node that contains other nodes.
- Child node is a node that is contained by another node.
- Leaf node is a node that does not contain child nodes.

Like the classic tree structure, the [C1TreeView](#) contains one or more nodes that consist of parent, child, and leaf nodes. The parent, child, and leaf nodes are referred as **C1TreeViewNodes**. The following image illustrates the nodes and structure of the [C1TreeView](#) control.



[C1TreeView](#) is drawn like an inverted tree where the root appears first. The [C1TreeView](#) can contain one or more root nodes. If a node has child nodes it can be collapsed or expanded. The [ShowExpandCollapse](#) property allows nodes to be expanded or collapsed when set to true. Each node can have text and an image associated with it, may be edited, selected, or display check

boxes depending upon the property settings for the [C1TreeView](#) and [C1TreeNode](#) objects. The tree node can be rendered as a hyperlink and have a URL associated with it.

See Also

[TreeView Creation](#)

TreeView Creation

C1TreeNodes can be defined on your page or user control by using any of the following methods:

- Static creation using declarative syntax
- Dynamic creation using a constructor to create new instances of the [C1TreeNode](#) class.
- Data source creation through binding [C1TreeView](#) to a SiteMapDataSource, XMLDataSource, or an AccessDataSource.

See Also

[Static Creation](#)

[Dynamic Creation](#)

Static Creation

Each node in the Tree is represented by a name/value pair, defined by the text and value properties of `TreeNode`, respectively. The text of a node is rendered, whereas the value of a node is not rendered and is typically used as additional data for handling postback events.

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

You can use the **TreeView Designer Form** designer to build the treeview system or you can use declarative syntax in the .aspx file to specify the nodes.

To display static **C1TreeNode**s using the designer, open the **TreeView Designer Form** and add **C1TreeNode**s to the parent. The properties for each [C1TreeNode](#) can be modified directly in the designer. For more information about the menu designer, see [TreeView Designer Form](#).

To display static **C1TreeNode**s using declarative syntax, first nest opening and closing `<Nodes>` tags between opening and closing tags of the [C1TreeView](#) control. Next, create the treeview structure by nesting `<asp:C1TreeNode>` elements between opening and closing `<Nodes>` tags. Each `<asp:C1TreeNode>` element represents a node in the control and maps to a [C1TreeNode](#) object.

Declarative syntax can be used to define the **C1TreeNode**s inline on your page.

For example:

Syntax	Copy Code
<pre> <cc1:C1TreeView ID="C1TreeView1" runat="server" AllowSorting="False" AutoCollapse="False" VisualStyle="Default" VisualStylePath="~/C1WebControls/C1TreeView/VisualStyles"> <Nodes> <cc1:C1TreeNode runat="server" Expanded="False" Text="C1TreeNode"> <Nodes> <cc1:C1TreeNode runat="server" Expanded="False" Text="C1TreeNode"> </cc1:C1TreeNode> <cc1:C1TreeNode runat="server" Expanded="False" Text="C1TreeNode"> </cc1:C1TreeNode> <cc1:C1TreeNode runat="server" Expanded="False" Text="C1TreeNode"> </cc1:C1TreeNode> </Nodes> </cc1:C1TreeNode> </Nodes> </cc1:C1TreeView> </pre>	

See Also

[Dynamic Creation](#)

Dynamic Creation

Dynamic treeviews can be created on the server side or client side. When creating dynamic treeview on the server side, use a constructor to dynamically create a new instance of the [C1TreeNode](#) class. For client-side, the [CreateInstance](#) constructor can be used to dynamically create a new instance of the [C1TreeView](#) control. For example the follow script creates a new **C1TreeView** control on the client side:

```
var aTreeView = C1.Web.C1TreeView.createInstance ();
```

```
document.body.appendChild(aTreeView.element);
```

[C1TreeView](#) or [C1TreeNode](#) constructors can be used to create a new instance of the [C1TreeView](#) or [C1TreeNode](#) class. Once the nodes are created, they can be added to the Node collection of a new node or treeview.

For example:

To write the code in Visual Basic:

Visual Basic	Copy Code
<pre>Protected Sub Page_Load(ByVal sender As Object, ByVal e As EventArgs) 'create an instance of the class Dim treeView As New C1TreeView() Placeholder1.Controls.Add(treeView) If Not Page.IsPostBack Then Dim P As New C1TreeNode() P.Text = "Products" P.Value = "PS" P.Expanded = True treeView.Nodes.Add(P) Dim Pr1 As New C1TreeNode() Pr1.Text = "Product 1" Pr1.Value = "Pr1" Pr1.Expanded = True P.Nodes.Add(Pr1)</pre>	

```
Dim Oview1 As New C1TreeNode()  
    Oview1.Text = "Overview"  
    Oview1.Value = "Oview1"  
    Pr1.Nodes.Add(Oview1)
```

```
Dim Down1 As New C1TreeNode()  
    Down1.Text = "Downloads"  
    Down1.Value = "Down1"  
    Pr1.Nodes.Add(Down1)
```

```
Dim Supp1 As New C1TreeNode()  
    Supp1.Text = "Support"  
    Supp1.Value = "Supp1"  
    Pr1.Nodes.Add(Supp1)
```

```
Dim Pr2 As New C1TreeNode()  
    Pr2.Text = "Products 2"  
    Pr2.Value = "Pr2"  
    Pr2.Expanded = True  
    P.Nodes.Add(Pr2)
```

```
Dim Oview2 As New C1TreeNode()  
    Oview2.Text = "Overview"  
    Oview2.Value = "Oview2"  
    Pr2.Nodes.Add(Oview2)
```

```
Dim Down2 As New C1TreeNode()  
    Down2.Text = "Downloads"  
    Down2.Value = "Down2"  
    Pr2.Nodes.Add(Down2)
```

```
Dim Supp2 As New C1TreeNode()  
    Supp2.Text = "Support"
```



```

    Supp2.Value = "Supp2"
    Pr2.Nodes.Add(Supp2)

End If
End Sub

```

To write the code in C#:

C#	Copy Code
<pre> protected void Page_Load(object sender, EventArgs e) { //create an instance of the class C1TreeView treeView = new C1TreeView(); Placeholder1.Controls.Add(treeView); if (!Page.IsPostBack) { C1TreeNode P = new C1TreeNode(); P.Text = "Products"; P.Value = "PS"; P.Expanded = true; treeView.Nodes.Add(P); C1TreeNode Pr1 = new C1TreeNode(); Pr1.Text = "Product 1"; Pr1.Value = "Pr1"; Pr1.Expanded = true; P.Nodes.Add(Pr1); C1TreeNode Oview1 = new C1TreeNode(); Oview1.Text = "Overview"; Oview1.Value = "Oview1"; Pr1.Nodes.Add(Oview1); </pre>	

```
C1TreeNode Down1 = new C1TreeNode();
    Down1.Text = "Downloads";
    Down1.Value = "Down1";
    Pr1.Nodes.Add(Down1);

C1TreeNode Supp1 = new C1TreeNode();
    Supp1.Text = "Support";
    Supp1.Value = "Supp1";
    Pr1.Nodes.Add(Supp1);

C1TreeNode Pr2 = new C1TreeNode();
    Pr2.Text = "Products 2";
    Pr2.Value = "Pr2";
    Pr2.Expanded = true;
    P.Nodes.Add(Pr2);

C1TreeNode Oview2 = new C1TreeNode();
    Oview2.Text = "Overview";
    Oview2.Value = "Oview2";
    Pr2.Nodes.Add(Oview2);

C1TreeNode Down2 = new C1TreeNode();
    Down2.Text = "Downloads";
    Down2.Value = "Down2";
    Pr2.Nodes.Add(Down2);

C1TreeNode Supp2 = new C1TreeNode();
    Supp2.Text = "Support";
    Supp2.Value = "Supp2";
    Pr2.Nodes.Add(Supp2);
}
}
```

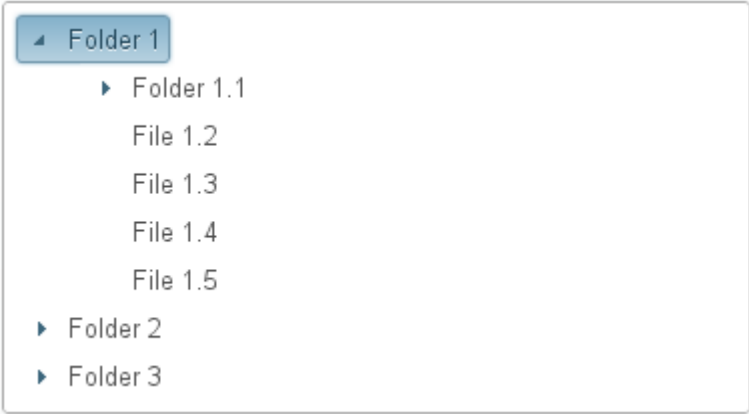
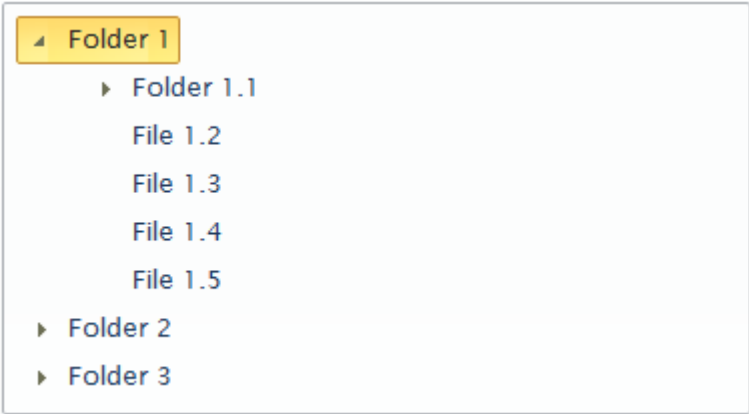
See Also

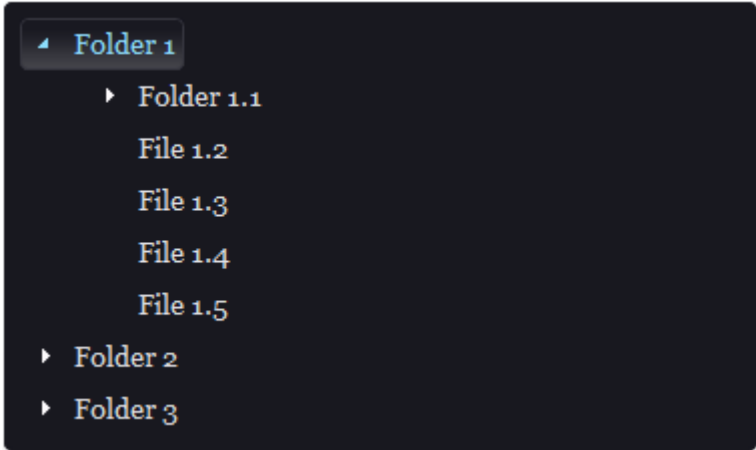

[Appearance and Behavior](#)

Appearance and Behavior

Themes

The [C1TreeView](#) control contains five built-in themes. When one of these themes is selected, all other ASP.NET Web Forms studio controls on the page will be skinned accordingly. The themes will appear on the [C1TreeView](#) control as follows:

Aristo	 The Aristo theme C1TreeView control shows a tree structure. The root node 'Folder 1' is highlighted with a blue background and a small blue triangle icon. It has three child nodes: 'Folder 1.1', 'Folder 2', and 'Folder 3'. 'Folder 1.1' is expanded, showing five file nodes: 'File 1.2', 'File 1.3', 'File 1.4', and 'File 1.5'. All nodes are in a light blue color scheme.
Cobalt	 The Cobalt theme C1TreeView control shows a tree structure. The root node 'Folder 1' is highlighted with a yellow background and a small yellow triangle icon. It has three child nodes: 'Folder 1.1', 'Folder 2', and 'Folder 3'. 'Folder 1.1' is expanded, showing five file nodes: 'File 1.2', 'File 1.3', 'File 1.4', and 'File 1.5'. All nodes are in a light yellow color scheme.

Midnight	 <p>A screenshot of a TreeView control using the 'Midnight' theme. The control is displayed on a dark, almost black background. The root node is 'Folder 1', which is expanded to show its sub-nodes: 'Folder 1.1', 'File 1.2', 'File 1.3', 'File 1.4', 'File 1.5', 'Folder 2', and 'Folder 3'. The text is light blue/white, and the expanded folder has a slight glow.</p>
Rocket	 <p>A screenshot of a TreeView control using the 'Rocket' theme. The control is displayed on a light gray background. The root node is 'Folder 1', which is expanded to show its sub-nodes: 'Folder 1.1', 'File 1.2', 'File 1.3', 'File 1.4', 'File 1.5', 'Folder 2', and 'Folder 3'. The text is dark gray, and the expanded folder has a slight shadow.</p>
Sterling	 <p>A screenshot of a TreeView control using the 'Sterling' theme. The control is displayed on a light gray background. The root node is 'Folder 1', which is expanded to show its sub-nodes: 'Folder 1.1', 'File 1.2', 'File 1.3', 'File 1.4', 'File 1.5', 'Folder 2', and 'Folder 3'. The text is dark gray, and the expanded folder has a slight shadow.</p>

To set the theme of the [C1TreeView](#) control, simply set its **Theme** property to one of the built-in themes.

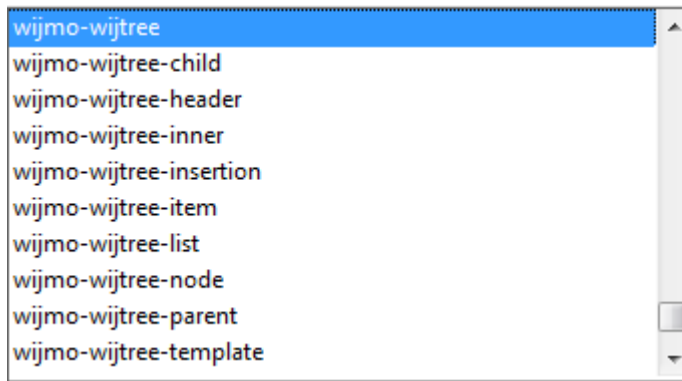
CSS Selectors

You can style many [C1TreeView](#) 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 [C1TreeView](#) control in your project and view the drop-down list next to the **CssClass** property in the Visual Studio Properties window.

[C1TreeView](#) **CSS** selectors begin with `wijmo-wijtree`:

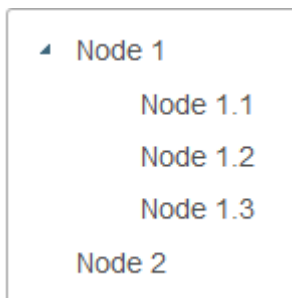


See Also

[Check Boxes](#)

Check Boxes

You can display check boxes next to each [C1TreeNode](#) when [ShowCheckBoxes](#) is set to true. When the check boxes are enabled for the [C1TreeView](#) you can use the **NodeCheckChanged** to create an action whenever the status of a check box changes between posts. If you want to respond immediately to changes in the check boxes on the client without postback you can use the [OnClientNodeCheckChanged](#) server-side event property.



Tri-State Checkboxes

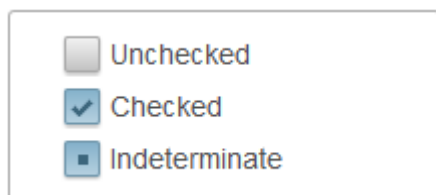
When the [ShowCheckBoxes](#) and **AllowTriState** properties are set to true you can use three types of checkbox states that appear next to the [C1TreeNode](#).

The following table describes the three check box states and how it visually affects each checkbox next to the **C1TreeViewNodes**.

The property [Checked](#) specifies the check state of [C1TreeNode](#).

Checkbox State	Description
Indeterminate	A dark shaded gray box appears in the parent node when only a few of the child nodes are selected.
Checked	A checkmark appears in the parent node when all of its child nodes are selected.
Unchecked	An empty checkbox appears in the parent node when none of the parent node or child nodes are selected.

The following image displays each check box state for the [C1TreeView](#) control: indeterminate, checked, and unchecked.



Tri-State's Effect on Child Nodes

If [AllowTriState](#) is set to True and a node of [C1TreeView](#) has child nodes, its [CheckState](#) is determined by the **CheckState** of its children. There are three cases which are as follows:

- **Case 1**

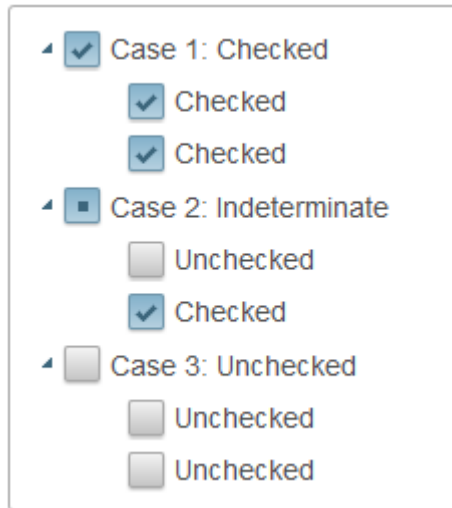
All of the [Checked](#) properties of child nodes are set to **True**, in which case the parent node's [Checked](#) property would be True and the [CheckState](#) is set to **Checked** automatically.

- **Case 2**

Some of the child nodes' [Checked](#) properties are set to True, in which case the parent node's [Checked](#) property will be **True**, but the [CheckState](#) property will be **Indeterminate**.

- **Case 3**

All child nodes' [Checked](#) properties are set to **False**, in which case the parent node's [Checked](#) property will be **False**, and the [CheckState](#) property will be **UnChecked**.




See Also

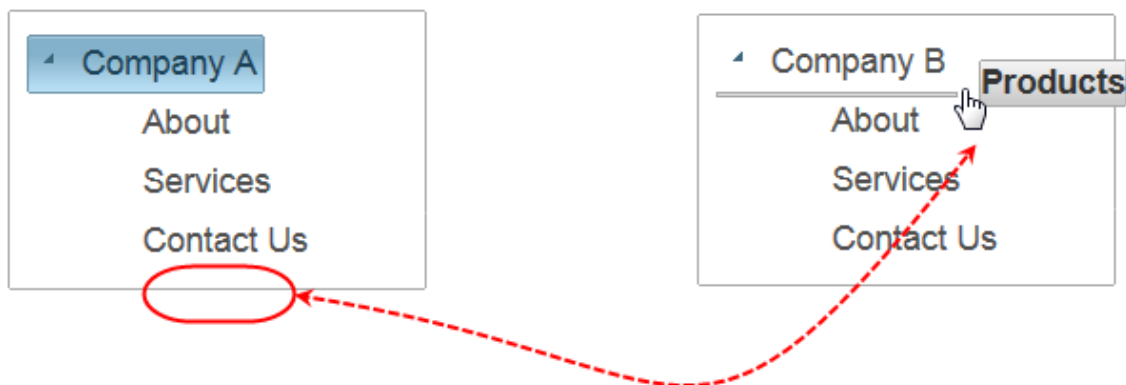
[Drag and Drop Nodes](#)

Drag and Drop Nodes

You can drag-and-drop **C1TreeViewNodes** on nodes, in between nodes, or from one tree to another tree when the [AllowDrag](#) and [AllowDrop](#) properties for C1TreeView control are set to **True**. You can also set the [AllowDrag](#) and [AllowDrop](#) properties for each C1TreeViewNode individually.

 **Note:** **AllowDrag** and **AllowDrop** properties set for a particular **C1TreeViewNode** are given preference over the **AllowDrag** and **AllowDrop** properties set for the whole **C1TreeView** control.

The following image shows a [C1TreeViewNode](#) being dragged from one [C1TreeView](#) to another [C1TreeView](#). A vertical gray line is used as a visual cue to show you where the [C1TreeViewNode](#) is going to be dropped.



When a [C1TreeViewNode](#) indicates a dropped node, it generates a **NodeDropped** server-side event. The event handler for the **NodeDropped** can perform a specific action by locating the dropped node. If you want to respond immediately when a node is being **dragged** or when its

dropped on the client without postback you can use the [OnClientNodeDragStarted](#), [OnClientNodeDragging](#), and [OnClientNodeDropped](#) server-side property events.

See Also

[Load on Demand](#)

Load on Demand

If you have a tree that contains many nodes and you only want vital information to be sent to the server you can set the [LoadOnDemand](#) and [AutoPostBack](#) properties to **True**.

See Also

[Node Selection](#)

Node Selection

When you click on a node at run time it is automatically marked as selected. Clicking a node will raise the [SelectedNodesChanged](#) event to provide custom functionality. To have the nodes marked as selected without clicking them you can enable the [Selected](#) property. Multiple nodes can be selected at one time by holding down the control key while mouse clicking multiple nodes. To unselect a node, click on it again. The nodes are marked as selected in the following [C1TreeView](#):



See Also

[Node Navigation](#)

Node Navigation

[C1TreeView](#) supports mouse and keyboard navigation.

Navigating C1TreeViewNodes using the mouse

The following table describes the actions and corresponding mouse commands when navigating through the **C1TreeViewNodes**:

Action	Mouse Command
Expand a node	Click on the plus sign at the left of the node's name.
Collapse a node	Click on the minus sign at the left of the node's name.

Select a node	Click on the node's name.
---------------	---------------------------

Navigating C1TreeViewNodes using the keyboard

The following table describes the actions and their associated keys to use when navigating through **C1TreeViewNodes**:

Action	Keyboard Command
Expand a node	+ KEY
Collapse a node	- KEY
Move up a node	UP ARROW KEY
Move down a node	DOWN ARROW KEY
Select multiple nodes	MOUSE + CTRL KEY
Shift focus to the next control	TAB KEY

See Also

[Task-Based Help](#)

Task-Based Help

The task-based help section assumes that you are familiar with programming in the Visual Studio ASP.NET environment, and know how to use the [C1TreeView](#) control in general. Each topic provides a solution for specific tasks using the [C1TreeView](#) control. Each task-based help topic also assumes that you have created a new ASP.NET project.

See Also

[Creating and Configuring Check Box Nodes](#)

[Working with Themes](#)

[Working with CSS Selectors](#)

[Adding a Top-Level Node](#)

[Adding a Child Node](#)

[Adding a Child Node with a ButtonClick Event](#)

[Populating C1TreeView with a Site Map](#)

[Populating C1TreeView with XML](#)

[Populating C1TreeView Dynamically](#)

[Saving and Loading a C1TreeView from XML](#)

[Setting the Auto Collapse Property](#)

[Setting C1TreeView to Open on Hover](#)

[Setting C1TreeView Node Icons](#)

[Setting C1TreeView Properties to Allow Drag-and-Drop Behaviors](#)

Creating and Configuring Check Box Nodes

This section contains several tasks that will help you create and configure check boxes for **C1TreeViewNodes**. For more information about check boxes, see [Check Boxes](#).

See Also

[Creating Node Check Boxes](#)

[Preventing Child Nodes from Being Automatically Checked](#)

Creating Node Check Boxes

To create a [C1TreeView](#) filled with check box nodes, simply set the C1TreeView's [ShowCheckBoxes](#) property to True and all of the nodes contained within the control will adopt check boxes.

In Design View

Complete the following steps:

1. Click the smart tag to open the **C1TreeView Tasks** menu. Select **Edit TreeView**.

The **C1TreeView Designer Form** dialog box opens.

2. In the designer treeview, select the C1TreeView control (**C1TreeView1** by default).
3. Navigate to the properties grid, locate the [ShowCheckBoxes](#) property, and set it to **True**.

In Source View

Add [ShowCheckBoxes](#) = True to the <cc1:C1TreeView> tag so that the markup resembles the following:

```
<cc1:C1TreeView ID="C1TreeView1" runat="server" ShowCheckBoxes="True">
```

In Code

Add the following code snippet to the **Page_Load** event:

To write the code in Visual Basic:

```
C1TreeView1.ShowCheckBoxes="True"
```

To write the code in C#:

```
C1TreeView1.ShowCheckBoxes="True";
```

See Also

[Preventing Child Nodes from Being Automatically Checked](#)

Preventing Child Nodes from Being Automatically Checked

When check boxes are utilized in a [C1TreeView](#), clicking a parent node will automatically check all of its child nodes as well. To prevent this, you can set the **C1TreeView**'s [AutoCheckNodes](#) property to **False**.

In this tutorial, you will create a [C1TreeView](#) control with one parent node and two child nodes. You will change the nodes to check boxes and then set the [AutoCheckNodes](#) property to **False** so the child nodes' [Checked](#) property settings will be independent of its parent node's [Checked](#) property setting.

Complete the following:

1. Click the **Source** tab to switch to Source view.
2. Add the following markup between the <cc1:C1TreeView> tags:

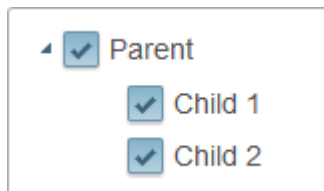
Markup	Copy Code
<pre><Nodes> <cc1:C1TreeViewNode runat="server" Text="Parent"> <Nodes> <cc1:C1TreeViewNode runat="server" Text="Child 1"> </cc1:C1TreeViewNode></pre>	

```

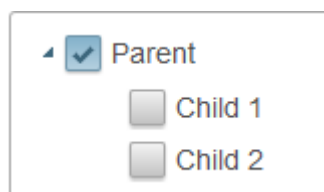
        <cc1:C1TreeNode runat="server" Text="Child 2">
        </cc1:C1TreeNode>
    </Nodes>
</cc1:C1TreeNode>
</Nodes>


```

3. This markup creates a parent node with two child nodes.
4. Enable the check boxes by adding `ShowCheckBoxes="True"` to the `<cc1:C1TreeView>` tag.
5. Press F5 to run the project and complete the following to see how the check boxes behave by default:
 - a. Expand the **Parent** node to reveal the TreeView hierarchy.
 - b. Click the **Parent** check box and observe that the child nodes are automatically selected.



- c. Terminate the program.
6. Click **Design** tab to enter Design view.
7. In the Properties window, select **C1TreeView1** from the drop-down list and then set the `AutoCheckNodes` property to **False**.
8. Press F5 to run the project and complete the following to see how the check boxes behave when the nodes are no longer auto-checked:
 - a. Expand the **Parent** node to reveal the TreeView hierarchy.
 - b. Check the **Parent** check box and observe that the child tabs don't get checked automatically.



 If you check the **Child 1** check box and leave the **Child 2** check box, you'll notice that the **Parent** check box contains an "indeterminate" mark. If you want to prevent this from happening, set the `C1TreeView` control's `AllowTriState` property to **False**.

See Also

[Working with Themes](#)

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](#)

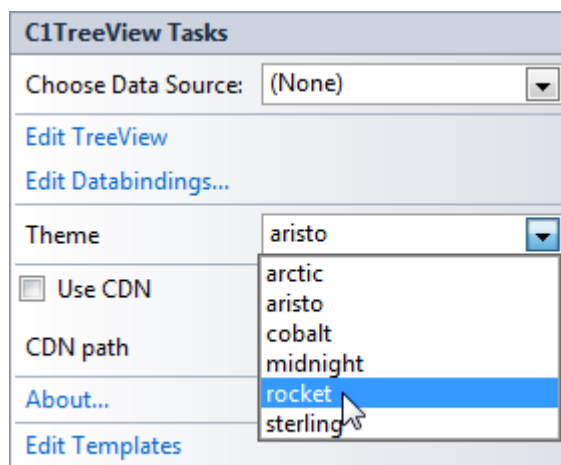
Using a Built-In Theme

A [C1TreeView](#) 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 [C1TreeView Themes](#).

Changing the Theme in Design View

Complete the following steps:

1. Click the [C1TreeView](#) smart tag  to open the **C1TreeView 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 C1TreeView control.

Changing the Theme in Source View

To change the theme of your [C1TreeView](#) in Source view, add `VisualStyle="rocket"` to the `<cc1:C1TreeView>` tag so that it resembles the following:

```
<cc1:C1TreeView ID="C1TreeView1" runat="server" Theme="rocket"/>
```

Changing the Theme in Code

Complete the following steps:

1. Import the following namespace into your project:

To write the code in Visual Basic:

Visual Basic	Copy Code
Imports C1.Web.Wijmo.Controls	

To write the code in C#:

C#	Copy Code
using C1.Web.Wijmo.Controls;	

2. Add the following code, which sets the **Theme** property, to the **Page_Load** event:

To write the code in Visual Basic:

Visual Basic	Copy Code
C1TreeView1.Theme = "rocket"	

To write the code in C#:

C#	Copy Code
C1TreeView1.Theme = "rocket";	

3. Run the program.

✓ This topic illustrates the following:

The following image shows a **C1TreeView** control with the **rocket** theme:



See Also

[Using a Custom Theme](#)

Using a Custom Theme

TreeView 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 [C1TreeView](#) in the following examples.

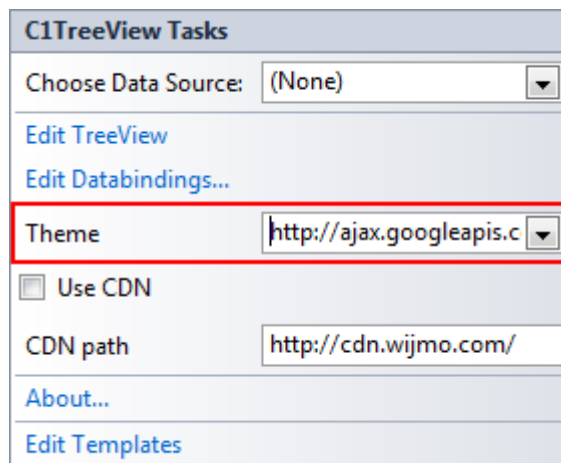
Using ThemeRoller for Visual Studio

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

Using a CDN URL

Complete the following steps:

1. Click the [C1TreeView](#) smart tag to open the **C1TreeView Tasks** menu.
2. In the **Theme** property, enter a CDN URL to specify the theme; CDN URLs can be found at <http://blog.jqueryui.com/2012/04/jquery-ui-1-8-19/>. In this example, we'll use the *le-frog* theme: <http://ajax.aspnetcdn.com/ajax/jquery.ui/1.8.19/themes/le-frog/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.

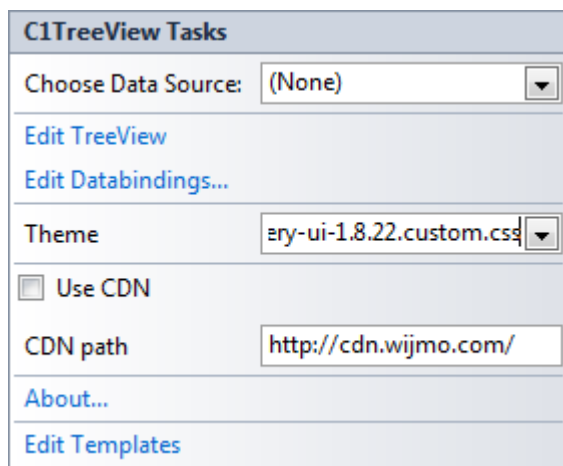
3. Run the project and notice the theme is applied to [C1TreeView](#).



Using jQuery ThemeRoller

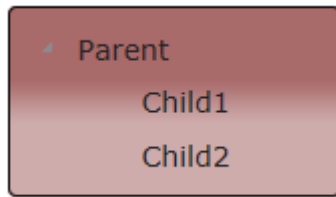
Complete the following steps:

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 **jquery-ui-1.8.22.custom.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 C1TreeView smart tag to open the **Tasks** menu.
7. In the **Theme** property, enter the path to your custom theme .css; for example, **C:\Users\...\Visual Studio 2010\Projects\TreeViewTest\TreeViewTest\customtheme\css\custom-theme\jquery-ui-1.8.22.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.

8. Run the project and notice the theme is applied to [C1TreeView](#).



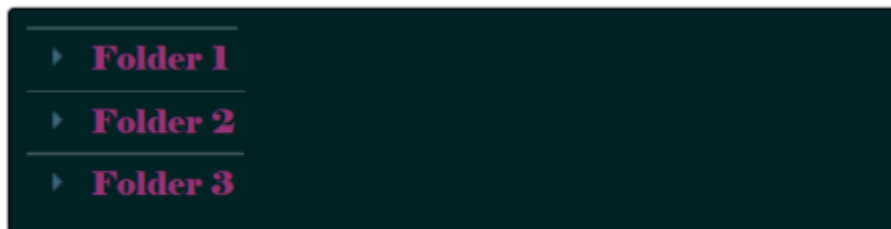
See Also

[Working with CSS Selectors](#)

Working with CSS Selectors

[C1TreeView](#) allows you to fully customize the control's appearance using CSS selectors. This topic will walk you through setting CSS selectors to customize the appearance of [C1TreeView](#).

1. Begin in Design View and navigate to the Properties window for your [C1TreeView](#) control.
2. Locate the **CssClass** property and use the drop-down menu to set it to **wijmo-wijtree**.
3. Switch to your Source View and locate the first set of `<asp:Content>` tags on the page. Add `<style type="text/css"></style>` tags. These tags will allow you to add CSS styling.
4. Add `.wijmo-wijtree { color: #993377; background: #02222; border-color: Black; font-family: Elephant; }` to the tags.
5. Run your program. The [C1TreeView](#) control should appear as in the following image.



See Also

[Adding a Top-Level Node](#)

Adding a Top-Level Node

This topic illustrates how to add a top-level node to a [C1TreeView](#) 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 **C1TreeView Tasks** menu. Select **Edit TreeView**.

The **C1TreeView Designer Form** dialog box opens.

- Click the **Add Child Item** button  to add a C1TreeNode to the C1TreeView control.
- Click **OK** to close the **C1TreeView Design Form** dialog box.

In Source View

Add the following markup between the <cc1:C1TreeView> tags:

Markup	Copy Code
<pre><cc1:C1TreeNode ID= "Node1" runat="server" Text="Node1"> </cc1:C1TreeNode></pre>	

In Code View

Complete the following steps:

- Import the following namespace into your project:

To write the code in Visual Basic:

Visual Basic	Copy Code
<pre>Imports C1.Web.Wijmo.Controls.C1TreeView</pre>	

To write the code in C#:

C#	Copy Code
<pre>using C1.Web.Wijmo.Controls.C1TreeView;</pre>	

- Add the following code to the **Page_Load** event:

To write the code in Visual Basic:

Visual Basic	Copy Code
<pre>Dim TreeViewNode1 As New C1TreeNode() C1TreeViewNode1.Text = "C1TreeViewNode1" C1TreeView1.Nodes.Add(C1TreeViewNode1)</pre>	

To write the code in C#:

C#	Copy Code
<pre>C1TreeNode TreeViewNode1 = new C1TreeNode(); C1TreeNode1.Text = "C1TreeNode1"; C1TreeView1.Nodes.Add(C1TreeNode1);</pre>	

3. Run the program.

See Also

[Adding a Child Node](#)

Adding a Child Node

This topic illustrates how to add a child node to a [C1TreeNode](#) control in Design view, in Source view, and in code. This topic assumes that you have completed [Adding a Top-Level Node to a TreeView](#).

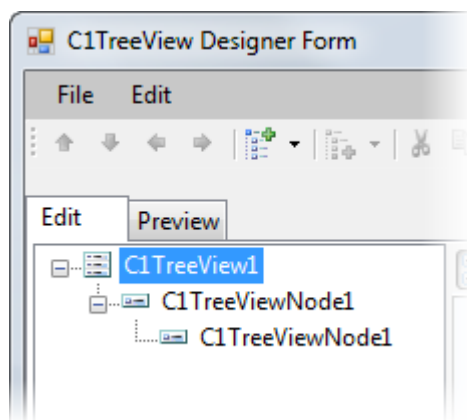
In Design View

Complete the following steps:

1. Click the smart tag to open the **C1TreeView Tasks** menu. Select **Edit TreeView**.

The **C1TreeView Designer Form** dialog box opens.

2. Select the node you wish to add the child node to.
3. Click the **Add Child Item** button to add the child node to the node you select. The treeview of the designer form will resemble the following:



4. Click **OK** to close the **C1TreeView Design Form** dialog box.

In Source View

Add the following markup between the `<cc1:C1TreeNode>` tags of the node to which you wish to add the child node:

Markup	Copy Code
<pre><Nodes> <cc1:C1TreeNode ID="Node1" runat="server" Text="Node1"> </cc1:C1TreeNode> </Nodes></pre>	

In Code View

Complete the following steps:

1. Import the following namespace into your project:

To write the code in Visual Basic:

Visual Basic	Copy Code
Imports C1.Web.Wijmo.Controls.C1TreeView	

To write the code in C#:

C#	Copy Code
using C1.Web.Wijmo.Controls.C1TreeView;	

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

To write the code in Visual Basic:

Visual Basic	Copy Code
<pre>' Create first node and add it to the C1TreeView. Dim C1TreeNode1 As New C1TreeNode() C1TreeNode1.Text = "C1TreeNode1" C1TreeView1.Nodes.Add(C1TreeNode1) ' Create the child node and add it to C1TreeNode1</pre>	

```
Dim C1TreeNode2 As New C1TreeNode()  
C1TreeNode2.Text = "C1TreeNode1"  
C1TreeNode1.Nodes.Add(C1TreeNode2)
```

To write the code in C#:

C#	Copy Code
<pre>// Create first node and add it to the C1TreeView. C1TreeNode C1TreeNode1 = new C1TreeNode(); C1TreeNode1.Text = "C1TreeNode1"; C1TreeView1.Nodes.Add(C1TreeNode1); // Create the child node and add it to C1TreeNode1 C1TreeNode C1TreeNode2 = new C1TreeNode(); C1TreeNode2.Text = "C1TreeNode2"; C1TreeNode1.Nodes.Add(C1TreeNode2);</pre>	

3. Run the program.

See Also

[Adding a Child Node with a ButtonClick Event](#)

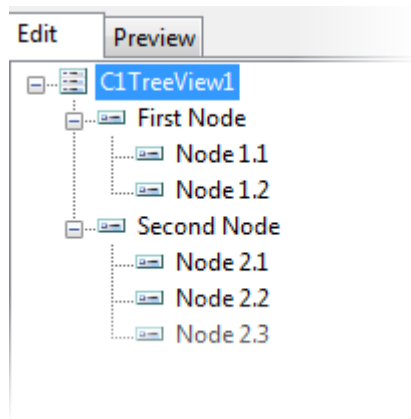
[Adding a Child Node with a ButtonClick Event](#)

Adding a Child Node with a ButtonClick Event

This topic illustrates adding a Child Node with a **ButtonClick()** event. To complete this Help, you will create a **C1TreeView** with child nodes, add a button to the application, and call the **ButtonClick()** event with jQuery syntax.

Complete the following steps:

1. Begin in Design View and add a **C1TreeView** control to your application.
2. Click the smart tag to open the **C1TreeView Tasks** menu. Select **Edit TreeView**.
3. Create a TreeView structure that resembles the following image:



4. Add a general **Button** control to the application and then switch to Source View.
5. Locate the `<asp:Button>` tag and add `OnClickClientClick="buttonClick(); return false;"` to the tag to allow the button to call the jQuery script you will add. The markup should resemble the following:
- 6.
- 7.
- 8.
9. Add the following script above the `<asp:Button>` tag. This will call the **ButtonClick()** event.

Script	Copy Code
<pre> <script type="text/javascript"> function buttonClick() { var nodes = \$("#<%= C1TreeView1.ClientID %>").c1treeview("getSelectedNodes"); //Find all selected nodes (user can use Ctrl/Shift to select multiple) //Add a new node to the first selected node nodes[0].element.c1treeviewnode("add", { text: "Test User 1", value: "user" }); } </script> </pre>	

10. Press F5 to run your application. Click the button to add a new node to the Tree.

See Also

[Populating C1TreeView with a Site Map](#)

Populating C1TreeView with a Site Map

This lesson illustrates how to populate a [C1TreeView](#) with site map data.

To create the Site Map and bind it to the [C1TreeView](#) 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:

Markup	Copy Code
<pre><?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></pre>	

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

Markup	Copy Code
<pre><?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> </siteMap></pre>	

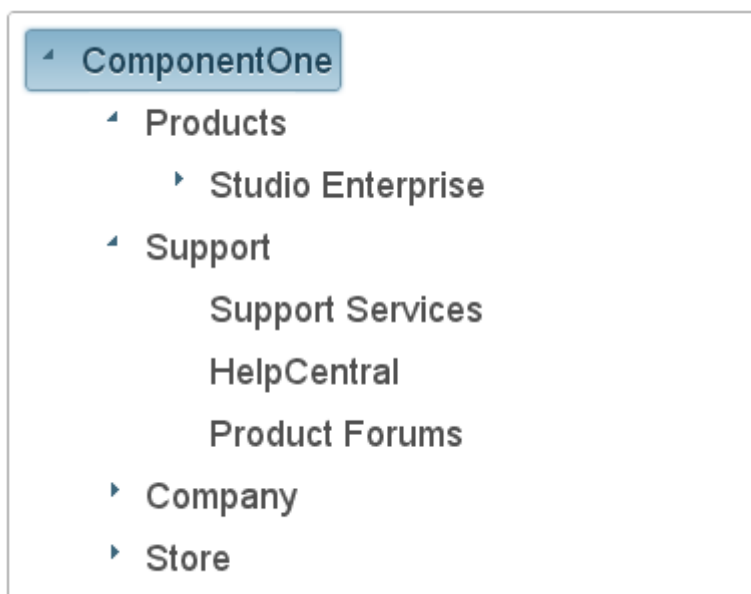
```
<siteMapNode title="Support">
  <siteMapNode title="Support Services" />
  <siteMapNode title="HelpCentral" />
  <siteMapNode title="Product Forums" />
</siteMapNode>
<siteMapNode title="Company">
  <siteMapNode title="About Us" />
  <siteMapNode title="Partners" />
  <siteMapNode title="Contact 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 **C1Treeview** 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.

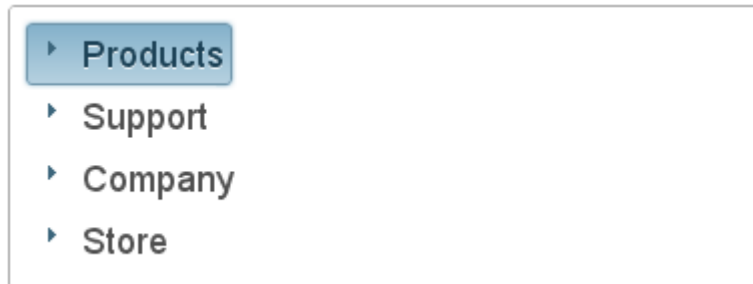
6. Press F5 to run the project and observe the following:

The data from the **Web.sitemap** file is reflected in the **C1TreeView** 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 [C1TreeView](#).

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.



See Also

[Populating C1TreeView with XML](#)

Populating C1TreeView with XML

This tutorial teaches you how installed templates, add the XML Data Source component to the Web site, assign it to the [C1TreeView](#) control, and then set the binding for the [C1TreeView](#).

Complete the following steps:

1. From the Toolbox, double-click the [C1TreeView](#) icon to add the control to your project.
2. Create and prepare the XML file by completing the following steps:
3. Right-click the **App_Data** in the Solution Explorer and select **Add New Item**. The **Add New Item** dialog box appears.
 - a. Select the XML File and rename it "TreeView_Hierarchy.xml". The XML file opens.
 - b. In the XML view, add the following data to the **TreeViewHierarchy.xml** document:

Markup	Copy Code
<pre><?xml version="1.0" encoding="utf-8" ?> <root> <TreeNode Text="CDs">\ <TreeNode Text="Back to Black"></TreeNode> <TreeNode Text="Frank"></TreeNode> <TreeNode Text="Nevermind"></TreeNode></pre>	

```
<TreeNode Text="In Utero"></TreeNode>
</TreeNode>
<TreeNode Text="Cassette Tapes">
  <TreeNode Text="Bleach"></TreeNode>
  <TreeNode Text="Cheap Thrills"></TreeNode>
  <TreeNode Text="Dangerous"></TreeNode>
  <TreeNode Text="Bad"></TreeNode>
</TreeNode>
<TreeNode Text="Vinyl Records">
  <TreeNode Text="Axis: Bold as Love"></TreeNode>
  <TreeNode Text="Full Circle"></TreeNode>
  <TreeNode Text="Off The Wall"></TreeNode>
  <TreeNode Text="Other Voices"></TreeNode>
</TreeNode>
</root>
```

4. Switch back to the Design view of the **.aspx** page and complete the following steps to create a new data source:
 - a. Click **C1TreeView**'s smart tag to open the **C1TreeView 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.

5. Complete the following steps to configure the data source:
 - a. Click **C1TreeView**'s smart tag to open the **C1TreeView Tasks** menu; click **Configure Data Source**.

The **Configure Data Source** dialog box opens.

- b. In the XPath expression text field, enter "root/TreeNode". This will select all TreeViewNodes that are children of root so that the TreeViewNodes 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 **TreeView_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.
6. Complete the following steps to bind the XML tags to the **C1TreeViewNodes**.
 - a. Click **C1TreeView**'s smart tag to open the **C1TreeView Tasks** menu and click **Edit Databindings**.

The **Bindings Collection Editor** dialog box opens.

- b. Click **Add** to add an empty binding to the project.

- c. Set the binding's properties as follows:
 - Set the **DataMember** property to "TreeNode".
 - Set the **TextField** property to "Text".
 - d. Click **OK** to close the **Bindings Collection Editor**:
7. Press F5 to run the project.

Observe that the data from the **TreeView_Hierarchy.xml** file is reflected in the [C1TreeView](#) control.



See Also

[Populating C1TreeView Dynamically](#)

Populating C1TreeView Dynamically

[C1TreeView](#) allows you to populate the control dynamically. The child nodes are loaded using AJAX when the parent node is expanded.

1. Double-click the [C1TreeView](#) control in the Visual Studio Toolbox to add it to your application.
2. Switch to Source View and add the following markup to your application to add an `<asp:SiteMapDataSource>` control:

- 3.
4. `<asp:SiteMapDataSource ID="SiteMapDataSource" runat="server" ShowStartingNode="False" />`
5. Locate the markup for the [C1TreeView](#) control and edit it to resemble the following:

Markup	Copy Code
<pre><cc1:C1TreeView ID="C1TreeView1" ShowCheckBoxes="true" LoadOnDemand="true" DataSourceID="SiteMapDataSource" ShowExpandCollapse="true" DataBindStartLevel="0" Width="350px" runat="server"> </cc1:C1TreeView></pre>	

6. Right-click your project's name in the Solution Explorer and select **Add | New Item** from the list.
7. Select **Site Map** in the **Add New Item** dialog and click **Add**. A new **Web.sitemap** will be added to your application. The **Web.sitemap** file should open immediately.
8. Add the following markup to the **Web.sitemap** file:

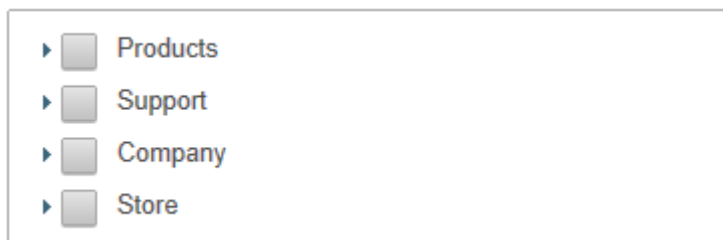
Markup	Copy Code
<pre><siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" > <siteMapNode url="RootNodeUrl" title="Root Node" description="Root Node Description" > <siteMapNode url="" title="Products" description="Products" > <siteMapNode url="" title="Studio Enterprise" description="Studio Enterprise"> <siteMapNode url="" title="Studio for WinForms" description="Studio for WinForms" /> <siteMapNode url="" title="Studio for ASP.NET" description="Studio for ASP.NET" /> <siteMapNode url="" title="Studio for WPF" description="Studio for WPF" /> <siteMapNode url="" title="Studio for Mobile" description="Studio for Mobile" /> <siteMapNode url="" title="Studio for ActiveX" description="Studio for ActiveX" /> <siteMapNode url="" title="Studio for Silverlight" description="Studio for Silverlight" /> </siteMapNode> <siteMapNode url="" title="IntelliSpell" description="IntelliSpell" /> <siteMapNode url="" title="Report Designer Edition"</pre>	

```

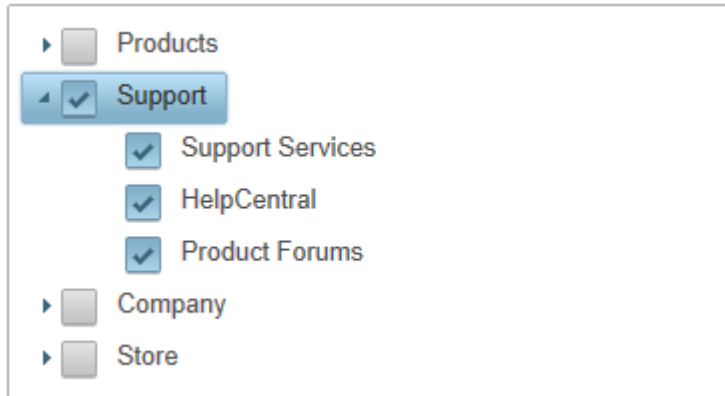
description="Report Designer Edition" />
</siteMapNode>
<siteMapNode url="" title="Support" description="Support" >
  <siteMapNode url="" title="Support Services" description="Support
Services" />
  <siteMapNode url="" title="HelpCentral" description="HelpCentral" />
  <siteMapNode url="" title="Product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

```

9. Run your application; it should resemble the following image:



And when you open one of the nodes, the [C1TreeView](#) control will resemble the following image:



See Also

[Saving and Loading a C1TreeView from XML](#)

Saving and Loading a C1TreeView from XML

The following tasks show you how to save your [C1TreeView](#) control as an .xml file and then load it into your project using the designer.

Save the C1TreeView as XML

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

1. Click **C1TreeView**'s smart tag and select **Edit TreeView** to open the **C1TreeView Designer Form**.
2. Navigate to **File | Save as XML**.
3. Name the .xml file for your [C1TreeView](#) and browse to where you would like to save it.
4. Click **OK** to close the **TreeView Designer Form** dialog box.

Load an Existing XML C1TreeView into your Project

To load the [C1TreeView](#) control you saved as an .xml file into your project:

1. Click **C1TreeView**'s smart tag and select **Edit TreeView** to open the **C1TreeView Designer Form**.
2. Open the **C1TreeView Designer Form**.
3. Navigate to **File | Load From XML** and click open to open the existing .xml file.

Load an Existing XML C1TreeView in Code

To load the [C1TreeView](#) control you saved as an .xml file into your project:

1. Create an XML file for the [C1TreeView](#) structure.
2. Call the [LoadLayout](#) method to load the items, passing in the path to the file:

To write the code in Visual Basic:

Visual Basic	Copy Code
<pre>C1TreeView1.LoadLayout("c:\\Visual Studio 2005\\WebSites\\LoadLayoutEX\\App_Data\\C1TreeViewControl.xml")</pre>	

To write the code in C#:

C#	Copy Code
<pre>C1TreeView1.LoadLayout("c:\\Visual Studio 2005\\WebSites\\LoadLayoutEX\\App_Data\\C1TreeViewControl.xml");</pre>	

3. Press F5 to run the program.

See Also

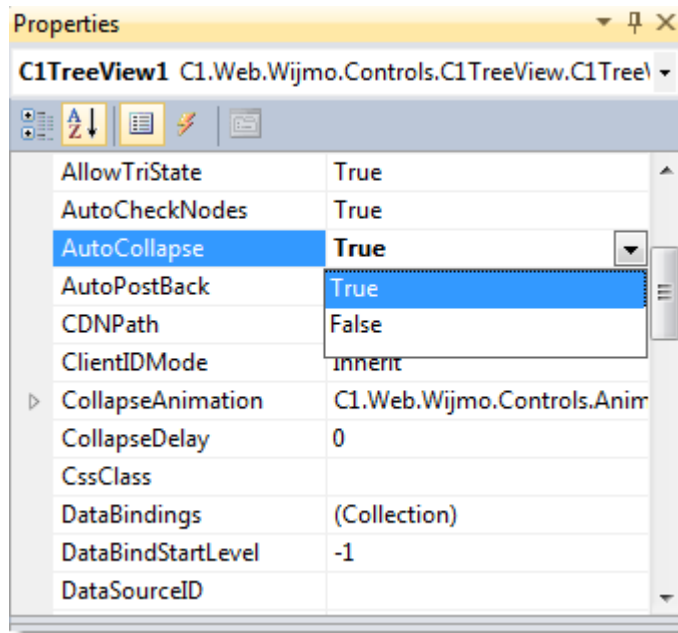
[Setting the Auto Collapse Property](#)

Setting the Auto Collapse Property

Autocollapse causes all expanded nodes to collapse if another node is expanded. This task-based help topic will walk you through setting the [AutoCollapse](#) property in Design View and in Source View.

In Design View

1. In the Design View, navigate to the [C1TreeView](#) Properties window.
2. Locate the [AutoCollapse](#) property and use the drop-down menu to set the property to "true".



3. Press F5 to run your program. Note that when one node is expanded, any other expanded nodes will automatically collapse.

In Source View

1. In the Source View, add the following markup to create the nodes for [C1TreeView](#).

Markup	Copy Code
<pre> <Nodes> <cc1:C1TreeViewNode Text="Folder 1"> <Nodes> <cc1:C1TreeViewNode Text="Folder 1.1"> <Nodes> <cc1:C1TreeViewNode Text="Folder 1.1.1"> </cc1:C1TreeViewNode> <cc1:C1TreeViewNode Text="Folder 1.1.2"> </cc1:C1TreeViewNode> <cc1:C1TreeViewNode Text="Folder 1.1.3"> </cc1:C1TreeViewNode> <cc1:C1TreeViewNode Text="Folder 1.1.4"> </cc1:C1TreeViewNode> </Nodes> </cc1:C1TreeViewNode> </Nodes> </cc1:C1TreeViewNode> <cc1:C1TreeViewNode Text="Folder 1.2"> </cc1:C1TreeViewNode> </pre>	


```
<cc1:C1TreeNode Text="Folder 1.3">
</cc1:C1TreeNode>
<cc1:C1TreeNode Text="Folder 1.4">
</cc1:C1TreeNode>
<cc1:C1TreeNode Text="Folder 1.5">
</cc1:C1TreeNode>
</Nodes>
</cc1:C1TreeNode>
<cc1:C1TreeNode Text="Folder 2">
  <Nodes>
    <cc1:C1TreeNode Text="Folder 2.1">
    </cc1:C1TreeNode>
    <cc1:C1TreeNode Text="Folder 2.2">
    </cc1:C1TreeNode>
    <cc1:C1TreeNode Text="Folder 2.3">
    </cc1:C1TreeNode>
    <cc1:C1TreeNode Text="Folder 2.4">
    </cc1:C1TreeNode>
    <cc1:C1TreeNode Text="Folder 2.5">
    </cc1:C1TreeNode>
  </Nodes>
</cc1:C1TreeNode>
<cc1:C1TreeNode Text="Folder 3">
  <Nodes>
    <cc1:C1TreeNode Text="Folder 3.1">
    </cc1:C1TreeNode>
    <cc1:C1TreeNode Text="Folder 3.2">
    </cc1:C1TreeNode>
    <cc1:C1TreeNode Text="Folder 3.3">
    </cc1:C1TreeNode>
    <cc1:C1TreeNode Text="Folder 3.4">
    </cc1:C1TreeNode>
    <cc1:C1TreeNode Text="Folder 3.5">
    </cc1:C1TreeNode>
  </Nodes>
</cc1:C1TreeNode>
</Nodes>
```

2. Add `Autocollapse="true"` to the `<cc1:C1TreeView>` tags as in the following sample:
- 3.

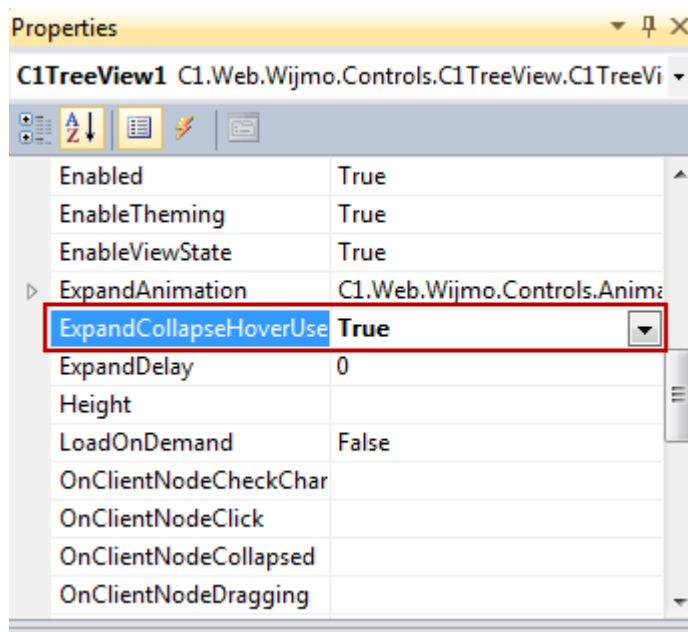
4. `<cc1:C1TreeView ID="C1TreeView1" runat="server" AutoCollapse="true">`
5. Press F5 to run your program. Note that when one node is expanded, any other expanded node will automatically collapse.

Setting C1TreeView to Open on Hover

[C1TreeView](#) supports expanding nodes when they are hovered over as an alternative to expanding on a mouse click. This topic will walk you through setting the [ExpandCollapseHoverUsed](#) property.

In Design View

1. In the Design View, navigate to the C1TreeView Properties window and locate the [ExpandCollapseHoverUsed](#) property.



2. Use the drop-down menu to set the property to "true".
3. Run your project. Note that the nodes will expand when hovered over with the mouse.

In Source View

Add `ExpandCollapseHoverUse="true"` to the `<cc1:C1TreeView>` tags to resemble the following sample:

```
<cc1:C1TreeView ID="C1TreeView1" runat="server" AutoCollapse="True"
```

```
ExpandCollapseHoverUsed="True">
```

See Also

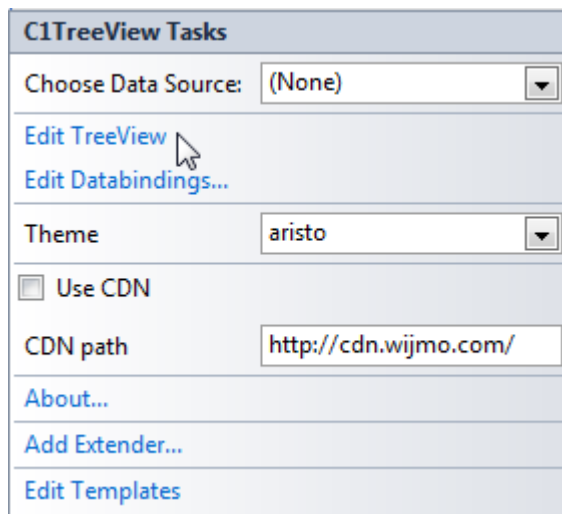
Setting C1TreeView Node Icons

Setting C1TreeView Node Icons

C1TreeView allows you to set node icons. In addition, you can switch icons based on C1TreeView's current state. This topic will walk you through setting the properties to display node icons and to change node icons based on C1TreeView's state.

In Design View

1. Click the **C1TreeView** smart tag  to open the **C1TreeView Tasks Menu**.
2. Click **Edit TreeView** to open the **C1TreeView Designer Form**.



3. Select the first node, Folder 1, to display its properties in the Properties Window.
4. Locate the **CollapsedIconClass** property and set it to **ui-icon-folder-collapsed**.
5. Locate the **ExpandedIconClass** property and set it to **ui-icon-folder-open**.
6. Locate the **ItemIconClass** property and set it to **ui-icon-document**. The Designer Form Properties Window should resemble the following image:

Misc	
(ID)	
CollapsedIconClass	ui-icon-folder-collapsed
DisplayVisible	True
Expanded	False
ExpandedIconClass	ui-icon-folder-open
ItemIconClass	ui-icon-document
Selected	False
Text	Folder 1
Url	

7. Press F5 to run your project. Your **C1TreeView** control should resemble the following image.



In Source View

1. In the Source View, locate the first set of `<cc1:C1TreeNode>` tags, the **Text** property of which should read "Folder 1".
2. Insert the following markup to set the node icons:

Markup	Copy Code
<pre>CollapsedIconClass="ui-icon-folder-collapsed" ExpandedIconClass="ui-icon-folder-open" ItemIconClass="ui-icon-document"</pre>	

3. Your `<cc1:C1TreeNode>` tags should resemble the following sample.

Sample	Copy Code
<pre><cc1:C1TreeNode Text="Folder 1" CollapsedIconClass="ui-icon-folder-collapsed" ExpandedIconClass="ui-icon-folder-open" ItemIconClass="ui-icon-document"></pre>	

4. Locate the `<cc1:C1TreeNode>` tags for "Folder 1.1" and insert the following markup.

Markup	Copy Code

```
CollapsedIconClass="ui-icon-folder-collapsed"
ExpandedIconClass="ui-icon-folder-open"
ItemIconClass="ui-icon-document"
```

5. Your <cc1:C1TreeNode> tags should resemble the following.

Sample	Copy Code
<pre><cc1:C1TreeNode Text="Folder 1.1" CollapsedIconClass="ui-icon-folder-collapsed" ExpandedIconClass="ui-icon-folder-open" ItemIconClass="ui-icon-document"></pre>	

6.
7. Locate the <cc1:C1TreeNode> tags for "Folder 1.1.1" and add ItemIconClass="ui-icon-document" to the tags. Your <cc1:C1TreeNode> tags should resemble the following sample.

Sample	Copy Code
<pre><cc1:C1TreeNode Text="Folder 1.1.1" ItemIconClass="ui-icon-document"></pre>	

8. Press F5 to run your program. Your C1TreeView control should resemble the following image.



See Also

[Setting C1TreeView Properties to Allow Drag-and-Drop Behaviors](#)

Setting C1TreeView Properties to Allow Drag-and-Drop Behaviors

[C1TreeView](#) supports drag-and-drop behavior which allows the end-user to drag and drop nodes to rearrange the tree structure. Drag and drop node behavior can occur within one tree, or between two tree structures. This task-based help will walk you through setting the basic properties that allow drag-and-drop behavior, and the properties that allow end-users to drag and drop nodes between two tree structures.

See Also

[Drag-and-drop Behaviors Within One Tree Structure](#)

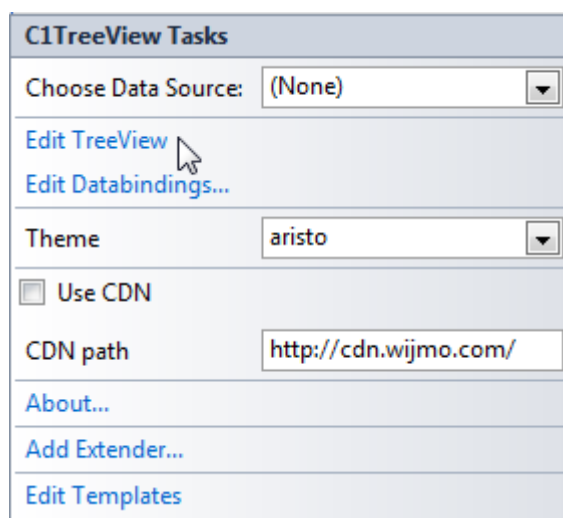
[Drag-and-drop Behaviors Between Two Tree Structures](#)

Drag-and-drop Behaviors Within One Tree Structure

This topic will walk you through setting [C1TreeView](#) properties to allow drag-and-drop behaviors within one tree structure.

In Design View

1. Click the C1TreeView smart tag  to open the **C1TreeView Tasks Menu**.
2. Click **Edit TreeView** to open the **C1TreeView Designer Form**.



3. Locate the **AllowDrag** and **AllowDrop** properties in the Designer Form Properties Window and use the drop-down menus to set both properties to "true".
4. Press F5 to run your program. Note that you are now able to rearrange the nodes.

In Source View

Add the following markup to the first set of `<cc1:C1TreeView>` tags.

Markup	Copy Code
<pre>AllowDrag="True" AllowDrop="True"</pre>	

The `<cc1:C1TreeView>` tags should resemble the following sample.

Sample	Copy Code
<pre><cc1:C1TreeView ID="C1TreeView1" runat="server" AllowDrag="True" AllowDrop="True"></pre>	

Press F5 to run your program. Note that you are now able to rearrange the nodes.

See Also

[Drag-and-drop Behaviors Between Two Tree Structures](#)

Drag-and-drop Behaviors Between Two Tree Structures

This topic will walk you through setting [C1TreeView](#) properties to allow drag-and-drop behaviors between two tree structures.

In Design View

1. In Design View, add two [C1TreeView](#) controls to your project and create several child nodes for each control.
2. Select the first [C1TreeView](#) control and navigate to the Properties window.
3. Locate the [AllowDrag](#) and [AllowDrop](#) properties and use the drop-down menus to set both to "true".
4. Select the second [C1TreeView](#) control and navigate to the Properties window.
5. Locate the [AllowDrop](#) property and use the drop-down menu to set it to "true". This will allow you to drag items from the first [C1TreeView](#) control and drop them into the second [C1TreeView](#) control at run time.
6. Press F5 to run your program. Note that you are able to drag nodes from the first tree and drop them into the second tree.

In Source View

1. Insert the following markup in the second set of <asp:Content> tags to create two [C1TreeView](#) controls.

Markup	Copy Code
<pre> <cc1:C1TreeView ID="C1TreeView1" runat="server"> <Nodes> <cc1:C1TreeNode runat="server" CheckState="Unchecked" NodeIndex="0" Owner="C1TreeView1" StaticKey="C1TreeView1_0" Text="C1TreeViewNode6" TreeView="C1TreeView1"> </cc1:C1TreeNode> <cc1:C1TreeNode runat="server" CheckState="Unchecked" NodeIndex="0" Owner="C1TreeView1" StaticKey="C1TreeView1_1" Text="C1TreeViewNode4" TreeView="C1TreeView1"> <Nodes> <cc1:C1TreeNode runat="server" CheckState="Unchecked" NodeIndex="0" Owner="" StaticKey="C1TreeView1_10" Text="C1TreeViewNode1" TreeView="C1TreeView1"> </cc1:C1TreeNode> <cc1:C1TreeNode runat="server" CheckState="Unchecked" NodeIndex="0" Owner="" StaticKey="C1TreeView1_11" Text="C1TreeViewNode2" TreeView="C1TreeView1"> </cc1:C1TreeNode> <cc1:C1TreeNode runat="server" CheckState="Unchecked" NodeIndex="0" Owner="" StaticKey="C1TreeView1_12" Text="C1TreeViewNode3" TreeView="C1TreeView1"> </cc1:C1TreeNode> </Nodes> </cc1:C1TreeNode> <cc1:C1TreeNode runat="server" CheckState="Unchecked" </pre>	


```

NodeIndex="0"
    Owner="C1TreeView1" StaticKey="C1TreeView1_2"
Text="C1TreeViewNode5"
    TreeView="C1TreeView1">
</cc1:C1TreeViewNode>
<cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Owner="C1TreeView1" StaticKey="C1TreeView1_3"
Text="C1TreeViewNode1"
    TreeView="C1TreeView1">
    <Nodes>
        <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Owner="" StaticKey="C1TreeView1_30"
Text="C1TreeViewNode1"
    TreeView="C1TreeView1">
</cc1:C1TreeViewNode>
<cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Owner="" StaticKey="C1TreeView1_31"
Text="C1TreeViewNode2"
    TreeView="C1TreeView1">
</cc1:C1TreeViewNode>
<cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Owner="" StaticKey="C1TreeView1_32"
Text="C1TreeViewNode3"
    TreeView="C1TreeView1">
</cc1:C1TreeViewNode>
    </Nodes>
</cc1:C1TreeViewNode>
<cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Owner="C1TreeView1" StaticKey="C1TreeView1_4"
Text="C1TreeViewNode2"
    TreeView="C1TreeView1">
</cc1:C1TreeViewNode>
<cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Owner="C1TreeView1" StaticKey="C1TreeView1_5"
Text="C1TreeViewNode3"
    TreeView="C1TreeView1">

```

```

        <Nodes>
            <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
                Owner="" StaticKey="C1TreeView1_50"
Text="C1TreeNode1"
                TreeView="C1TreeView1">
            </cc1:C1TreeNode>
            <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
                Owner="" StaticKey="C1TreeView1_51"
Text="C1TreeNode2"
                TreeView="C1TreeView1">
            </cc1:C1TreeNode>
            <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
                Owner="" StaticKey="C1TreeView1_52"
Text="C1TreeNode3"
                TreeView="C1TreeView1">
            </cc1:C1TreeNode>
        </Nodes>
    </cc1:C1TreeNode>
    <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Owner="C1TreeView1" StaticKey="C1TreeView1_6"
Text="C1TreeNode7"
        TreeView="C1TreeView1">
    </cc1:C1TreeNode>
</Nodes>
</cc1:C1TreeView
<cc1:C1TreeView ID="C1TreeView2" runat="server">
    <Nodes>
        <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
            Text="C1TreeNode6">
        <Nodes>
            <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
                Text="C1TreeNode1">
            </cc1:C1TreeNode>
            <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
                Text="C1TreeNode2">

```

```

        </cc1:C1TreeNode>
    </Nodes>
</cc1:C1TreeNode>
<cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Text="C1TreeNode7">
</cc1:C1TreeNode>
<cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Text="C1TreeNode8">
    <Nodes>
        <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
            Text="C1TreeNode1">
        </cc1:C1TreeNode>
        <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
            Text="C1TreeNode2">
        </cc1:C1TreeNode>
        <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
            Text="C1TreeNode3">
        </cc1:C1TreeNode>
        <cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
            Text="C1TreeNode4">
        </cc1:C1TreeNode>
    </Nodes>
</cc1:C1TreeNode>
<cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Text="C1TreeNode1">
</cc1:C1TreeNode>
<cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Text="C1TreeNode2">
</cc1:C1TreeNode>
<cc1:C1TreeNode runat="server" CheckState="Unchecked"
NodeIndex="0"
    Text="C1TreeNode3">
    <Nodes>
        <cc1:C1TreeNode runat="server" CheckState="Unchecked"

```

```

NodeIndex="0"
        Text="C1TreeViewNode1">
    </cc1:C1TreeViewNode>
    <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Text="C1TreeViewNode2">
    </cc1:C1TreeViewNode>
    <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Text="C1TreeViewNode3">
    </cc1:C1TreeViewNode>
    <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Text="C1TreeViewNode4">
    </cc1:C1TreeViewNode>
    <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Text="C1TreeViewNode5">
    </cc1:C1TreeViewNode>
    <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Text="C1TreeViewNode6">
    </cc1:C1TreeViewNode>
    <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Text="C1TreeViewNode7">
    </cc1:C1TreeViewNode>
    </Nodes>
    </cc1:C1TreeViewNode>
    <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Text="C1TreeViewNode4">
    </cc1:C1TreeViewNode>
    <cc1:C1TreeViewNode runat="server" CheckState="Unchecked"
NodeIndex="0"
        Text="C1TreeViewNode5">
    </cc1:C1TreeViewNode>
    </Nodes>
    </cc1:C1TreeView>

```

2. Locate the first set of <cc1:C1TreeView> tags and add AllowDrag="True" and AllowDrop="True". The tag should resemble the following sample.

- 3.
4.

```
<cc1:C1TreeView ID="C1TreeView1" runat="server" AllowDrag="True"
                AllowDrop="True">
```
5. Locate the second set of `<cc1:C1TreeView>` tags and add `AllowDrop="True"`. The tag should resemble the following sample.
- 6.
7.

```
<cc1:C1TreeView ID="C1TreeView2" runat="server" AllowDrop="True">
```
8. Press F5 to run your program. Note that you are now able to drag nodes from the first tree and drop them into the second tree.

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 **TreeView 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.

TreeView

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

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.

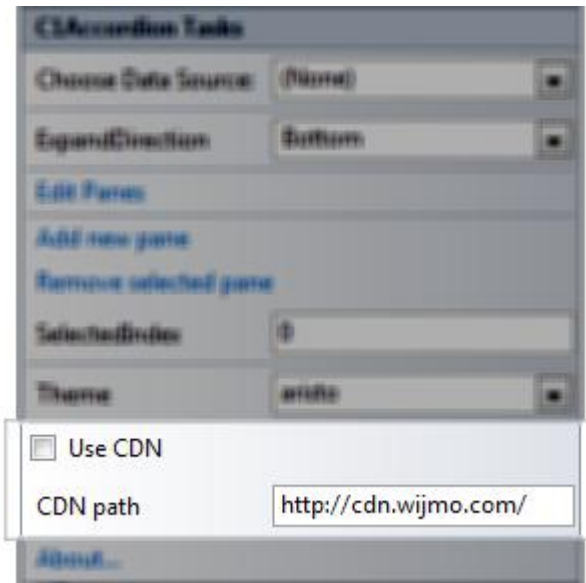
When web browsers load content, they commonly will check to see if they already have a copy of the file cached. By using a CDN, you can benefit from this. If a user had previously visited a site using the same CDN, they will already have a cached version of the files on their machine. Your page will 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:

Markup	Copy Code
<pre> <!--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> </pre>	

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	
	C1TargetControlBase	
	C1TargetControlHelper	
	C1TargetControlHelper<T>	
	WijChartNavigator	

See Also

Reference

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

Classes

C1BaseStateManager

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

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

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

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



















	EnableCombinedJavaScripts	
	Enabled	Overridden.
	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	
	ThemeSwatch	
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	UniqueID	(Inherited from System.Web.UI.Control)
	UseCDN	
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Width	(Inherited from System.Web.UI.WebControls.WebControl)
	WijmoControlMode	
	WijmoCssAdapter	

[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)
 GetPropertyValue<V>	
 GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
 GetScriptDescriptors	
 GetScriptReferences	
 GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
 HasControls	(Inherited from System.Web.UI.Control)
 MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
 RegisterDesignTimeStyleSheet	
 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)
 SetPropertyValue<V>	
 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>	
	GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
	GetScriptDescriptors	
	GetScriptReferences	
	GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
	HasControls	(Inherited from System.Web.UI.Control)
	MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	RegisterDesignTimeStyleSheet	
	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>	
	SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1TargetControlBase Class](#)

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

GetPropertyValue<V> Method

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

nullValue

Type Parameters

V

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

GetScriptDescriptors Method

Syntax

Visual Basic (Declaration)

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

C#

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

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

GetScriptReferences Method

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>	

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

RegisterDesignTimeStyleSheet Method

Syntax

Visual Basic (Declaration)	
<pre>Public Sub RegisterDesignTimeStyleSheet(_ ByVal writer As System.Web.UI.HtmlTextWriter, _ ByVal cssResourceName As System.String _)</pre>	
C#	
<pre>public void RegisterDesignTimeStyleSheet(System.Web.UI.HtmlTextWriter writer, System.string cssResourceName)</pre>	

Parameters

writer

cssResourceName

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

SetPropertyValue<V> Method

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

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	
	CDNPath	
	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	
	EnableCombinedJavaScripts	
	Enabled	Overridden.
	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	
	ThemeSwatch	
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	UniqueID	(Inherited from System.Web.UI.Control)
	UseCDN	
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Width	(Inherited from System.Web.UI.WebControls.WebControl)
	WijmoControlMode	
	WijmoCssAdapter	

[Top](#)

See Also

Reference

[C1TargetControlBase Class](#)

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

CDNDependencyPaths Property

Syntax

Visual Basic (Declaration)	
<pre><C1DescriptionAttribute(Key="C1Base.CDNDependencyPaths", Description="A string array value specifies all css/js references that wijmo control depends on.")> <System.ComponentModel.DefaultValueAttribute()> <System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)></pre>	

```
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Hidden)>
Public Overridable Property CDNDependencyPaths As System.String()
```

C#

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

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

CDNPath Property

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1Base.CDNPath", Description="Content Delivery Network path.")>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Hidden)>
Public Overridable Property CDNPath As System.String
```

C#

```
[C1Description(Key="C1Base.CDNPath", Description="Content Delivery Network path.")]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[C1Category("Behavior")]
```

```
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
public virtual System.string CDNPath {get; set;}
```

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

DisplayVisible Property

Syntax

Visual Basic (Declaration)

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

C#

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

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

EnableCombinedJavaScripts Property

Syntax

Visual Basic (Declaration)

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

C#

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

See Also

Reference

[C1TargetControlBase Class](#)[C1TargetControlBase Members](#)

Enabled Property

Syntax

Visual Basic (Declaration)

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

C#

```
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[C1Description(Key="C1Base.Enabled", Description="Indicates whether control
```

```
is enabled."}]
[C1Category("Behavior")]
public override System.bool Enabled {get; set;}
```

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

Theme Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<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()>
<TypeConverterAttribute("C1.Web.Wijmo.Controls.WijmoThemeNameConverter,
C1.Web.Wijmo.Controls.4, Version=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Hidden)>
Public Overridable Property Theme As System.String
```

C#

```
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[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()]
[TypeConverter("C1.Web.Wijmo.Controls.WijmoThemeNameConverter,
C1.Web.Wijmo.Controls.4, Version=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
```

```

isibility.Hidden)]
public virtual System.string Theme {get; set;}

```

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

ThemeSwatch Property

Syntax

Visual Basic (Declaration)

```

<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Hidden)>
<C1CategoryAttribute("Appearance")>
<TypeConverterAttribute("C1.Web.Wijmo.Controls.WijmoThemeSwatchConverter,
C1.Web.Wijmo.Controls.4, Version=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<C1DescriptionAttribute(Key="C1Base.ThemeSwatch", Description="A value that
indicates the theme swatch of the control, this property only works when
WijmoControlMode property is Mobile.")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
Public Overridable Property ThemeSwatch As System.String

```

C#

```

[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
[C1Category("Appearance")]
[TypeConverter("C1.Web.Wijmo.Controls.WijmoThemeSwatchConverter,
C1.Web.Wijmo.Controls.4, Version=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[C1Description(Key="C1Base.ThemeSwatch", Description="A value that indicates
the theme swatch of the control, this property only works when
WijmoControlMode property is Mobile.")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
public virtual System.string ThemeSwatch {get; set;}

```

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

UseCDN Property

Syntax

Visual Basic (Declaration)

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

C#

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

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

WijmoControlMode Property

Syntax

Visual Basic (Declaration)

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

```
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<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.")>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
Public Overridable Property WijmoControlMode As WijmoControlMode
```

C#

```
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Hidden)]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[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.")]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
public virtual WijmoControlMode WijmoControlMode {get; set;}
```

See Also

Reference

[C1TargetControlBase Class](#)

[C1TargetControlBase Members](#)

WijmoCssAdapter Property

Syntax

Visual Basic (Declaration)

```
<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=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea")>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Hidden)>
Public Overridable Property WijmoCssAdapter As System.String
```

C#

```
[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=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[C1Category("Behavior")]
[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
 	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
 	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.
≡	RegisterDesignTimeStyleSheet	
≡	RegisterDesignTimeStyleSheets	
≡	RegisterIncludes	
≡	RegisterOnSubmitStatement	
≡	RegisterRunTimeStylesheets	
≡	RegisterScriptDescriptors	
≡	RegisterScriptDescriptorsWithoutScriptManager	
≡	RegisterScriptDescriptorsWithScriptManager	
≡	RenderJsonDataField	
≡	ShowAbout	

[Top](#)

See Also

Reference

[C1TargetControlHelper<T> Class](#)

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

C1TargetControlHelper<T> Constructor

Syntax

Visual Basic (Declaration)	
Public Function New(_	

<pre> ByVal control As T _) </pre>	
C#	
<pre> public C1TargetControlHelper<T>(T control) </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.
≡	RegisterDesignTimeStyleSheet	
≡	RegisterDesignTimeStyleSheets	
≡	RegisterIncludes	
≡	RegisterOnSubmitStatement	
≡	RegisterRunTimeStylesheets	
≡	RegisterScriptDescriptors	

	RegisterScriptDescriptorsWithoutScriptManager	
	RegisterScriptDescriptorsWithScriptManager	
	RenderJsonDataField	
	ShowAbout	

[Top](#)

See Also

Reference

[C1TargetControlHelper<T> Class](#)

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

EnsureEnabledState Method

Syntax

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

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

GetScriptDescriptors Method

Overload List

Overload	Description
GetScriptDescriptors()	
GetScriptDescriptors(String)	

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

[GetScriptDescriptors\(\) Method](#)

Syntax

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

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

[Overload List](#)

[GetScriptDescriptors\(String\) Method](#)

Syntax

Visual Basic (Declaration)	
Public Overloads Function GetScriptDescriptors(_ ByVal <i>targetSelector</i> As System.String _) As System.Collections.Generic.IEnumerable(Of ScriptDescriptor)	
C#	
public System.Collections.Generic.IEnumerable<ScriptDescriptor> GetScriptDescriptors(System.string <i>targetSelector</i>)	

Parameters

targetSelector

See Also

Reference

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

RegisterDesignTimeStyleSheet Method

Syntax

Visual Basic (Declaration)	
<pre>Public Sub RegisterDesignTimeStyleSheet(_ ByVal <i>writer</i> As System.Web.UI.HtmlTextWriter, _ ByVal <i>cssResourceName</i> As System.String _)</pre>	
C#	
<pre>public void RegisterDesignTimeStyleSheet(System.Web.UI.HtmlTextWriter <i>writer</i>, System.string <i>cssResourceName</i>)</pre>	

Parameters

writer

cssResourceName

See Also

Reference

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

RegisterDesignTimeStyleSheets Method

Syntax

Visual Basic (Declaration)	
<pre>Public Sub RegisterDesignTimeStyleSheets(_ ByVal <i>writer</i> As System.Web.UI.HtmlTextWriter _</pre>	

)	
C#	
public void RegisterDesignTimeStyleSheets(System.Web.UI.HtmlTextWriter <i>writer</i>)	

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 <i>callback</i> As System.String _)	

C#	
<pre>public void RegisterOnSubmitStatement(System.string <i>callback</i>)</pre>	

Parameters

callback

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

RegisterRunTimeStylesheets Method

Syntax

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

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

RegisterScriptDescriptors Method

Syntax

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

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 writer As System.Web.UI.HtmlTextWriter _  
)
```

C#

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

Parameters

writer

See Also

Reference

[C1TargetControlHelper<T> Class](#)

[C1TargetControlHelper<T> Members](#)

ShowAbout Method

Syntax

Visual Basic (Declaration)

```
Public Sub ShowAbout()
```

C#

```
public void ShowAbout()
```

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.C1TreeView

Namespace

Overview


Classes

	Class	Description
	C1TreeView	
	C1TreeViewEventArgs	
	C1TreeViewNode	
	C1TreeViewNodeBinding	
	C1TreeViewNodeBindingCollection	
	C1TreeViewNodeCollection	
	C1TreeViewNodeDroppedEventArgs	
	C1TreeViewSerializer	


Interfaces

	Interface	Description
	IC1TreeViewNodeCollectionOwner	

Delegates

	Delegate	Description
	C1TreeViewEventHandler	

Enumerations

	Enumeration	Description
	C1TreeViewNodeCheckState	

See Also

Reference

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

Classes

C1TreeView

Object Model

C1TreeView

Syntax

Visual Basic (Declaration)	
<pre> <System.Drawing.ToolboxBitmapAttribute()> <System.ComponentModel.DesignerAttribute(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner", DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1TreeView.C1TreeViewDesigner, C1.Web.Wijmo.Controls.Design.4, Version=4.0.20152.216, Culture=neutral, PublicKeyToken=9b75583953471eea")> <WidgetDependenciesAttribute(System.Object[])> <System.ComponentModel.LicenseProviderAttribute()> <System.Web.UI.ToolboxDataAttribute("<{0}:C1TreeView runat=server></{0}:C1TreeView">")> Public Class C1TreeView Inherits C1.Web.Wijmo.Controls.Base.C1TargetHierarchicalDataBoundControlBase Implements IC1TreeViewNodeCollectionOwner </pre>	

C#

```
[System.Drawing.ToolboxBitmap()]
[System.ComponentModel.Designer(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1TreeView.C1TreeViewDesigner,
C1.Web.Wijmo.Controls.Design.4, Version=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[WidgetDependencies(System.Object[])]
[System.ComponentModel.LicenseProvider()]
[System.Web.UI.ToolboxData("<{0}:C1TreeView runat=server></{0}:C1TreeView>")]
public class C1TreeView :
C1.Web.Wijmo.Controls.Base.C1TargetHierarchicalDataBoundControlBase,
IC1TreeViewNodeCollectionOwner
```

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.C1TreeView.C1TreeView
```

See Also

Reference

[C1TreeView Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

C1TreeView

Syntax

Visual Basic (Declaration)

```
<System.Drawing.ToolboxBitmapAttribute>
<System.ComponentModel.DesignerAttribute(DesignerBaseTypeName="System.ComponentModel.Design.IDesigner",
```

```

DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1TreeView.C1TreeViewDesigner,
C1.Web.Wijmo.Controls.Design.4, Version=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea">
<WidgetDependenciesAttribute(System.Object[])>
<System.ComponentModel.LicenseProviderAttribute()>
<System.Web.UI.ToolboxDataAttribute("<{0}:C1TreeView
runat=server></{0}:C1TreeView">")>
Public Class C1TreeView
    Inherits
C1.Web.Wijmo.Controls.Base.C1TargetHierarchicalDataBoundControlBase
    Implements IC1TreeViewNodeCollectionOwner

```

C#

```

[System.Drawing.ToolboxBitmap()]
[System.ComponentModel.Designer(DesignerBaseTypeName="System.ComponentModel.D
esign.IDesigner",
DesignerTypeName="C1.Web.Wijmo.Controls.Design.C1TreeView.C1TreeViewDesigner,
C1.Web.Wijmo.Controls.Design.4, Version=4.0.20152.216, Culture=neutral,
PublicKeyToken=9b75583953471eea")]
[WidgetDependencies(System.Object[])]
[System.ComponentModel.LicenseProvider()]
[System.Web.UI.ToolboxData("<{0}:C1TreeView runat=server></{0}:C1TreeView">")]
public class C1TreeView :
C1.Web.Wijmo.Controls.Base.C1TargetHierarchicalDataBoundControlBase,
IC1TreeViewNodeCollectionOwner

```

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.C1TreeView.C1TreeView

```

See Also

Reference


[C1TreeView Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members













The following tables list the members exposed by [C1TreeView](#).
















Public Constructors



















	Name	Description
	C1TreeView Constructor	


















[Top](#)




Public Properties

	Name	Description
	AccessKey	(Inherited from System.Web.UI.WebControls.WebControl)
	AllowDrag	
	AllowDrop	
	AllowEdit	
	AllowSorting	
	AllowTriState	
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from System.Web.UI.WebControls.WebControl)
	AutoCheckNodes	
	AutoCollapse	
	AutoPostBack	
	ClientID	(Inherited from System.Web.UI.Control)

	ClientIDMode	(Inherited from System.Web.UI.Control)
	CollapseAnimation	
	CollapseDelay	
	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)
	DataBindings	
	DataBindStartLevel	
	DataSource	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	DataSourceID	(Inherited from System.Web.UI.WebControls.HierarchicalDataBoundControl)
	EnableTheming	(Inherited from System.Web.UI.WebControls.WebControl)
	EnableViewState	(Inherited from System.Web.UI.Control)
	ExpandAnimation	
	ExpandCollapseHoverUsed	
	ExpandDelay	














	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	Height	(Inherited from System.Web.UI.WebControls.WebControl)
	ID	(Inherited from System.Web.UI.Control)
	LoadOnDemand	
	NamingContainer	(Inherited from System.Web.UI.Control)
	Nodes	
	NodesTemplate	
	OnClientNodeCheckChanged	
	OnClientNodeCheckChanging	
	OnClientNodeClick	
	OnClientNodeCollapsed	
	OnClientNodeDragging	
	OnClientNodeDragStarted	
	OnClientNodeDropped	
	OnClientNodeExpanded	
	OnClientNodeMouseOut	
	OnClientNodeMouseOver	
	OnClientNodeTextChanged	












	OnClientSelectedNodeChanged	
	Owner	
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	SelectedNodes	
	ShowCheckBoxes	
	ShowExpandCollapse	
	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.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)
	UniqueID	Overridden.

	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Width	(Inherited from System.Web.UI.WebControls.WebControl)

[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)
	CreateTreeNode	
	DataBind	Overridden.
	Dispose	(Inherited from System.Web.UI.Control)
	ExpandAllNode	Overloaded.
	FindControl	(Inherited from System.Web.UI.Control)
	FindNodeByNavigateUrl	
	FindNodeByText	
	FindNodeByValue	
	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.
	MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	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)
	SaveLayout	Overloaded.
	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)
	Load	(Inherited from System.Web.UI.Control)

	NodeCheckChanged	
	NodeClicked	
	NodeCollapsed	
	NodeDataBound	
	NodeDropped	
	NodeExpanded	
	NodeTextChanged	
	PreRender	(Inherited from System.Web.UI.Control)
	SelectedNodesChanged	
	Unload	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1TreeView Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

C1TreeView Constructor

Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1TreeView()	

See Also

Reference











[C1TreeView Class](#)[C1TreeView Members](#)[Overload List](#)

Methods

For a list of all members of this type, see [C1TreeView 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)
≡	CreateTreeViewNode	
≡	DataBind	Overridden.
≡	Dispose	(Inherited from System.Web.UI.Control)
≡	ExpandAllNode	Overloaded.
≡	FindControl	(Inherited from System.Web.UI.Control)
≡	FindNodeByNavigateUrl	
≡	FindNodeByText	
≡	FindNodeByValue	
≡	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.
 MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
 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)
 SaveLayout	Overloaded.
 SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1TreeView Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

CreateTreeViewNode Method

Syntax

Visual Basic (Declaration)

```
Public Function CreateTreeViewNode() As C1TreeViewNode
```

C#

```
public C1TreeViewNode CreateTreeViewNode()
```

See Also

Reference

[C1TreeView Class](#)[C1TreeView Members](#)

DataBind Method

Syntax

Visual Basic (Declaration)	
Public Overrides NotOverridable Sub DataBind()	
C#	
public override void DataBind()	

See Also

Reference

[C1TreeView Class](#)[C1TreeView Members](#)

ExpandAllNode Method

Overload List

Overload	Description
ExpandAllNode()	
ExpandAllNode(IC1TreeViewNodeCollectionOwner)	

See Also

Reference

[C1TreeView Class](#)[C1TreeView Members](#)

ExpandAllNode() Method

Syntax

Visual Basic (Declaration)	
Public Overloads Overridable Sub ExpandAllNode()	

C#	
----	--

public virtual void ExpandAllNode()

See Also

Reference

[C1TreeView Class](#)[C1TreeView Members](#)[Overload List](#)

ExpandAllNode(IC1TreeViewNodeCollectionOwner) Method

Syntax

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

Public Overloads Overridable Sub ExpandAllNode(_ ByVal <i>control</i> As IC1TreeViewNodeCollectionOwner _)
--

C#	
----	--

public virtual void ExpandAllNode(IC1TreeViewNodeCollectionOwner <i>control</i>)
--

Parameters

control

See Also

Reference

[C1TreeView Class](#)[C1TreeView Members](#)[Overload List](#)

FindNodeByNavigateUrl Method

Syntax

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

Public Function FindNodeByNavigateUrl(_ ByVal <i>url</i> As System.String _

```
) As C1TreeNode
```

```
C#
```

```
public C1TreeNode FindNodeByNavigateUrl(  
    System.string url  
)
```

Parameters

url

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

FindNodeByText Method

Syntax

```
Visual Basic (Declaration)
```

```
Public Function FindNodeByText( _  
    ByVal text As System.String _  
) As C1TreeNode
```

```
C#
```

```
public C1TreeNode FindNodeByText(  
    System.string text  
)
```

Parameters

text

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

FindNodeByValue Method

Syntax

Visual Basic (Declaration)

```
Public Function FindNodeByValue( _  
    ByVal value As System.String _  
) As C1TreeNode
```

C#

```
public C1TreeNode FindNodeByValue(  
    System.string value  
)
```

Parameters

value

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

LoadLayout Method

Overload List

Overload	Description
LoadLayout(String)	
LoadLayout(Stream)	
LoadLayout(String,LayoutType)	
LoadLayout(Stream,LayoutType)	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

LoadLayout(String) Method

Syntax

Visual Basic (Declaration)

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

C#

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

Parameters

filename

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

[Overload List](#)

LoadLayout(Stream) Method

Syntax

Visual Basic (Declaration)

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

C#

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

Parameters

stream

See Also

Reference

[C1TreeView Class](#)
[C1TreeView Members](#)
[Overload List](#)

LoadLayout(String,LayoutType) Method

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

layoutTypes

See Also

Reference

[C1TreeView Class](#)
[C1TreeView Members](#)
[Overload List](#)

LoadLayout(Stream,LayoutType) Method

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**layoutTypes***See Also****Reference**[C1TreeView Class](#)[C1TreeView Members](#)[Overload List](#)**SaveLayout Method****Overload List**

Overload	Description
SaveLayout(String)	
SaveLayout(Stream)	

See Also**Reference**[C1TreeView Class](#)[C1TreeView Members](#)**SaveLayout(String) Method****Syntax****Visual Basic (Declaration)**

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

C#	
----	--

<pre>public void SaveLayout(System.string <i>filename</i>)</pre>	
--	--

Parameters

filename

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

[Overload List](#)

SaveLayout(Stream) Method

Syntax

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

<pre>Public Overloads Sub SaveLayout(_ ByVal <i>stream</i> As System.IO.Stream _)</pre>	
---	--

C#	
----	--

<pre>public void SaveLayout(System.IO.Stream <i>stream</i>)</pre>	
---	--

Parameters

stream

See Also

Reference

[C1TreeView Class](#)

















[C1TreeView Members](#)
















[Overload List](#)




















Properties

















For a list of all members of this type, see [C1TreeView members](#).

Public Properties

	Name	Description
	AccessKey	(Inherited from System.Web.UI.WebControls.WebControl)
	AllowDrag	
	AllowDrop	
	AllowEdit	
	AllowSorting	
	AllowTriState	
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from System.Web.UI.WebControls.WebControl)
	AutoCheckNodes	
	AutoCollapse	
	AutoPostBack	
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	CollapseAnimation	
	CollapseDelay	
	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)
	DataBindings	
	DataBindStartLevel	
	DataSource	(Inherited from System.Web.UI.WebControls.BaseDataBoundControl)
	DataSourceID	(Inherited from System.Web.UI.WebControls.HierarchicalDataBoundControl)
	EnableTheming	(Inherited from System.Web.UI.WebControls.WebControl)
	EnableViewState	(Inherited from System.Web.UI.Control)
	ExpandAnimation	
	ExpandCollapseHoverUsed	
	ExpandDelay	
	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	Height	(Inherited from System.Web.UI.WebControls.WebControl)
	ID	(Inherited from System.Web.UI.Control)

	LoadOnDemand	
	NamingContainer	(Inherited from System.Web.UI.Control)
	Nodes	
	NodesTemplate	
	OnClientNodeCheckChanged	
	OnClientNodeCheckChanging	
	OnClientNodeClick	
	OnClientNodeCollapsed	
	OnClientNodeDragging	
	OnClientNodeDragStarted	
	OnClientNodeDropped	
	OnClientNodeExpanded	
	OnClientNodeMouseOut	
	OnClientNodeMouseOver	
	OnClientNodeTextChanged	
	OnClientSelectedNodeChanged	
	Owner	
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)

	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	SelectedNodes	
	ShowCheckBoxes	
	ShowExpandCollapse	
	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.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)
	UniqueID	Overridden.
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	ViewStateMode	(Inherited from System.Web.UI.Control)
	Width	(Inherited from System.Web.UI.WebControls.WebControl)

[Top](#)

See Also

Reference

[C1TreeView Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

AllowDrag Property

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>  
<System.ComponentModel.DefaultValueAttribute()>  
<WidgetOptionAttribute()>  
<C1DescriptionAttribute(Key="C1TreeView.AllowDrag", Description="Allow to  
TreeView nodes to perform drag .")>  
<C1CategoryAttribute("Behavior")>  
Public Property AllowDrag As System.Boolean
```

C#

```
[Layout(LayoutType.Behavior)]  
[System.ComponentModel.DefaultValue()]  
[WidgetOption()]  
[C1Description(Key="C1TreeView.AllowDrag", Description="Allow to TreeView  
nodes to perform drag .")]  
[C1Category("Behavior")]  
public System.bool AllowDrag {get; set;}
```

AllowDrop Property

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>  
<C1DescriptionAttribute(Key="C1TreeView.AllowDrop", Description="Allow to  
TreeView nodes to perform drop .")>  
<System.ComponentModel.DefaultValueAttribute()>  
<WidgetOptionAttribute()>  
<C1CategoryAttribute("Behavior")>  
Public Property AllowDrop As System.Boolean
```

C#

```
[Layout(LayoutType.Behavior)]  
[C1Description(Key="C1TreeView.AllowDrop", Description="Allow to TreeView  
nodes to perform drop .")]  
[System.ComponentModel.DefaultValue()]  
[WidgetOption()]  
[C1Category("Behavior")]  
public System.bool AllowDrop {get; set;}
```

AllowEdit Property

Visual Basic (Declaration)

```
<C1CategoryAttribute("Behavior")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeView.AllowEdit", Description="Allow nodes
to be edited at run time.")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property AllowEdit As System.Boolean
```

C#

```
[C1Category("Behavior")]
[WidgetOption()]
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeView.AllowEdit", Description="Allow nodes to be
edited at run time.")]
[System.ComponentModel.DefaultValue()]
public System.bool AllowEdit {get; set;}
```

AllowSorting Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1TreeView.AllowSorting", Description="Allow
nodes to be sorted at run time.")>
Public Property AllowSorting As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[Layout(LayoutType.Behavior)]
[C1Category("Behavior")]
[C1Description(Key="C1TreeView.AllowSorting", Description="Allow nodes to be
sorted at run time.")]
public System.bool AllowSorting {get; set;}
```

AllowTriState Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1TreeView.AllowTriState", Description="Allow
TriState of CheckBox")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
Public Property AllowTriState As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1TreeView.AllowTriState", Description="Allow TriState of
CheckBox")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
[C1Category("Behavior")]
public System.bool AllowTriState {get; set;}
```

AutoCheckNodes Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1TreeView.AutoCheckNodes", Description="Allow
sub nodes to be checked on parent node check.")>
Public Property AutoCheckNodes As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[Layout(LayoutType.Behavior)]
[C1Category("Behavior")]
[C1Description(Key="C1TreeView.AutoCheckNodes", Description="Allow sub nodes
to be checked on parent node check.")]
public System.bool AutoCheckNodes {get; set;}
```

AutoCollapse Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Behavior")>
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeView.AutoCollapse", Description="If this
option is set to true, expanded node will be collapsed if other node is
expanded.")>
<WidgetOptionAttribute()>
Public Property AutoCollapse As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Behavior")]
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeView.AutoCollapse", Description="If this option is
set to true, expanded node will be collapsed if other node is expanded.")]
[WidgetOption()]
public System.bool AutoCollapse {get; set;}
```

AutoPostBack Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1TreeView.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.")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
Public Property AutoPostBack As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Behavior")]
[C1Description(Key="C1TreeView.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.")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
public System.bool AutoPostBack {get; set;}
```

CollapseAnimation Property

Visual Basic (Declaration)

```
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<C1DescriptionAttribute(Key="C1TreeView.CollapseAnimation", Description="Gets
or sets the animation effect when the node is collapsed.")>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeri-
alizationVisibility.Content)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
Public Property CollapseAnimation As Animation
```

C#

```
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[C1Description(Key="C1TreeView.CollapseAnimation", Description="Gets or sets
the animation effect when the node is collapsed.")]
[C1Category("Behavior")]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV-
isibility.Content)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[WidgetOption()]
[Layout(LayoutType.Behavior)]
public Animation CollapseAnimation {get; set;}
```

CollapseDelay Property

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeView.CollapseDelay", Description="Collapse
delay in milliseconds before child nodes collapse.")>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
Public Property CollapseDelay As System.Integer
```

C#

```
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeView.CollapseDelay", Description="Collapse delay in
milliseconds before child nodes collapse.")]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
public System.int CollapseDelay {get; set;}
```

DataBindings Property

Visual Basic (Declaration)

```
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<C1CategoryAttribute("Data")>
<C1DescriptionAttribute(Key="C1TreeView.DataBindings", Description="Data
bindings for items in the control.")>
<LayoutAttribute(LayoutType.Data)>
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.MergablePropertyAttribute(False)>
Public ReadOnly Property DataBindings As C1TreeViewNodeBindingCollection
```

C#

```
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[C1Category("Data")]
[C1Description(Key="C1TreeView.DataBindings", Description="Data bindings for
items in the control.")]
[Layout(LayoutType.Data)]
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.MergableProperty(false)]
public C1TreeViewNodeBindingCollection DataBindings {get;}
```

DataBindStartLevel Property

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeView.DataBindStartLevel",
Description="Gets or sets the value that indicates DataBind start level.")>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Data)>
Public Property DataBindStartLevel As System.Integer
```

C#

```
[C1Description(Key="C1TreeView.DataBindStartLevel", Description="Gets or sets
the value that indicates DataBind start level.")]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Data)]
public System.int DataBindStartLevel {get; set;}
```

ExpandAnimation Property

Visual Basic (Declaration)

```
<C1CategoryAttribute("Behavior")>
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeView.ExpandAnimation", Description="Gets
or sets the animation effect when the node is expanded.")>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Content)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter,
System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<WidgetOptionAttribute()>
Public Property ExpandAnimation As Animation
```

C#

```
[C1Category("Behavior")]
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeView.ExpandAnimation", Description="Gets or sets
the animation effect when the node is expanded.")]
[System.ComponentModel.NotifyParentProperty(true)]
[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")]
[WidgetOption()]
public Animation ExpandAnimation {get; set;}
```

ExpandCollapseHoverUsed Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1TreeView.ExpandCollapseHoverUsed",
Description="Allow nodes to use hover for Expand or Collapse elements.")>
Public Property ExpandCollapseHoverUsed As System.Boolean
```

C#

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

```
[WidgetOption()]
[Layout(LayoutType.Behavior)]
[C1Category("Behavior")]
[C1Description(Key="C1TreeView.ExpandCollapseHoverUsed", Description="Allow
nodes to use hover for Expand or Collapse elements.")]
public System.bool ExpandCollapseHoverUsed {get; set;}
```

ExpandDelay Property

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeView.ExpandDelay", Description="Expand
delay in milliseconds before child nodes expand.")>
<LayoutAttribute(LayoutType.Behavior)>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
Public Property ExpandDelay As System.Integer
```

C#

```
[C1Description(Key="C1TreeView.ExpandDelay", Description="Expand delay in
milliseconds before child nodes expand.")]
[Layout(LayoutType.Behavior)]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
public System.int ExpandDelay {get; set;}
```

LoadOnDemand Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1TreeView.LoadOnDemand", Description="Gets or
sets the value that indicates whether or not Loads on demand is enabled.")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetOptionAttribute()>
Public Property LoadOnDemand As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Category("Behavior")]
[C1Description(Key="C1TreeView.LoadOnDemand", Description="Gets or sets the
value that indicates whether or not Loads on demand is enabled.")]
[Layout(LayoutType.Behavior)]
[WidgetOption()]
public System.bool LoadOnDemand {get; set;}
```

Nodes Property

Visual Basic (Declaration)


```
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Content)>
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.BrowsableAttribute(False)>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<TypeConverterAttribute("System.ComponentModel.ExpandableObjectConverter, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")>
<WidgetOptionAttribute()>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
Public ReadOnly Property Nodes As C1TreeViewNodeCollection
```

C#

```
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Content)]
[Layout(LayoutType.Appearance)]
[System.ComponentModel.Browsable(false)]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[TypeConverter("System.ComponentModel.ExpandableObjectConverter, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089")]
[WidgetOption()]
[System.ComponentModel.NotifyParentProperty(true)]
public C1TreeViewNodeCollection Nodes {get;}
```

NodesTemplate Property

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(False)>
<C1DescriptionAttribute(Key="C1TreeView.NodesTemplate", Description="If specified, this template will be applied for all menu items that does not have other defined templates.")>
<System.ComponentModel.BindableAttribute(Bindable=False, Direction=BindingDirection.OneWay)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Content)>
<System.Web.UI.PersistencemodeAttribute(Persistencemode.InnerProperty)>
<System.Web.UI.TemplateInstanceAttribute(TemplateInstance.Single)>
Public Overridable Property NodesTemplate As System.Web.UI.ITemplate
```

C#

```
[System.ComponentModel.Browsable(false)]
[C1Description(Key="C1TreeView.NodesTemplate", Description="If specified, this template will be applied for all menu items that does not have other defined templates.")]
[System.ComponentModel.Bindable(Bindable=false, Direction=BindingDirection.OneWay)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationVisibility.Content)]
[System.Web.UI.Persistencemode(Persistencemode.InnerProperty)]
[System.Web.UI.TemplateInstance(TemplateInstance.Single)]
public virtual System.Web.UI.ITemplate NodesTemplate {get; set;}
```

OnClientNodeCheckChanged Property

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("nodeCheckChanged")>
<LayoutAttribute(LayoutType.Behavior)>
<WidgetEventAttribute("ev, node")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeCheckChanged",
Description="If specified, the OnClientNodeCheckChanged client-side event
handler is called after a TreeViewNode check status has been changed.(If
Checkboxes are enabled)")>
<C1CategoryAttribute("Client-Side Events")>
Public Property OnClientNodeCheckChanged As System.String
```

C#

```
[WidgetOptionName("nodeCheckChanged")]
[Layout(LayoutType.Behavior)]
[WidgetEvent("ev, node")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1TreeView.OnClientNodeCheckChanged", Description="If
specified, the OnClientNodeCheckChanged client-side event handler is called
after a TreeViewNode check status has been changed.(If Checkboxes are
enabled)")]
[C1Category("Client-Side Events")]
public System.string OnClientNodeCheckChanged {get; set;}
```

OnClientNodeCheckChanging Property

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeCheckChanging",
Description="If specified, the OnClientNodeCheckChanged client-side event
handler is called before a TreeViewNode check status has been changed(If
Checkboxes are enabled). This event can be cancelled by returning false.")>
<WidgetEventAttribute("ev, node")>
<LayoutAttribute(LayoutType.Behavior)>
<C1CategoryAttribute("Client-Side Events")>
<WidgetOptionNameAttribute("nodeCheckChanging")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property OnClientNodeCheckChanging As System.String
```

C#

```
[C1Description(Key="C1TreeView.OnClientNodeCheckChanging", Description="If
specified, the OnClientNodeCheckChanged client-side event handler is called
before a TreeViewNode check status has been changed(If Checkboxes are
enabled). This event can be cancelled by returning false.")]
[WidgetEvent("ev, node")]
[Layout(LayoutType.Behavior)]
[C1Category("Client-Side Events")]
[WidgetOptionName("nodeCheckChanging")]
[System.ComponentModel.DefaultValue()]
public System.string OnClientNodeCheckChanging {get; set;}
```

OnClientNodeClick Property

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeClicked", Description="If  
specified, the OnClientNodeClicked client-side event handler is called after  
a TreeViewNode has been clicked.")>  
<WidgetOptionNameAttribute("nodeClick")>  
<LayoutAttribute(LayoutType.Behavior)>  
<System.ComponentModel.DefaultValueAttribute()>  
<C1CategoryAttribute("Client-Side Events")>  
<WidgetEventAttribute("ev, node")>  
Public Property OnClientNodeClick As System.String
```

C#

```
[C1Description(Key="C1TreeView.OnClientNodeClicked", Description="If  
specified, the OnClientNodeClicked client-side event handler is called after  
a TreeViewNode has been clicked.")]  
[WidgetOptionName("nodeClick")]  
[Layout(LayoutType.Behavior)]  
[System.ComponentModel.DefaultValue()]  
[C1Category("Client-Side Events")]  
[WidgetEvent("ev, node")]  
public System.string OnClientNodeClick {get; set;}
```

OnClientNodeCollapsed Property

Visual Basic (Declaration)

```
<WidgetEventAttribute("ev, node")>  
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeCollapsed",  
Description="If specified, the OnClientNodeCollapsed client-side event  
handler is called after a TreeViewNode has been collapsed.")>  
<C1CategoryAttribute("Client-Side Events")>  
<LayoutAttribute(LayoutType.Behavior)>  
<WidgetOptionNameAttribute("nodeCollapsed")>  
<System.ComponentModel.DefaultValueAttribute()>  
Public Property OnClientNodeCollapsed As System.String
```

C#

```
[WidgetEvent("ev, node")]  
[C1Description(Key="C1TreeView.OnClientNodeCollapsed", Description="If  
specified, the OnClientNodeCollapsed client-side event handler is called  
after a TreeViewNode has been collapsed.")]  
[C1Category("Client-Side Events")]  
[Layout(LayoutType.Behavior)]  
[WidgetOptionName("nodeCollapsed")]  
[System.ComponentModel.DefaultValue()]  
public System.string OnClientNodeCollapsed {get; set;}
```

OnClientNodeDragging Property

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeDragging",
Description="If specified, the OnClientNodeDragging client-side event handler
is called if the user moves the mouse while dragging the node.")>
<LayoutAttribute(LayoutType.Behavior)>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("ev, node")>
<WidgetOptionNameAttribute("nodeDragging")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property OnClientNodeDragging As System.String
```

C#

```
[C1Description(Key="C1TreeView.OnClientNodeDragging", Description="If
specified, the OnClientNodeDragging client-side event handler is called if
the user moves the mouse while dragging the node.")]
[Layout(LayoutType.Behavior)]
[C1Category("Client-Side Events")]
[WidgetEvent("ev, node")]
[WidgetOptionName("nodeDragging")]
[System.ComponentModel.DefaultValue()]
public System.string OnClientNodeDragging {get; set;}
```

OnClientNodeDragStarted Property

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeDragStarted",
Description="If specified, the OnClientNodeDragStarted client-side event
handler is called before drag is started.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("ev, node")>
<WidgetOptionNameAttribute("nodeDragStarted")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property OnClientNodeDragStarted As System.String
```

C#

```
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeView.OnClientNodeDragStarted", Description="If
specified, the OnClientNodeDragStarted client-side event handler is called
before drag is started.")]
[C1Category("Client-Side Events")]
[WidgetEvent("ev, node")]
[WidgetOptionName("nodeDragStarted")]
[System.ComponentModel.DefaultValue()]
public System.string OnClientNodeDragStarted {get; set;}
```

OnClientNodeDropped Property

Visual Basic (Declaration)

```
<WidgetOptionNameAttribute("nodeDropped")>
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeDropped", Description="If
specified, the OnClientNodeDropped client-side event handler is called before
```

```
a TreeNode has been dropped by user.">
<C1CategoryAttribute("Client-Side Events">
<WidgetEventAttribute("ev, data">
<System.ComponentModel.DefaultValueAttribute()>
Public Property OnClientNodeDropped As System.String
```

C#

```
[WidgetOptionName("nodeDropped")]
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeView.OnClientNodeDropped", Description="If
specified, the OnClientNodeDropped client-side event handler is called before
a TreeNode has been dropped by user.")]
[C1Category("Client-Side Events")]
[WidgetEvent("ev, data")]
[System.ComponentModel.DefaultValue()]
public System.string OnClientNodeDropped {get; set;}
```

OnClientNodeExpanded Property

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeExpanded",
Description="If specified, the OnClientNodeExpanded client-side event handler
is called after a TreeNode has been expanded.">
<C1CategoryAttribute("Client-Side Events">
<WidgetEventAttribute("ev, node">
<WidgetOptionNameAttribute("nodeExpanded">
Public Property OnClientNodeExpanded As System.String
```

C#

```
[Layout(LayoutType.Behavior)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1TreeView.OnClientNodeExpanded", Description="If
specified, the OnClientNodeExpanded client-side event handler is called after
a TreeNode has been expanded.")]
[C1Category("Client-Side Events")]
[WidgetEvent("ev, node")]
[WidgetOptionName("nodeExpanded")]
public System.string OnClientNodeExpanded {get; set;}
```

OnClientNodeMouseOut Property

Visual Basic (Declaration)

```
<WidgetEventAttribute("ev, node">
<LayoutAttribute(LayoutType.Behavior)>
<C1CategoryAttribute("Client-Side Events">
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeMouseOut",
Description="If specified, the OnClientNodeMouseOut client-side event handler
is called if the user moves the mouse pointer out of the node.">
<WidgetOptionNameAttribute("nodeMouseOut">
<System.ComponentModel.DefaultValueAttribute()>
```

Public Property OnClientNodeMouseOut As System.String

C#

```
[WidgetEvent("ev, node")]
[Layout(LayoutType.Behavior)]
[C1Category("Client-Side Events")]
[C1Description(Key="C1TreeView.OnClientNodeMouseOut", Description="If
specified, the OnClientNodeMouseOut client-side event handler is called if
the user moves the mouse pointer out of the node.")]
[WidgetOptionName("nodeMouseOut")]
[System.ComponentModel.DefaultValue()]
public System.string OnClientNodeMouseOut {get; set;}
```

OnClientNodeMouseOver Property

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeMouseOver",
Description="If specified, the OnClientNodeMouseOut client-side event handler
is called if the user moves the mouse pointer over of the node.")>
<WidgetOptionNameAttribute("nodeMouseOver")>
<WidgetEventAttribute("ev, node")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Client-Side Events")>
Public Property OnClientNodeMouseOver As System.String
```

C#

```
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeView.OnClientNodeMouseOver", Description="If
specified, the OnClientNodeMouseOut client-side event handler is called if
the user moves the mouse pointer over of the node.")]
[WidgetOptionName("nodeMouseOver")]
[WidgetEvent("ev, node")]
[System.ComponentModel.DefaultValue()]
[C1Category("Client-Side Events")]
public System.string OnClientNodeMouseOver {get; set;}
```

OnClientNodeTextChanged Property

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeView.OnClientNodeTextChanged",
Description="If specified, the OnClientNodeTextChanged client-side event
handler is called after a TreeViewNode's has been edited.")>
<C1CategoryAttribute("Client-Side Events")>
<WidgetEventAttribute("ev, node")>
<WidgetOptionNameAttribute("nodeTextChanged")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property OnClientNodeTextChanged As System.String
```

C#

```
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeView.OnClientNodeTextChanged", Description="If
specified, the OnClientNodeTextChanged client-side event handler is called
after a TreeViewNode's has been edited.")]
[C1Category("Client-Side Events")]
[WidgetEvent("ev, node")]
[WidgetOptionName("nodeTextChanged")]
[System.ComponentModel.DefaultValue()]
public System.string OnClientNodeTextChanged {get; set;}
```

OnClientSelectedNodeChanged Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetEventAttribute("ev, node")>
<LayoutAttribute(LayoutType.Behavior)>
<C1CategoryAttribute("Client-Side Events")>
<C1DescriptionAttribute(Key="C1TreeView.OnClientSelectedNodesChanged",
Description="If specified, the OnClientSelectedNodesChanged client-side event
handler is called after a TreeViewNode has been selected.")>
<WidgetOptionNameAttribute("selectedNodeChanged")>
Public Property OnClientSelectedNodeChanged As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[WidgetEvent("ev, node")]
[Layout(LayoutType.Behavior)]
[C1Category("Client-Side Events")]
[C1Description(Key="C1TreeView.OnClientSelectedNodesChanged", Description="If
specified, the OnClientSelectedNodesChanged client-side event handler is
called after a TreeViewNode has been selected.")]
[WidgetOptionName("selectedNodeChanged")]
public System.string OnClientSelectedNodeChanged {get; set;}
```

Owner Property

Syntax

Visual Basic (Declaration)	
<pre><System.ComponentModel.BrowsableAttribute(False)> Public ReadOnly Property Owner As IC1TreeViewNodeCollectionOwner</pre>	
C#	
<pre>[System.ComponentModel.Browsable(false)] public IC1TreeViewNodeCollectionOwner Owner {get;}</pre>	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

SelectedNodes Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(False)>
Public ReadOnly Property SelectedNodes As System.Collections.Generic.List(Of
C1TreeNode)
```

C#

```
[System.ComponentModel.Browsable(false)]
public System.Collections.Generic.List<C1TreeNode> SelectedNodes {get;}
```

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

ShowCheckBoxes Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1TreeView.ShowCheckBoxes", Description="Allow
the check box to be shown on the nodes.")>
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
Public Property ShowCheckBoxes As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1TreeView.ShowCheckBoxes", Description="Allow the check
box to be shown on the nodes.")]
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Category("Behavior")]
public System.bool ShowCheckBoxes {get; set;}
```


ShowExpandCollapse Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<WidgetOptionAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<C1CategoryAttribute("Behavior")>
<C1DescriptionAttribute(Key="C1TreeView.ShowExpandCollapse",
Description="Allow nodes to be expanded or collapsed.")>
Public Property ShowExpandCollapse As System.Boolean
```

C#

```
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[Layout(LayoutType.Appearance)]
[C1Category("Behavior")]
[C1Description(Key="C1TreeView.ShowExpandCollapse", Description="Allow nodes
to be expanded or collapsed.")]
public System.bool ShowExpandCollapse {get; set;}
```

UniqueID Property

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<System.ComponentModel.BrowsableAttribute(False)>
<WidgetOptionNameAttribute("UniqueID")>
<WidgetOptionAttribute()>
Public Overrides ReadOnly Property UniqueID As System.String
```



C#














```
[System.ComponentModel.DefaultValue()]
[System.ComponentModel.Browsable(false)]
[WidgetOptionName("UniqueID")]
[WidgetOption()]
public override System.string UniqueID {get;}
```

Events

For a list of all members of this type, see [C1TreeView 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)
	Load	(Inherited from System.Web.UI.Control)
	NodeCheckChanged	
	NodeClicked	
	NodeCollapsed	
	NodeDataBound	
	NodeDropped	
	NodeExpanded	
	NodeTextChanged	
	PreRender	(Inherited from System.Web.UI.Control)
	SelectedNodesChanged	
	Unload	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1TreeView Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

NodeCheckChanged Event

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Action")>
<C1DescriptionAttribute(Key="C1TreeView.NodeCheckChanged",
Description="Occurs on the server if a node in the TreeView control changes
its check status.")>
Public Event NodeCheckChanged As C1TreeViewEventHandler
```

C#

```
[C1Category("Action")]
[C1Description(Key="C1TreeView.NodeCheckChanged", Description="Occurs on the
server if a node in the TreeView control changes its check status.")]
public event C1TreeViewEventHandler NodeCheckChanged
```

Event Data

The event handler receives an argument of type [C1TreeViewEventArgs](#) containing data related to this event. The following **C1TreeViewEventArgs** properties provide information specific to this event.

Property	Description
Node	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

NodeClicked Event

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Action")>
<C1DescriptionAttribute(Key="C1TreeView.NodeClicked", Description="Occurs on
the server if a node in the TreeView control has been clicked.")>
Public Event NodeClicked As C1TreeViewEventHandler
```

C#

```
[C1Category("Action")]
```

```
[C1Description(Key="C1TreeView.NodeClicked", Description="Occurs on the  
server if a node in the TreeView control has been clicked.")]  
public event C1TreeViewEventHandler NodeClicked
```

Event Data

The event handler receives an argument of type [C1TreeViewEventArgs](#) containing data related to this event. The following **C1TreeViewEventArgs** properties provide information specific to this event.

Property	Description
Node	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

NodeCollapsed Event

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeView.NodeCollapsed", Description="Occurs  
on the server after the node is collapsed.")>  
<C1CategoryAttribute("Action")>  
Public Event NodeCollapsed As C1TreeViewEventHandler
```

C#

```
[C1Description(Key="C1TreeView.NodeCollapsed", Description="Occurs on the  
server after the node is collapsed.")]  
[C1Category("Action")]  
public event C1TreeViewEventHandler NodeCollapsed
```

Event Data

The event handler receives an argument of type [C1TreeViewEventArgs](#) containing data related to this event. The following **C1TreeViewEventArgs** properties provide information specific to this event.

Property	Description
Node	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

NodeDataBound Event

Syntax

Visual Basic (Declaration)	
<pre><C1CategoryAttribute("Action")> <C1DescriptionAttribute(Key="C1TreeView.NodeDataBound", Description="Occurs after a node is data bound.")> Public Event NodeDataBound As C1TreeViewEventHandler</pre>	
C#	
<pre>[C1Category("Action")] [C1Description(Key="C1TreeView.NodeDataBound", Description="Occurs after a node is data bound.")] public event C1TreeViewEventHandler NodeDataBound</pre>	

Event Data

The event handler receives an argument of type [C1TreeViewEventArgs](#) containing data related to this event. The following **C1TreeViewEventArgs** properties provide information specific to this event.

Property	Description
Node	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

NodeDropped Event

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeView.NodeDropped", Description="Occurs on
the server after the node has been dropped.")>
<C1CategoryAttribute("Action")>
Public Event NodeDropped As C1TreeViewEventHandler
```

C#

```
[C1Description(Key="C1TreeView.NodeDropped", Description="Occurs on the
server after the node has been dropped.")]
[C1Category("Action")]
public event C1TreeViewEventHandler NodeDropped
```

Event Data

The event handler receives an argument of type [C1TreeViewEventArgs](#) containing data related to this event. The following **C1TreeViewEventArgs** properties provide information specific to this event.

Property	Description
Node	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

NodeExpanded Event

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeView.NodeExpanded", Description="Occurs on
the server after the node is expanded.")>
<C1CategoryAttribute("Action")>
Public Event NodeExpanded As C1TreeViewEventHandler
```

C#

```
[C1Description(Key="C1TreeView.NodeExpanded", Description="Occurs on the
server after the node is expanded.")]
[C1Category("Action")]
public event C1TreeViewEventHandler NodeExpanded
```

Event Data

The event handler receives an argument of type [C1TreeViewEventArgs](#) containing data related to this event. The following **C1TreeViewEventArgs** properties provide information specific to this event.

Property	Description
Node	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

NodeTextChanged Event

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Action")>
<C1DescriptionAttribute(Key="C1TreeView.NodeTextChanged", Description="Occurs
on the server when a node's Text property is changed.")>
Public Event NodeTextChanged As C1TreeViewEventHandler
```

C#

```
[C1Category("Action")]
[C1Description(Key="C1TreeView.NodeTextChanged", Description="Occurs on the
server when a node's Text property is changed.")]
public event C1TreeViewEventHandler NodeTextChanged
```

Event Data

The event handler receives an argument of type [C1TreeViewEventArgs](#) containing data related to this event. The following **C1TreeViewEventArgs** properties provide information specific to this event.

Property	Description
Node	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

SelectedNodesChanged Event

Syntax

Visual Basic (Declaration)	
<pre><C1DescriptionAttribute(Key="C1TreeView.SelectedNodesChanged", Description="Occurs on the server if a node in the TreeView control has been selected")> <C1CategoryAttribute("Action")> Public Event SelectedNodesChanged As C1TreeViewEventHandler</pre>	
C#	
<pre>[C1Description(Key="C1TreeView.SelectedNodesChanged", Description="Occurs on the server if a node in the TreeView control has been selected")] [C1Category("Action")] public event C1TreeViewEventHandler SelectedNodesChanged</pre>	

Event Data

The event handler receives an argument of type [C1TreeViewEventArgs](#) containing data related to this event. The following **C1TreeViewEventArgs** properties provide information specific to this event.

Property	Description
Node	

See Also

Reference

[C1TreeView Class](#)

[C1TreeView Members](#)

C1TreeViewEventArgs

Object Model

C1TreeViewEventArgs

Syntax

Visual Basic (Declaration)	
Public Class C1TreeViewEventArgs Inherits System.EventArgs	
C#	
public class C1TreeViewEventArgs : System.EventArgs	

Inheritance Hierarchy

System.Object

System.EventArgs

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewEventArgs

[C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewNodeDroppedEventArgs](#)

See Also

Reference

[C1TreeViewEventArgs Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

C1TreeViewEventArgs

Syntax

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

```
Public Class C1TreeViewEventArgs
    Inherits System.EventArgs
```

```
C#
```

```
public class C1TreeViewEventArgs : System.EventArgs
```

Inheritance Hierarchy

System.Object

System.EventArgs

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewEventArgs

[C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewNodeDroppedEventArgs](#)

See Also

Reference


[C1TreeViewEventArgs Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members

The following tables list the members exposed by [C1TreeViewEventArgs](#).

Public Constructors

	Name	Description
	C1TreeViewEventArgs Constructor	

[Top](#)

Public Properties

	Name	Description
	Node	

[Top](#)

See Also

Reference

[C1TreeViewEventArgs Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

C1TreeViewEventArgs Constructor

Syntax

Visual Basic (Declaration)

```
Public Function New( _  
    ByVal node As C1TreeNode _  
)
```

C#

```
public C1TreeViewEventArgs(  
    C1TreeNode node  
)
```

Parameters

node

See Also

Reference

[C1TreeViewEventArgs Class](#)

[C1TreeViewEventArgs Members](#)

Properties

For a list of all members of this type, see [C1TreeViewEventArgs members](#).

Public Properties

	Name	Description
	Node	

[Top](#)

See Also

Reference

[C1TreeViewEventArgs Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Node Property

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property Node As C1TreeNode
```

C#

```
public C1TreeNode Node {get;}
```

See Also

Reference

[C1TreeViewEventArgs Class](#)

[C1TreeViewEventArgs Members](#)

C1TreeNode

Object Model

C1TreeNode

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.ToolboxItemAttribute(ToolboxItemType=,  
ToolboxItemTypeName="")>
```

```
Public Class C1TreeNode
```

```
    Inherits C1.Web.Wijmo.Controls.Base.UIElement
```

```
    Implements IC1TreeNodeCollectionOwner
```

C#

```
[System.ComponentModel.ToolboxItem(ToolboxItemType=, ToolboxItemTypeName="")]
```

```
public class C1TreeNode : C1.Web.Wijmo.Controls.Base.UIElement,
```

```
IC1TreeNodeCollectionOwner
```

Inheritance Hierarchy

System.Object

System.Web.UI.Control

System.Web.UI.WebControls.WebControl

C1.Web.Wijmo.Controls.C1TreeView.C1TreeNode

See Also

Reference

[C1TreeNode Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

C1TreeNode

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.ToolboxItemAttribute(ToolboxItemType=,
ToolboxItemTypeName="")>
Public Class C1TreeNode
    Inherits C1.Web.Wijmo.Controls.Base.UIElement
    Implements IC1TreeNodeCollectionOwner
```

C#

```
[System.ComponentModel.ToolboxItem(ToolboxItemType=, ToolboxItemTypeName="")]
public class C1TreeNode : C1.Web.Wijmo.Controls.Base.UIElement,
IC1TreeNodeCollectionOwner
```

Inheritance Hierarchy

System.Object

System.Web.UI.Control

System.Web.UI.WebControls.WebControl

C1.Web.Wijmo.Controls.C1TreeView.C1TreeNode

See Also

Reference


[C1TreeNode Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members













The following tables list the members exposed by [C1TreeNode](#).


















Public Constructors













	Name	Description
	C1TreeNode Constructor	

[Top](#)

Public Properties





	Name	Description
	AccessKey	(Inherited from System.Web.UI.WebControls.WebControl)
	AllowDrag	
	AllowDrop	
	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from System.Web.UI.WebControls.WebControl)
	Checked	
	CheckState	
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	CollapsedIconClass	
	Controls	(Inherited from System.Web.UI.Control)
	ControlStyle	(Inherited from System.Web.UI.WebControls.WebControl)















	ControlStyleCreated	(Inherited from System.Web.UI.WebControls.WebControl)
	Expanded	
	ExpandedIconClass	
	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	ID	(Inherited from System.Web.UI.Control)
	ItemIconClass	
	NamingContainer	(Inherited from System.Web.UI.Control)
	Nodes	
	Owner	
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	Selected	
	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)
	Template	
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Text	
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	TreeView	
	UniqueID	(Inherited from System.Web.UI.Control)
	Url	
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	Value	
	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)
	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)
	IsSearchedUrl	
	MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	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)
	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

[C1TreeNode Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

C1TreeNode Constructor

Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1TreeNode()	

See Also

Reference



[C1TreeNode Class](#)

[C1TreeNode Members](#)

Methods

For a list of all members of this type, see [C1TreeNode 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)
⇒	GetRouteUrl	Overloaded. (Inherited from System.Web.UI.Control)
⇒	GetUniqueIDRelativeTo	(Inherited from System.Web.UI.Control)
⇒	HasControls	(Inherited from System.Web.UI.Control)
⇒	IsSearchedUrl	
⇒	MergeStyle	(Inherited from System.Web.UI.WebControls.WebControl)
⇒	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)
⇒	SetRenderMethodDelegate	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1TreeNode Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

IsSearchedUrl Method

Syntax

Visual Basic (Declaration)

```
Public Function IsSearchedUrl( _  
    ByVal searchedUrl As System.String, _  
    ByVal page As System.Web.UI.Page _  
) As System.Boolean
```

C#

```
public System.bool IsSearchedUrl(  
    System.string searchedUrl,  
    System.Web.UI.Page page  
)
```

Parameters

searchedUrl

page

See Also

Reference




[C1TreeNode Class](#)


















[C1TreeNode Members](#)


















Properties





For a list of all members of this type, see [C1TreeNode members](#).

Public Properties

	Name	Description
	AccessKey	(Inherited from System.Web.UI.WebControls.WebControl)
	AllowDrag	
	AllowDrop	

	AppRelativeTemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Attributes	(Inherited from System.Web.UI.WebControls.WebControl)
	Checked	
	CheckState	
	ClientID	(Inherited from System.Web.UI.Control)
	ClientIDMode	(Inherited from System.Web.UI.Control)
	CollapsedIconClass	
	Controls	(Inherited from System.Web.UI.Control)
	ControlStyle	(Inherited from System.Web.UI.WebControls.WebControl)
	ControlStyleCreated	(Inherited from System.Web.UI.WebControls.WebControl)
	Expanded	
	ExpandedIconClass	
	HasAttributes	(Inherited from System.Web.UI.WebControls.WebControl)
	ID	(Inherited from System.Web.UI.Control)
	ItemIconClass	
	NamingContainer	(Inherited from System.Web.UI.Control)
	Nodes	

	Owner	
	Page	(Inherited from System.Web.UI.Control)
	Parent	(Inherited from System.Web.UI.Control)
	RenderingCompatibility	(Inherited from System.Web.UI.Control)
	Selected	
	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)
	Template	
	TemplateControl	(Inherited from System.Web.UI.Control)
	TemplateSourceDirectory	(Inherited from System.Web.UI.Control)
	Text	
	ToolTip	(Inherited from System.Web.UI.WebControls.WebControl)
	TreeView	
	UniqueID	(Inherited from System.Web.UI.Control)

	Url	
	ValidateRequestMode	(Inherited from System.Web.UI.Control)
	Value	
	ViewStateMode	(Inherited from System.Web.UI.Control)

[Top](#)

See Also

Reference

[C1TreeNode Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

AllowDrag Property

Syntax

Visual Basic (Declaration)	
<pre><WidgetOptionAttribute()> <C1DescriptionAttribute(Key="C1TreeNode.AllowDrag", Description="Allow to node to perform drag.")> <C1CategoryAttribute("Behavior")> <System.ComponentModel.DefaultValueAttribute()> <LayoutAttribute(LayoutType.Behavior)> Public Property AllowDrag As System.Nullable(Of Boolean)</pre>	
C#	
<pre>[WidgetOption()] [C1Description(Key="C1TreeNode.AllowDrag", Description="Allow to node to perform drag.")] [C1Category("Behavior")] [System.ComponentModel.DefaultValue()] [Layout(LayoutType.Behavior)] public System.Nullable<bool> AllowDrag {get; set;}</pre>	

See Also

Reference

[C1TreeNode Class](#)[C1TreeNode Members](#)

AllowDrop Property

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeNode.AllowDrop", Description="Allow to
node to perform drop.")>
<WidgetOptionAttribute()>
<C1CategoryAttribute("Behavior")>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
Public Property AllowDrop As System.Nullable(Of Boolean)
```

C#

```
[C1Description(Key="C1TreeNode.AllowDrop", Description="Allow to node to
perform drop.")]
[WidgetOption()]
[C1Category("Behavior")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Behavior)]
public System.Nullable<bool> AllowDrop {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)[C1TreeNode Members](#)

Checked Property

Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeNode.Checked", Description="Gets a
value that indicates whether tree view node is checked or not.")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Behavior")>
```



```
<System.ComponentModel.LocalizableAttribute(True)>
```

```
Public Property Checked As System.Boolean
```

C#

```
[Layout(LayoutType.Behavior)]
```

```
[C1Description(Key="C1TreeNode.Checked", Description="Gets a value that indicates whether tree view node is checked or not.")]
```

```
[WidgetOption()]
```

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

```
[C1Category("Behavior")]
```

```
[System.ComponentModel.Localizable(true)]
```

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

See Also

Reference

[C1TreeNode Class](#)

[C1TreeNode Members](#)

CheckState Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
```

```
<C1CategoryAttribute("Behavior")>
```

```
<System.ComponentModel.LocalizableAttribute(True)>
```

```
<WidgetOptionAttribute()>
```

```
<C1DescriptionAttribute(Key="C1TreeNode.CheckState", Description="Gets a value that indicates treeview node's check state.")>
```

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

```
Public Property CheckState As C1TreeNodeCheckState
```

C#

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

```
[C1Category("Behavior")]
```

```
[System.ComponentModel.Localizable(true)]
```

```
[WidgetOption()]
```

```
[C1Description(Key="C1TreeNode.CheckState", Description="Gets a value that indicates treeview node's check state.")]
```

```
[Layout(LayoutType.Behavior)]
```

```
public C1TreeNodeCheckState CheckState {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)

[C1TreeNode Members](#)

CollapsedIconClass Property

Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1TreeNode.CollapsedIconClass",
Description="Set the icon class of the node when it's collapsed.")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property CollapsedIconClass As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[WidgetOption()]
[C1Description(Key="C1TreeNode.CollapsedIconClass", Description="Set the
icon class of the node when it's collapsed.")]
[System.ComponentModel.DefaultValue()]
public System.string CollapsedIconClass {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)

[C1TreeNode Members](#)

Expanded Property

Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<C1DescriptionAttribute(Key="C1TreeNode.Expanded", Description="Gets a
value that indicates whether the node is expanded.")>
```

```
<System.ComponentModel.DefaultValueAttribute(>>
<LayoutAttribute(LayoutType.Behavior)>
Public Property Expanded As System.Boolean
```

C#

```
[WidgetOption()]
[C1Description(Key="C1TreeNode.Expanded", Description="Gets a value that
indicates whether the node is expanded.")]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Behavior)]
public System.bool Expanded {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)

[C1TreeNode Members](#)

ExpandedIconClass Property

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeNode.ExpandedIconClass",
Description="Set the icon class of the node when it's expanded.")>
<System.ComponentModel.DefaultValueAttribute(>>
<WidgetOptionAttribute(>>
<LayoutAttribute(LayoutType.Appearance)>
Public Property ExpandedIconClass As System.String
```

C#

```
[C1Description(Key="C1TreeNode.ExpandedIconClass", Description="Set the
icon class of the node when it's expanded.")]
[System.ComponentModel.DefaultValue()]
[WidgetOption()]
[Layout(LayoutType.Appearance)]
public System.string ExpandedIconClass {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)[C1TreeNode Members](#)

ItemIconClass Property

Syntax

Visual Basic (Declaration)

```
<LayoutAttribute(LayoutType.Appearance)>
<System.ComponentModel.DefaultValueAttribute(>
<C1DescriptionAttribute(Key="C1TreeNode.ItemIconClass", Description="Set
the icon class of the node.")>
<WidgetOptionAttribute(>
Public Property ItemIconClass As System.String
```

C#

```
[Layout(LayoutType.Appearance)]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1TreeNode.ItemIconClass", Description="Set the icon
class of the node.")]
[WidgetOption()]
public System.string ItemIconClass {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)[C1TreeNode Members](#)

Nodes Property

Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute(>
<System.ComponentModel.NotifyParentPropertyAttribute(True)>
<System.ComponentModel.RefreshPropertiesAttribute(RefreshProperties.All)>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeriali
zationVisibility.Content)>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<System.ComponentModel.MergablePropertyAttribute(False)>
<LayoutAttribute(LayoutType.Behavior)>
```

```
<System.ComponentModel.BrowsableAttribute(False)>
Public ReadOnly Property Nodes As C1TreeViewNodeCollection
```

C#

```
[WidgetOption()]
[System.ComponentModel.NotifyParentProperty(true)]
[System.ComponentModel.RefreshProperties(RefreshProperties.All)]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[System.ComponentModel.MergableProperty(false)]
[Layout(LayoutType.Behavior)]
[System.ComponentModel.Browsable(false)]
public C1TreeViewNodeCollection Nodes {get;}
```

See Also

Reference

[C1TreeViewNode Class](#)

[C1TreeViewNode Members](#)

Owner Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(False)>
Public ReadOnly Property Owner As IC1TreeViewNodeCollectionOwner
```

C#

```
[System.ComponentModel.Browsable(false)]
public IC1TreeViewNodeCollectionOwner Owner {get;}
```

See Also

Reference

[C1TreeViewNode Class](#)

[C1TreeViewNode Members](#)

Selected Property

Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Behavior)>
<C1DescriptionAttribute(Key="C1TreeNode.Selected", Description="Gets or
sets a value that indicates whether tree view node is selected or not.")>
Public Property Selected As System.Boolean
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Behavior)]
[C1Description(Key="C1TreeNode.Selected", Description="Gets or sets a
value that indicates whether tree view node is selected or not.")]
public System.bool Selected {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)

[C1TreeNode Members](#)

Template Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BindableAttribute(Bindable=False,
Direction=BindingDirection.OneWay)>
<System.Web.UI.PersistenceModeAttribute(PersistenceMode.InnerProperty)>
<System.Web.UI.TemplateInstanceAttribute(TemplateInstance.Single)>
<System.ComponentModel.BrowsableAttribute(False)>
<C1DescriptionAttribute(Key="C1TreeNode.Template", Description="Gets or
sets the template that will be used for the specified C1TreeNodeNodes.")>
<System.ComponentModel.DesignerSerializationVisibilityAttribute(DesignerSeria
lizationVisibility.Content)>
Public Overridable Property Template As System.Web.UI.ITemplate
```

C#

```
[System.ComponentModel.Bindable(Bindable=false,
```

```
Direction=BindingDirection.OneWay))
[System.Web.UI.PersistenceMode(PersistenceMode.InnerProperty)]
[System.Web.UI.TemplateInstance(TemplateInstance.Single)]
[System.ComponentModel.Browsable(false)]
[C1Description(Key="C1TreeNode.Template", Description="Gets or sets the
template that will be used for the specified C1TreeNodees.")]
[System.ComponentModel.DesignerSerializationVisibility(DesignerSerializationV
isibility.Content)]
public virtual System.Web.UI.ITemplate Template {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)

[C1TreeNode Members](#)

Text Property

Syntax

Visual Basic (Declaration)

```
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
<C1DescriptionAttribute(Key="C1TreeNode.Text", Description="The text
displayed for the treeview node.")>
Public Property Text As System.String
```

C#

```
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
[C1Description(Key="C1TreeNode.Text", Description="The text displayed for
the treeview node.")]
public System.string Text {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)

[C1TreeNode Members](#)

TreeView Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.BrowsableAttribute(False)>
Public ReadOnly Property TreeView As C1TreeView
```

C#

```
[System.ComponentModel.Browsable(false)]
public C1TreeView TreeView {get;}
```

See Also

Reference

[C1TreeNode Class](#)

[C1TreeNode Members](#)

Url Property

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeNode.Url", Description="Gets or sets
the URL to navigate to when the node is clicked.")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Url As System.String
```

C#

```
[C1Description(Key="C1TreeNode.Url", Description="Gets or sets the URL to
navigate to when the node is clicked.")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
public System.string Url {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)[C1TreeNode Members](#)

Value Property

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeNode.Value", Description="The value of
the treeview node.")>
<WidgetOptionAttribute()>
<System.ComponentModel.DefaultValueAttribute()>
<LayoutAttribute(LayoutType.Appearance)>
Public Property Value As System.String
```

C#

```
[C1Description(Key="C1TreeNode.Value", Description="The value of the
treeview node.")]
[WidgetOption()]
[System.ComponentModel.DefaultValue()]
[Layout(LayoutType.Appearance)]
public System.string Value {get; set;}
```

See Also

Reference

[C1TreeNode Class](#)[C1TreeNode Members](#)

C1TreeNodeBinding

Object Model

C1TreeNodeBinding

Syntax

Visual Basic (Declaration)

```
<System.Web.AspNetHostingPermissionAttribute(Level=AspNetHostingPermissionLev
el.Minimal,
    Action=SecurityAction.Demand Or _
```

```

        SecurityAction.Deny Or _
        SecurityAction.LinkDemand,
        Unrestricted=False)>
Public Class C1TreeNodeBinding
C#
[System.Web.AspNetHostingPermission(Level=AspNetHostingPermissionLevel.Minima
1,
    Action=SecurityAction.Demand |
    SecurityAction.Deny |
    SecurityAction.LinkDemand,
    Unrestricted=false)]
public class C1TreeNodeBinding

```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1TreeView.C1TreeNodeBinding

See Also

Reference

[C1TreeNodeBinding Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

C1TreeNodeBinding

Syntax

```

Visual Basic (Declaration)
<System.Web.AspNetHostingPermissionAttribute(Level=AspNetHostingPermissionLev
el.Minimal,
    Action=SecurityAction.Demand Or _
    SecurityAction.Deny Or _
    SecurityAction.LinkDemand,
    Unrestricted=False)>
Public Class C1TreeNodeBinding

```

C#

```
[System.Web.AspNetHostingPermission(Level=AspNetHostingPermissionLevel.Minimal,
    Action=SecurityAction.Demand |
    SecurityAction.Deny |
    SecurityAction.LinkDemand,
    Unrestricted=false)]
public class C1TreeViewNodeBinding
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewNodeBinding

See Also


Reference

[C1TreeViewNodeBinding Members](#)[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members




The following tables list the members exposed by [C1TreeViewNodeBinding](#).









Public Constructors

	Name	Description
	C1TreeViewNodeBinding Constructor	Overloaded.

[Top](#)

Public Properties

	Name	Description
	CollapsedIconClassField	
	DataMember	
	Depth	

	EnabledField	
	ExpandedField	
	ExpandedIconClassField	
	FormatString	
	ItemIconClassField	
	NavigateUrlField	
	TextField	
	ValueField	

[Top](#)

See Also

Reference

[C1TreeNodeBinding Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

C1TreeNodeBinding Constructor

Overload List

Overload	Description
C1TreeNodeBinding Constructor()	
C1TreeNodeBinding Constructor(String)	

See Also

Reference

[C1TreeNodeBinding Class](#)

[C1TreeNodeBinding Members](#)

C1TreeNodeBinding Constructor()

Syntax

Visual Basic (Declaration)	
Public Function New()	
C#	
public C1TreeNodeBinding()	

See Also

Reference

[C1TreeNodeBinding Class](#)

[C1TreeNodeBinding Members](#)

[Overload List](#)

C1TreeNodeBinding Constructor(String)

Syntax

Visual Basic (Declaration)	
Public Function New(_ ByVal <i>dataMember</i> As System.String _)	
C#	
public C1TreeNodeBinding(System.string <i>dataMember</i>)	

Parameters

dataMember

See Also

Reference

[C1TreeNodeBinding Class](#)












[C1TreeNodeBinding Members](#)

[Overload List](#)

Properties

For a list of all members of this type, see [C1TreeViewNodeBinding members](#).

Public Properties

	Name	Description
	CollapsedIconClassField	
	DataMember	
	Depth	
	EnabledField	
	ExpandedField	
	ExpandedIconClassField	
	FormatString	
	ItemIconClassField	
	NavigateUrlField	
	TextField	
	ValueField	

[Top](#)

See Also

Reference

[C1TreeViewNodeBinding Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

CollapsedIconClassField Property

Syntax

Visual Basic (Declaration)

```
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Databindings")>
<C1DescriptionAttribute(Key="C1TreeViewNodeBinding.CollapsedIconClassField",
Description="Gets or sets the name of the field from the data source to bind
to the CollapsedIconClass property of a C1TreeViewNode object to which the
C1TreeViewNodeBinding object is applied.")>
Public Property CollapsedIconClassField As System.String
```

C#

```
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[System.ComponentModel.DefaultValue()]
[C1Category("Databindings")]
[C1Description(Key="C1TreeViewNodeBinding.CollapsedIconClassField",
Description="Gets or sets the name of the field from the data source to bind
to the CollapsedIconClass property of a C1TreeViewNode object to which the
C1TreeViewNodeBinding object is applied.")]
public System.string CollapsedIconClassField {get; set;}
```

See Also

Reference

[C1TreeViewNodeBinding Class](#)

[C1TreeViewNodeBinding Members](#)

DataMember Property

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Data")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1TreeViewNodeBinding.DataMember",
Description="Gets or sets the value to match against a IHierarchyData. Type
property for a data item to determine whether to apply the tree node
```

```
binding.")>
Public Property DataMember As System.String
```

C#

```
[C1Category("Data")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1TreeViewNodeBinding.DataMember", Description="Gets or
sets the value to match against a IHierarchyData. Type property for a data
item to determine whether to apply the tree node binding.")]
public System.string DataMember {get; set;}
```

See Also

Reference

[C1TreeViewNodeBinding Class](#)

[C1TreeViewNodeBinding Members](#)

Depth Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1TreeViewNodeBinding.Depth", Description="Gets
or sets the node depth at which the C1TreeViewNodeBinding object is
applied.")>
<TypeConverterAttribute("System.Web.UI.Design.WebControls.TreeNodeBindingDept
hConverter, System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<C1CategoryAttribute("Data")>
Public Property Depth As System.Integer
```

C#

```
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1TreeViewNodeBinding.Depth", Description="Gets or sets
the node depth at which the C1TreeViewNodeBinding object is applied.")]
[TypeConverter("System.Web.UI.Design.WebControls.TreeNodeBindingDepthConverte
r, System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[C1Category("Data")]
public System.int Depth {get; set;}
```


See Also

Reference

[C1TreeViewNodeBinding Class](#)

[C1TreeViewNodeBinding Members](#)

EnabledField Property

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeViewNodeBinding.EnabledField",
Description="Gets or sets the name of the field from the data source to bind
to the Enabled property of a C1TreeViewNode object to which the
C1TreeViewNodeBinding 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 EnabledField As System.String
```

C#

```
[C1Description(Key="C1TreeViewNodeBinding.EnabledField", Description="Gets or
sets the name of the field from the data source to bind to the Enabled
property of a C1TreeViewNode object to which the C1TreeViewNodeBinding 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 EnabledField {get; set;}
```

See Also

Reference

[C1TreeViewNodeBinding Class](#)

[C1TreeViewNodeBinding Members](#)

ExpandedField Property

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Databindings")>
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1TreeViewNodeBinding.ExpandedField",
Description="Gets or sets the name of the field from the data source to bind
to the Expanded property of a C1TreeViewNode object to which the
C1TreeViewNodeBinding object is applied.")>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
Public Property ExpandedField As System.String
```

C#

```
[C1Category("Databindings")]
[System.ComponentModel.DefaultValue()]
[C1Description(Key="C1TreeViewNodeBinding.ExpandedField", Description="Gets
or sets the name of the field from the data source to bind to the Expanded
property of a C1TreeViewNode object to which the C1TreeViewNodeBinding object
is applied.")]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
public System.string ExpandedField {get; set;}
```

See Also

Reference

[C1TreeViewNodeBinding Class](#)

[C1TreeViewNodeBinding Members](#)

ExpandedIconClassField Property

Syntax

Visual Basic (Declaration)

```
<C1DescriptionAttribute(Key="C1TreeViewNodeBinding.ExpandedIconClassField",
Description="Gets or sets the name of the field from the data source to bind
```

```
to the ExpandedIconClass property of a C1TreeNode object to which the
C1TreeNodeBinding object is applied.")>
<C1CategoryAttribute("Databindings")>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property ExpandedIconClassField As System.String
```

C#

```
[C1Description(Key="C1TreeNodeBinding.ExpandedIconClassField",
Description="Gets or sets the name of the field from the data source to bind
to the ExpandedIconClass property of a C1TreeNode object to which the
C1TreeNodeBinding object is applied.")]
[C1Category("Databindings")]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[System.ComponentModel.DefaultValue()]
public System.string ExpandedIconClassField {get; set;}
```

See Also

Reference

[C1TreeNodeBinding Class](#)

[C1TreeNodeBinding Members](#)

FormatString Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<C1DescriptionAttribute(Key="C1TreeNodeBinding.FormatString",
Description="Format String.")>
<C1CategoryAttribute("Databindings")>
<System.ComponentModel.LocalizableAttribute(True)>
Public Property FormatString As System.String
```

C#

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

```
[C1Description(Key="C1TreeNodeBinding.FormatString", Description="Format
String.")]
[C1Category("Databindings")]
[System.ComponentModel.Localizable(true)]
public System.string FormatString {get; set;}
```

See Also

Reference

[C1TreeNodeBinding Class](#)

[C1TreeNodeBinding Members](#)

ItemIconClassField Property

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Databindings")>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<C1DescriptionAttribute(Key="C1TreeNodeBinding.ItemIconClassField",
Description="Gets or sets the name of the field from the data source to bind
to the ItemIconClass property of a C1TreeNode object to which the
C1TreeNodeBinding object is applied.")>
<System.ComponentModel.DefaultValueAttribute()>
Public Property ItemIconClassField As System.String
```

C#

```
[C1Category("Databindings")]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[C1Description(Key="C1TreeNodeBinding.ItemIconClassField",
Description="Gets or sets the name of the field from the data source to bind
to the ItemIconClass property of a C1TreeNode object to which the
C1TreeNodeBinding object is applied.")]
[System.ComponentModel.DefaultValue()]
public System.string ItemIconClassField {get; set;}
```

See Also

Reference

[C1TreeNodeBinding Class](#)[C1TreeNodeBinding Members](#)

NavigateUrlField Property

Syntax

Visual Basic (Declaration)

```
<C1CategoryAttribute("Databindings")>
<C1DescriptionAttribute(Key="C1TreeNodeBinding.NavigateUrlField",
Description="Gets or sets the name of the field from the data source to bind
to the Url property of a C1TreeNode object to which the
C1TreeNodeBinding object is applied.")>
<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#

```
[C1Category("Databindings")]
[C1Description(Key="C1TreeNodeBinding.NavigateUrlField",
Description="Gets or sets the name of the field from the data source to bind
to the Url property of a C1TreeNode object to which the
C1TreeNodeBinding object is applied.")]
[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

[C1TreeNodeBinding Class](#)[C1TreeNodeBinding Members](#)

TextField Property

Syntax

Visual Basic (Declaration)

```
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<C1DescriptionAttribute(Key="C1TreeViewNodeBinding.TextField",
Description="Gets or sets the name of the field from the data source to bind
to the Text property of a C1TreeViewNode object to which the
C1TreeViewNodeBinding object is applied.")>
<System.ComponentModel.DefaultValueAttribute()>
<C1CategoryAttribute("Databindings")>
Public Property TextField As System.String
```

C#

```
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[C1Description(Key="C1TreeViewNodeBinding.TextField", Description="Gets or
sets the name of the field from the data source to bind to the Text property
of a C1TreeViewNode object to which the C1TreeViewNodeBinding object is
applied.")]
[System.ComponentModel.DefaultValue()]
[C1Category("Databindings")]
public System.string TextField {get; set;}
```

See Also

Reference

[C1TreeViewNodeBinding Class](#)

[C1TreeViewNodeBinding Members](#)

ValueField Property

Syntax

Visual Basic (Declaration)

```
<System.ComponentModel.DefaultValueAttribute()>
<TypeConverterAttribute("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")>
<C1DescriptionAttribute(Key="C1TreeViewNodeBinding.ValueField",
Description="Gets or sets the name of the field from the data source to bind
to the Value property of a C1TreeViewNode object to which the
```

```
C1TreeNodeBinding object is applied.")>
<C1CategoryAttribute("Databindings")>
Public Property ValueField As System.String
```

C#

```
[System.ComponentModel.DefaultValue()]
[TypeConverter("System.Web.UI.Design.DataSourceViewSchemaConverter,
System.Design, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a")]
[C1Description(Key="C1TreeNodeBinding.ValueField", Description="Gets or
sets the name of the field from the data source to bind to the Value property
of a C1TreeNode object to which the C1TreeNodeBinding object is
applied.")]
[C1Category("Databindings")]
public System.string ValueField {get; set;}
```

See Also

Reference

[C1TreeNodeBinding Class](#)

[C1TreeNodeBinding Members](#)

C1TreeNodeBindingCollection

Object Model

C1TreeNodeBindingCollection

Syntax

Visual Basic (Declaration)

```
<System.Reflection.DefaultMemberAttribute("Item")>
<System.Web.AspNetHostingPermissionAttribute(Level=AspNetHostingPermissionLev
el.Minimal,
    Action=SecurityAction.Demand Or _
    SecurityAction.Deny Or _
    SecurityAction.LinkDemand,
    Unrestricted=False)>
Public Class C1TreeNodeBindingCollection
    Inherits System.Web.UI.StateManagedCollection
```

C#

```
[System.Reflection.DefaultMember("Item")]
[System.Web.AspNetHostingPermission(Level=AspNetHostingPermissionLevel.Minimal,
    Action=SecurityAction.Demand |
        SecurityAction.Deny |
        SecurityAction.LinkDemand,
    Unrestricted=false)]
public class C1TreeViewNodeBindingCollection :
    System.Web.UI.StateManagedCollection
```

Inheritance Hierarchy

System.Object

System.Web.UI.StateManagedCollection

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewNodeBindingCollection

See Also

Reference

[C1TreeViewNodeBindingCollection Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

C1TreeViewNodeBindingCollection

Syntax

Visual Basic (Declaration)

```
<System.Reflection.DefaultMemberAttribute("Item")>
<System.Web.AspNetHostingPermissionAttribute(Level=AspNetHostingPermissionLevel.Minimal,
    Action=SecurityAction.Demand Or _
        SecurityAction.Deny Or _
        SecurityAction.LinkDemand,
    Unrestricted=False)>
Public Class C1TreeViewNodeBindingCollection
    Inherits System.Web.UI.StateManagedCollection
```


C#

```
[System.Reflection.DefaultMember("Item")]
[System.Web.AspNetHostingPermission(Level=AspNetHostingPermissionLevel.Minima
1,
    Action=SecurityAction.Demand |
    SecurityAction.Deny |
    SecurityAction.LinkDemand,
    Unrestricted=false)]
public class C1TreeViewNodeBindingCollection :
System.Web.UI.StateManagedCollection
```

Inheritance Hierarchy

System.Object

System.Web.UI.StateManagedCollection

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewNodeBindingCollection

See Also



Reference

[C1TreeViewNodeBindingCollection Members](#)[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members

The following tables list the members exposed by [C1TreeViewNodeBindingCollection](#).









Public Properties

	Name	Description
	Count	(Inherited from System.Web.UI.StateManagedCollection)
	Item	

[Top](#)

Public Methods

	Name	Description
	Add	

	Clear	(Inherited from System.Web.UI.StateManagedCollection)
	Contains	
	CopyTo	Overloaded.
	IndexOf	
	Insert	
	Remove	
	RemoveAt	
	SetDirty	(Inherited from System.Web.UI.StateManagedCollection)

[Top](#)

See Also

Reference





[C1TreeNodeBindingCollection Class](#)


[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Methods

For a list of all members of this type, see [C1TreeNodeBindingCollection members](#).

Public Methods

	Name	Description
	Add	
	Clear	(Inherited from System.Web.UI.StateManagedCollection)
	Contains	
	CopyTo	Overloaded.

	IndexOf	
	Insert	
	Remove	
	RemoveAt	
	SetDirty	(Inherited from System.Web.UI.StateManagedCollection)

[Top](#)

See Also

Reference

[C1TreeNodeBindingCollection Class](#)[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Add Method

Syntax

Visual Basic (Declaration)

```
Public Function Add( _  
    ByVal binding As C1TreeNodeBinding _  
) As System.Integer
```

C#

```
public System.int Add(  
    C1TreeNodeBinding binding  
)
```

Parameters

binding

See Also

Reference

[C1TreeNodeBindingCollection Class](#)[C1TreeNodeBindingCollection Members](#)

Contains Method

Syntax

Visual Basic (Declaration)	
<pre>Public Function Contains(_ ByVal <i>binding</i> As C1TreeNodeBinding _) As System.Boolean</pre>	
C#	
<pre>public System.bool Contains(C1TreeNodeBinding <i>binding</i>)</pre>	

Parameters

binding

See Also

Reference

[C1TreeNodeBindingCollection Class](#)
[C1TreeNodeBindingCollection Members](#)

CopyTo Method

Overload List

Overload	Description
CopyTo(C1TreeNodeBinding[],Int32)	
CopyTo(Array,Int32)	(Inherited from System.Web.UI.StateManagedCollection)

See Also

Reference

[C1TreeNodeBindingCollection Class](#)
[C1TreeNodeBindingCollection Members](#)

CopyTo(C1TreeViewNodeBinding[],Int32) Method

Syntax

Visual Basic (Declaration)

```
Public Overloads Sub CopyTo( _
    ByVal bindingArray() As C1TreeViewNodeBinding, _
    ByVal index As System.Integer _
)
```

C#

```
public void CopyTo(
    C1TreeViewNodeBinding[] bindingArray,
    System.int index
)
```

Parameters*bindingArray**index***See Also****Reference**[C1TreeViewNodeBindingCollection Class](#)[C1TreeViewNodeBindingCollection Members](#)[Overload List](#)

IndexOf Method

Syntax

Visual Basic (Declaration)

```
Public Function IndexOf( _
    ByVal binding As C1TreeViewNodeBinding _
) As System.Integer
```

C#

```
public System.int IndexOf(
    C1TreeViewNodeBinding binding
)
```

Parameters

binding

See Also

Reference

[C1TreeViewNodeBindingCollection Class](#)

[C1TreeViewNodeBindingCollection Members](#)

Insert Method

Syntax

Visual Basic (Declaration)	
<pre>Public Sub Insert(_ ByVal index As System.Integer, _ ByVal binding As C1TreeViewNodeBinding _)</pre>	
C#	
<pre>public void Insert(System.int index, C1TreeViewNodeBinding binding)</pre>	

Parameters

index

binding

See Also

Reference

[C1TreeViewNodeBindingCollection Class](#)

[C1TreeViewNodeBindingCollection Members](#)

Remove Method

Syntax

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

```
Public Sub Remove( _  
    ByVal binding As C1TreeNodeBinding _  
)
```

C#

```
public void Remove(  
    C1TreeNodeBinding binding  
)
```

Parameters

binding

See Also

Reference

[C1TreeNodeBindingCollection Class](#)

[C1TreeNodeBindingCollection Members](#)

RemoveAt Method

Syntax

Visual Basic (Declaration)

```
Public Sub RemoveAt( _  
    ByVal index As System.Integer _  
)
```

C#

```
public void RemoveAt(  
    System.int index  
)
```

Parameters

index

See Also

Reference



[C1TreeNodeBindingCollection Class](#)

[C1TreeNodeBindingCollection Members](#)

Properties

For a list of all members of this type, see [C1TreeViewNodeBindingCollection members](#).

Public Properties

	Name	Description
	Count	(Inherited from System.Web.UI.StateManagedCollection)
	Item	

[Top](#)

See Also

Reference

[C1TreeViewNodeBindingCollection Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Item Property

Syntax

Visual Basic (Declaration)	
<pre>Public Default Property Item(_ ByVal i As System.Integer _) As C1TreeViewNodeBinding</pre>	
C#	
<pre>public C1TreeViewNodeBinding this[System.int i]; {get; set;}</pre>	

Parameters

i

See Also

Reference

[C1TreeViewNodeBindingCollection Class](#)

[C1TreeViewNodeBindingCollection Members](#)

C1TreeViewNodeCollection

Object Model

C1TreeViewNodeCollection

Syntax

Visual Basic (Declaration)

```
Public Class C1TreeViewNodeCollection
    Inherits
    C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection(Of
    IC1TreeViewNodeCollectionOwner, C1TreeViewNode)
```

C#

```
public class C1TreeViewNodeCollection :
    C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection<IC1TreeView
    NodeCollectionOwner, C1TreeViewNode>
```

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.Collection<T>

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewNodeCollection

See Also

Reference

[C1TreeViewNodeCollection Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

C1TreeViewNodeCollection

Syntax

Visual Basic (Declaration)

```
Public Class C1TreeViewNodeCollection
```

Inherits

C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection(Of IC1TreeViewNodeCollectionOwner, C1TreeViewNode)

C#

```
public class C1TreeViewNodeCollection :
C1.Web.Wijmo.Controls.Base.Collections.C1ObservableItemCollection<IC1TreeView
NodeCollectionOwner, C1TreeViewNode>
```

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.Collection<T>

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewNodeCollection

See Also

Reference

[C1TreeViewNodeCollection Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members


The following tables list the members exposed by [C1TreeViewNodeCollection](#).

Public Constructors

	Name	Description
	C1TreeViewNodeCollection Constructor	













[Top](#)

Public Properties

	Name	Description
	Item	(Inherited from System.Collections.ObjectModel.Collection<C1TreeViewNode>)

[Top](#)

Public Methods

	Name	Description
	Add	
	CheckNodes	
	Clear	(Inherited from System.Collections.ObjectModel.Collection<C1TreeViewNode>)
	Contains	(Inherited from System.Collections.ObjectModel.Collection<C1TreeViewNode>)
	CopyTo	(Inherited from System.Collections.ObjectModel.Collection<C1TreeViewNode>)
	FindNode	
	FindNodeByText	
	FindNodeByValue	
	IndexOf	(Inherited from System.Collections.ObjectModel.Collection<C1TreeViewNode>)
	Insert	
	Remove	
	RemoveAt	(Inherited from System.Collections.ObjectModel.Collection<C1TreeViewNode>)

[Top](#)

See Also

Reference

[C1TreeViewNodeCollection Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

C1TreeViewNodeCollection Constructor

Syntax

Visual Basic (Declaration)

```
Public Function New( _  
    ByVal owner As IC1TreeViewNodeCollectionOwner _  
)
```

C#

```
public C1TreeViewNodeCollection(  
    IC1TreeViewNodeCollectionOwner owner  
)
```

Parameters

owner

See Also

Reference





[C1TreeViewNodeCollection Class](#)









[C1TreeViewNodeCollection Members](#)

Methods

For a list of all members of this type, see [C1TreeViewNodeCollection members](#).

Public Methods

	Name	Description
	Add	
	CheckNodes	
	Clear	(Inherited from System.Collections.ObjectModel.Collection<C1TreeViewNode>)
	Contains	(Inherited from System.Collections.ObjectModel.Collection<C1TreeViewNode>)

	CopyTo	(Inherited from System.Collections.ObjectModel.Collection<C1TreeNode>)
	FindNode	
	FindNodeByText	
	FindNodeByValue	
	IndexOf	(Inherited from System.Collections.ObjectModel.Collection<C1TreeNode>)
	Insert	
	Remove	
	RemoveAt	(Inherited from System.Collections.ObjectModel.Collection<C1TreeNode>)

[Top](#)

See Also

Reference

[C1TreeNodeCollection Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Add Method

Syntax

Visual Basic (Declaration)

```
Public Shadows Sub Add( _  
    ByVal node As C1TreeNode _  
)
```

C#

```
public new void Add(  
    C1TreeNode node  
)
```

Parameters

node

See Also

Reference

[C1TreeViewNodeCollection Class](#)

[C1TreeViewNodeCollection Members](#)

CheckNodes Method

Syntax

Visual Basic (Declaration)	
<pre>Public Sub CheckNodes(_ ByVal <i>check</i> As System.Boolean _)</pre>	
C#	
<pre>public void CheckNodes(System.bool <i>check</i>)</pre>	

Parameters

check

See Also

Reference

[C1TreeViewNodeCollection Class](#)

[C1TreeViewNodeCollection Members](#)

FindNode Method

Syntax

Visual Basic (Declaration)	
<pre>Public Function FindNode(_ ByVal <i>id</i> As System.String _) As C1TreeViewNode</pre>	

C#

```
public C1TreeNode FindNode(  
    System.string id  
)
```

Parameters*id*

See Also

Reference[C1TreeNodeCollection Class](#)[C1TreeNodeCollection Members](#)

FindNodeByText Method

Syntax

Visual Basic (Declaration)

```
Public Function FindNodeByText( _  
    ByVal text As System.String _  
) As C1TreeNode
```

C#

```
public C1TreeNode FindNodeByText(  
    System.string text  
)
```

Parameters*text*

See Also

Reference[C1TreeNodeCollection Class](#)[C1TreeNodeCollection Members](#)

FindNodeByValue Method

Syntax

Visual Basic (Declaration)

```
Public Function FindNodeByValue( _  
    ByVal value As System.String _  
) As C1TreeNode
```

C#

```
public C1TreeNode FindNodeByValue(  
    System.string value  
)
```

Parameters

value

See Also

Reference

[C1TreeNodeCollection Class](#)

[C1TreeNodeCollection Members](#)

Insert Method

Syntax

Visual Basic (Declaration)

```
Public Shadows Sub Insert( _  
    ByVal index As System.Integer, _  
    ByVal node As C1TreeNode _  
)
```

C#

```
public new void Insert(  
    System.int index,  
    C1TreeNode node  
)
```

Parameters

index

node

See Also

Reference

[C1TreeViewNodeCollection Class](#)

[C1TreeViewNodeCollection Members](#)

Remove Method

Syntax

Visual Basic (Declaration)

```
Public Shadows Sub Remove( _  
    ByVal node As C1TreeViewNode _  
)
```

C#

```
public new void Remove(  
    C1TreeViewNode node  
)
```

Parameters

node

See Also

Reference

[C1TreeViewNodeCollection Class](#)

[C1TreeViewNodeCollection Members](#)

C1TreeViewNodeDroppedEventArgs

Object Model

C1TreeViewNodeDroppedEventArgs

Syntax

Visual Basic (Declaration)

```
Public Class C1TreeViewNodeDroppedEventArgs  
    Inherits C1TreeViewEventArgs
```

C#

```
public class C1TreeNodeDroppedEventArgs : C1TreeViewEventArgs
```

Inheritance Hierarchy

System.Object

System.EventArgs

[C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewEventArgs](#)

C1.Web.Wijmo.Controls.C1TreeView.C1TreeNodeDroppedEventArgs

See Also

Reference

[C1TreeNodeDroppedEventArgs Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

C1TreeNodeDroppedEventArgs

Syntax

Visual Basic (Declaration)

```
Public Class C1TreeNodeDroppedEventArgs  
    Inherits C1TreeViewEventArgs
```

C#

```
public class C1TreeNodeDroppedEventArgs : C1TreeViewEventArgs
```

Inheritance Hierarchy

System.Object

System.EventArgs

[C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewEventArgs](#)

C1.Web.Wijmo.Controls.C1TreeView.C1TreeNodeDroppedEventArgs

See Also

Reference

[C1TreeNodeDroppedEventArgs Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members




The following tables list the members exposed by [C1TreeViewNodeDroppedEventArgs](#).

Public Constructors

	Name	Description
	C1TreeViewNodeDroppedEventArgs Constructor	

[Top](#)

Public Properties

	Name	Description
	CopyUsed	
	Handled	
	Node	(Inherited from C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewEventArgs)

[Top](#)

See Also

Reference

[C1TreeViewNodeDroppedEventArgs Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

C1TreeViewNodeDroppedEventArgs Constructor

Syntax

Visual Basic (Declaration)	
<pre>Public Function New(_ ByVal <i>node</i> As C1TreeViewNode, _ ByVal <i>handled</i> As System.Boolean, _ ByVal <i>copyUsed</i> As System.Boolean _)</pre>	

C#

```
public C1TreeNodeDroppedEventArgs(  
    C1TreeNode node,  
    System.bool handled,  
    System.bool copyUsed  
)
```

Parameters

node

handled

copyUsed

See Also

Reference




[C1TreeNodeDroppedEventArgs Class](#)

[C1TreeNodeDroppedEventArgs Members](#)

Properties

For a list of all members of this type, see [C1TreeNodeDroppedEventArgs members](#).

Public Properties

	Name	Description
	CopyUsed	
	Handled	
	Node	(Inherited from C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewEventArgs)

[Top](#)

See Also

Reference

[C1TreeNodeDroppedEventArgs Class](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

CopyUsed Property

Syntax

Visual Basic (Declaration)	
Public ReadOnly Property CopyUsed As System.Boolean	
C#	
public System.bool CopyUsed {get;}	

See Also

Reference

[C1TreeNodeDroppedEventArgs Class](#)

[C1TreeNodeDroppedEventArgs Members](#)

Handled Property

Syntax

Visual Basic (Declaration)	
Public ReadOnly Property Handled As System.Boolean	
C#	
public System.bool Handled {get;}	

See Also

Reference

[C1TreeNodeDroppedEventArgs Class](#)

[C1TreeNodeDroppedEventArgs Members](#)

C1TreeViewSerializer

Object Model

C1TreeViewSerializer

Syntax

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

```
Public Class C1TreeViewSerializer
    Inherits C1.Web.Wijmo.Controls.Base.C1BaseSerializer(Of
C1TreeView,C1TreeViewNode,IC1TreeViewNodeCollectionOwner)
```

C#

```
public class C1TreeViewSerializer :
C1.Web.Wijmo.Controls.Base.C1BaseSerializer<C1TreeView,C1TreeViewNode,IC1Tree
ViewNodeCollectionOwner>
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewSerializer

See Also

Reference

[C1TreeViewSerializer Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

C1TreeViewSerializer

Syntax

Visual Basic (Declaration)

```
Public Class C1TreeViewSerializer
    Inherits C1.Web.Wijmo.Controls.Base.C1BaseSerializer(Of
C1TreeView,C1TreeViewNode,IC1TreeViewNodeCollectionOwner)
```

C#

```
public class C1TreeViewSerializer :
C1.Web.Wijmo.Controls.Base.C1BaseSerializer<C1TreeView,C1TreeViewNode,IC1Tree
ViewNodeCollectionOwner>
```

Inheritance Hierarchy

System.Object

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewSerializer

See Also

Reference

[C1TreeViewSerializer Members](#)
[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members

The following tables list the members exposed by [C1TreeViewSerializer](#).

Public Constructors

	Name	Description
	C1TreeViewSerializer Constructor	

[Top](#)

See Also

Reference

[C1TreeViewSerializer Class](#)
[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

C1TreeViewSerializer Constructor

Syntax

Visual Basic (Declaration)	
<pre>Public Function New(_ ByVal obj As System.Object _)</pre>	
C#	
<pre>public C1TreeViewSerializer(System.object obj)</pre>	

Parameters

obj

See Also

Reference

[C1TreeViewSerializer Class](#)

[C1TreeViewSerializer Members](#)

Enumerations

C1TreeViewNodeCheckState

Syntax

Visual Basic (Declaration)	
<pre>Public Enum C1TreeViewNodeCheckState Inherits System.Enum</pre>	
C#	
<pre>public enum C1TreeViewNodeCheckState : System.Enum</pre>	

Members

Member	Description
Checked	
Indeterminate	
Unchecked	

Inheritance Hierarchy

System.Object

System.ValueType

System.Enum

C1.Web.Wijmo.Controls.C1TreeView.C1TreeViewNodeCheckState

See Also

Reference

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Interfaces

IC1TreeViewNodeCollectionOwner

Object Model

IC1TreeViewNodeCollectionOwner

Syntax

Visual Basic (Declaration)

```
Public Interface IC1TreeViewNodeCollectionOwner
```

C#

```
public interface IC1TreeViewNodeCollectionOwner
```

See Also

Reference

[IC1TreeViewNodeCollectionOwner Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Overview

Object Model

IC1TreeViewNodeCollectionOwner

Syntax

Visual Basic (Declaration)

```
Public Interface IC1TreeViewNodeCollectionOwner
```

C#

```
public interface IC1TreeViewNodeCollectionOwner
```

See Also

Reference

[IC1TreeViewNodeCollectionOwner Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Members

The following tables list the members exposed by [IC1TreeViewNodeCollectionOwner](#).

Public Properties

	Name	Description
	Nodes	
	Owner	

[Top](#)

See Also

Reference

[IC1TreeViewNodeCollectionOwner Interface](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Properties

For a list of all members of this type, see [IC1TreeViewNodeCollectionOwner members](#).

Public Properties

	Name	Description
	Nodes	
	Owner	

[Top](#)

See Also

Reference

[IC1TreeViewNodeCollectionOwner Interface](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)

Nodes Property

Syntax

Visual Basic (Declaration)	
ReadOnly Property Nodes As C1TreeViewNodeCollection	
C#	
C1TreeViewNodeCollection Nodes {get;}	

See Also

Reference

[IC1TreeViewNodeCollectionOwner Interface](#)

[IC1TreeViewNodeCollectionOwner Members](#)

Owner Property

Syntax

Visual Basic (Declaration)	
ReadOnly Property Owner As IC1TreeViewNodeCollectionOwner	
C#	
IC1TreeViewNodeCollectionOwner Owner {get;}	

See Also

Reference

[IC1TreeViewNodeCollectionOwner Interface](#)

[IC1TreeViewNodeCollectionOwner Members](#)

Delegates

C1TreeViewEventHandler

Syntax

Visual Basic (Declaration)	
Public Delegate Sub C1TreeViewEventHandler(_	

```
ByVal sender As System.Object, _  
    ByVal e As C1TreeViewEventArgs _  
)
```

C#

```
public delegate void C1TreeViewEventHandler(  
    System.Object sender,  
    C1TreeViewEventArgs e  
)
```

Parameters

sender

e

See Also

Reference

[C1TreeViewEventHandler Members](#)

[C1.Web.Wijmo.Controls.C1TreeView Namespace](#)