

# Umbraco 4.7.1 Razor + DynamicNode Cheat Sheet



Document Version 1.1

Current Page	Collections	Traversing	Property Checking	
@Model	.Children .[DocTypeNames]s (Pluralized) .Ancestors() .Ancestors(int level) .Ancestors(string nodeTypeAlias) .Ancestors(Func< DynamicNode, bool > func) .AncestorsOrSelf() .AncestorsOrSelf(string nodeTypeAlias) .AncestorsOrSelf(int level) .AncestorsOrSelf(Func< DynamicNode, bool > func) .Descendants() .Descendants(string nodeTypeAlias) .Descendants(int level) .Descendants(Func< INode, bool > func) .DescendantsOrSelf() .DescendantsOrSelf(int level) .DescendantsOrSelf(string nodeTypeAlias) .DescendantsOrSelf(Func< INode, bool > func) .XPath(string xPath) .GetChildrenAsList	.Parent .First() .Last() .Up() .Up(int) (note 0 = 1) .Down() .Down(int) (note 0 = 1) .Next() .Next(int) (note 0 = 1) .Previous() .Previous(int) (note 0 = 1)	.HasProperty(string propertyAlias) .HasValue(string propertyAlias) .IsNull(string propertyAlias)	
Dynamic Node Properties			IsHelpers	
.Parent .Id .Template .SortOrder .Name .Visible (requires umbracoNaviHide property) .Url .UrlName .NodeTypeAlias .WriterName .CreatorName .WriterID .CreatorID .Path .CreateDate .UpdateDate .Version .NiceUrl .Level .PropertiesAsList .ChildrenAsList .Postion()		.IsFirst([valueIfTrue][,valueIfFalse]) .IsNotFirst([valueIfTrue][,valueIfFalse]) .IsLast([valueIfTrue][,valueIfFalse]) .IsNotLast([valueIfTrue][,valueIfFalse]) .IsPostion(int,[valueIfTrue][,valueIfFalse]) .IsNotPosition(int,[valueIfTrue][,valueIfFalse]) .IsModZero([valueIfTrue][,valueIfFalse]) .IsNotModZero([valueIfTrue][,valueIfFalse]) .IsEven([valueIfTrue][,valueIfFalse]) .IsOdd([valueIfTrue][,valueIfFalse]) .IsEqual(DynamicNode[,valueIfTrue][,valueIfFalse]) .IsDecendant(DynamicNode[,valueIfTrue][,valueIfFalse]) .IsDecendantOrSelf(DynamicNode[,valueIfTrue][,valueIfFalse]) .IsAncestor(DynamicNode[,valueIfTrue][,valueIfFalse]) .IsAncestorOrSelf(DynamicNode[,valueIfTrue][,valueIfFalse])		
DynamicMedia Properties		.UmbracoFile .UmbracoSize .UmbracoWidth .UmbrachHeight		Permissions
				.HasAccess() .IsProtected()
Custom Property Access (Content & Media)	<b>Filtering &amp; Ordering &amp; Extensions</b> .Where("Condition"[,valueIfTrue,valueIfFalse]) .OrderBy("propertyAlias [desc][,propertyAlias]") .GroupBy("propertyAlias") .Pluck("propertyName") .Take(int) .Skip(int) .Count()  <b>Functions</b> @functions{ public bool isFooBar(string foo, string bar){ return foo == bar; } } @{ var foo = "black"; if(isFooBar(foo,"black")){ <p>Yup</p> } }	<b>Macro Parameters</b> @Parameter.ParameterName  <b>Media (use for media picker property)</b> .Media("propertyAlias","mediaPropertyAlias") .Media("propertyAlias")  <b>Types for Casting and Newing</b> DynamicNode(int nodeId) DynamicMedia(int mediaId) DynamicNodeList dynamic (allows .PropertyAlias notation)		
Dictionary				
@Dictionary .DictionaryItemAlias				
@Library Helper				
Loaders	Manipulation		<b>Visual Studio DynamicNode Intellisense</b> Put the following at the head of the CSHTML file to get some Intellisense support for DynamicNode and DynamicNodeList @using umbraco.MacroEngines @inherits umbraco.MacroEngines.DynamicNodeContext	
.NodeById(int string) .MediaNodeById(int string)	.Coalesce(value,value[,value ...]) .Concatenate(value,value[,value ...]) .Join(seperator,value,value[,value ...]) .Truncate(htmlString,int[[,bool addElipsis][,bool treatTagsAsContent]]) .StripHTML(htmlString[, tagsToString])			
Conditionals				
.If(booleanProp,valueIfTrue[,valueIfFalse])				