

The CMSTreeView control displays a multi-level tree menu based on page data from Kentico.

Derived from: [ASP.NET TreeView](#)
Web part equivalent (portal engine): Tree view

Getting started

The following is a step-by-step tutorial that shows how to display a tree menu based on Kentico content using the CMSTreeView control:

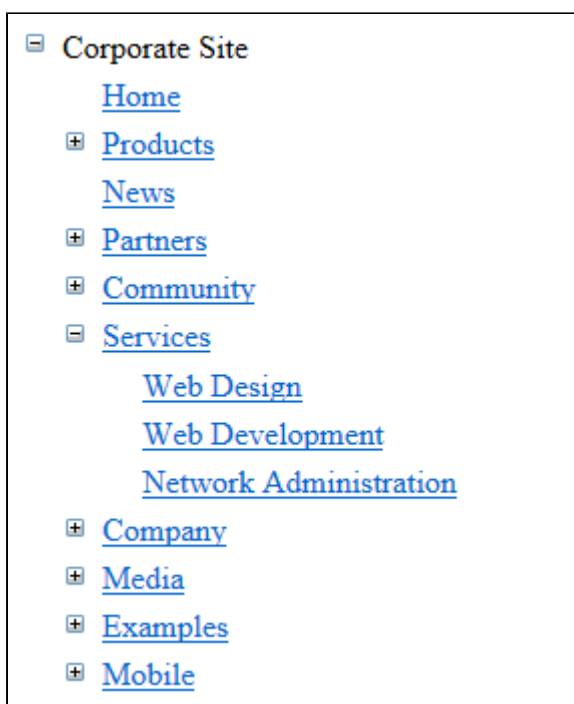
1. Create a new **Web form** somewhere in your web project.
2. Drag the **CMSTreeView** control from the toolbox onto the form.

- The code of the CMSTreeView looks like this:

```
<cms:CMSTreeView ID="CMSTreeView1" runat="server" />
```

3. Save the web form.
4. Right-click the web form in the Solution explorer and select **View in Browser**.

The resulting page displays a tree menu.



Configuration

You can set the following properties for the CMSTreeView control:

CMSTreeView properties	Description	Sample value
DisplayDocumentTypeImages	If true, control loads the images displayed for items in the tree according to the page types of the items.	

DynamicBehaviour	<p>If true, the control dynamically loads sub-items when their parent node is expanded.</p> <p>This is recommended if there is a large amount of nodes in the tree, or if the content depends on other dynamic parameters.</p>	
EncodeMenuCaption	Indicates whether the control HTML encodes the captions of menu items. Enable if you need to display pages whose names contain HTML code.	
ExpandAllOnStartup	Indicates whether all nodes in the tree are expanded by default.	
ExpandCurrentPath	Indicates whether the control highlights all nodes along the path to the currently selected item by default.	
ExpandSubTree	Indicates whether the sub-tree under the currently selected item is expanded by default.	
FixBrokenLines	Indicates whether the control fixes broken lines.	
HideRootNode	Indicates whether the root node of the tree is hidden.	
HighlightSelectedItem	Indicates whether the currently selected item is highlighted.	
IgnoreDocumentMenuAction	<p>If true, the control ignores the Menu actions of pages. You can access the menu settings of individual pages in the Kentico administration interface (Pages -> Edit -> Properties -> Navigation).</p> <p>Enable this property to use the JavaScript action defined by the OnClickAction property.</p>	
InactivateAllItemsInPath	If enabled, all nodes on the path of the currently selected page are inactive (will not perform any action when clicked).	
InactivateSelectedItem	If enabled, the node of the currently selected page is inactive.	
InactiveNodeImage	Indicates if the images displayed next to nodes also perform the specified action for the given item when clicked.	
InactiveRoot	Indicates whether the root node is inactive.	
NodeImageUrl	Gets or sets the path to the image that the control displays next to regular nodes in the tree.	~ /App_Themes/ /Site/Images/ /treeitem.gif
OnClickAction	Gets or sets the JavaScript command that the control executes when a user clicks on an item in the tree. If empty, the items in the tree serve as links to the corresponding pages.	
OrderBy	<p>Gets or sets the ORDER BY clause of the SQL query that the control uses to load data.</p> <p>Important: The root of the displayed page tree (or sub-tree) must be first in the resulting order, otherwise the control may not display all pages correctly. You can ensure this by starting the Order by value with the NodeLevel column.</p>	"NodeLevel, NodeOrder"
RootImageUrl	Gets or sets the path to the image that the control displays next to the root node of the tree.	~ /App_Themes/ /Site/Images/ /root.gif
RootText	Gets or sets the text caption of the root node in the tree. This text is displayed instead of the page name of the item in the root of the tree.	

Common navigation properties	Description	Sample value
ApplyMenuDesign	Indicates whether the control applies page menu design settings . True by default.	
Columns	<p>Contains the names of columns that the control loads for pages (menu items) in addition to the default columns. If you need data from other columns, type their names separated by commas.</p> <p>Note: To find a full list of the default navigation columns, use the SQL queries debugging tool and inspect the query performed by your navigation control.</p>	"Document PageTitle, Document PageKeywords"
HideControlForZeroRows	Indicates whether the control should be hidden when no data is loaded. Default value is False.	
HighlightAllItemsInPath	Indicates whether the control highlights all items on the visitor's current path.	
SubmenuIndicator	Path to an image displayed next to every menu item that contains sub-items.	
UseAlternatingStyles	Indicates whether the control uses alternating styles for even and odd items on the same menu level.	
UseItemImagesForHighlightedItem	Indicates whether the control uses the item image if the highlighted image is not specified.	
WordWrap	Indicates whether text displayed by the control uses word wrapping. If disabled, text that is too long is replaced by 'nbsp'.	
ZeroRowsText	Text shown if no records are found. This text is not visible when the control is hidden by the HideControlForZeroRows property.	"No records found."

Page filtering properties	Description	Sample value
CheckPermissions	<p>Indicates if the control checks the permissions of the user viewing the page. If the value is <i>false</i> (default value) no permissions are checked.</p> <p>If true, the control only loads pages for which the user viewing the page has read permissions.</p>	
ClassNames	<p>Specifies which page types the control loads and displays. Identify page types through their code names, separated by semicolons (;).</p> <p>You can use the * wildcard as a substitute for any number of characters. For example <i>Product.*</i> includes the page types <i>Product.Camera</i>, <i>Product.CellPhone</i>, <i>Product.Computer</i> etc.</p> <p>If the property is left empty, the control retrieves all page types by default. In the case of menu and navigation controls, only <i>CMS.Menuitem</i> pages are loaded by default.</p> <p>Note: If the control loads all page types (empty value), only the data from the <i>View_CMS_Tree_Joined</i> and the <i>COM_SKU</i> table (for product pages) are available in the retrieved data. The specific fields of individual page types are not included. You need to keep this in mind when writing the code of transformations, WHERE conditions, ORDER BY expressions etc.</p>	"cms. news" "cms. news; cms. article"

CombineWithDefaultCulture	Indicates whether the control loads pages from the website's default culture version if the required pages are not available in the user's selected culture. Only applies if you do not set the TreeProvider property manually.	
CultureCode	Specifies the culture code of the pages that the control loads. If not specified, the control automatically uses the preferred culture of the user viewing the page.	"en-us"
DataSource	Allows you to manually assign a DataSet or DataTable containing the pages that the control displays. You do not need to set this property for standard scenarios.	
FilterOutDuplicates	Indicates if the control filters out duplicated (linked) pages from the data.	
MaxRelativeLevel	Specifies the maximum number of content tree sub-levels from which the control displays pages. This number is relative, i.e. counted from the location of the page where the control is placed, not from the root of the website. Enter -1 to load all child pages.	
Path	Path of the pages that the control loads. See: Writing page path expressions	
SelectOnlyPublished	If enabled, the control only loads published pages.	
TreeProvider	Gets or sets the <i>TreeProvider</i> object used by the control to access page data. If you do not assign a TreeProvider object, the control automatically creates a new instance.	

CMS Base control properties	Description	Sample value
CacheDependencies	List of the cache keys on which the control's cached data depends. When the specified cache items change, the control clears its cache. Each item (dependency) must be on one line. If you leave this property empty, the control uses default dependencies. See also: Setting cache dependencies , Configuring caching	cms. user all
CacheItemName	Sets the name of the cache key used to store the control's content. If you leave the value empty, the system generates a default name containing variables, such as the control ID, the selected culture and the name of the user who loaded the page. The system cache is shared by all pages in your application, so cache item names representing different data must be unique globally. If you have multiple controls that load the same data, you can share the cache keys between the controls (optimizes loading of content and avoids redundant data in the cache). If the content displayed by the control depends on variables, such as URL parameters, you can set a custom name dynamically in the page's code behind. See also: Caching the data of page components , Configuring caching	"CMSRepeaterNews" + Request.QueryString["id"].ToString()

CacheMinutes	<p>Sets the number of minutes for which the control caches content retrieved from the database.</p> <ul style="list-style-type: none"> 0 indicates that control does not cache content -1 indicates that the control uses the site-level content caching settings <p>Allows you to set up caching of content so that the control doesn't have to retrieve content from the database on each request.</p> <p>The caching mechanism uses absolute expiration time. This means that cache items expire after a specified time period even if the page containing the control wasn't requested.</p> <p>See also: Caching the data of page components, Configuring caching</p>	
FilterControl	Gets or sets the filter control used to limit the data read by the control.	
FilterName	Gets or sets the code name of the filter control used to limit the data read by this control.	
OrderBy	Gets or sets the ORDER BY clause of the SQL query that the control uses to load data.	"NewsReleaseDate DESC"
SelectedColumns	Database table columns that the control loads for pages, separated by commas (,). If null or empty, the control loads all available columns.	
SiteName	Specifies the code name of the Kentico website for which the control loads data.	
StopProcessing	If true, the control stops all processing — does not load or display any data or other HTML output.	
TopN	Specifies the maximum number of database records that the control loads.	
WhereCondition	Gets or sets the WHERE clause of the SQL query that the control uses to loads data.	"Product Price > 100"



Note: The CMSTreeView inherits from the ASP.NET TreeView control, so you can also set any of the [base properties](#).

Appearance and styling

You can adjust the appearance of the CMSTreeView control by setting the inherited standard properties of the [ASP.NET TreeView](#) control.

Additionally, you can modify the design of the CMSTreeView by assigning CSS classes to the following properties:

Property name	Description
InactiveItemClass	CSS class of inactive items.
InactiveItemStyle	Style of inactive items.
SelectedItemClass	CSS class of selected items.
SelectedItemStyle	Style of selected items.

The recommended place to define these classes is in a Kentico [stylesheet](#) using the **CSS stylesheets** application.

You can apply stylesheets to:

- Entire websites
- Individual pages that contain the control