|  |  |  |
| --- | --- | --- |
| **View Base Class**  @inherits KarbonView  General base class for views and partials.  @inherits KarbonView<TCurrentPageType>  Base class for views / partials with a strongly typed current page.  @inherits KarbonView<TCurrentPageType, THomePageType>  Base class for views / partials with a strongly typed current page and home page.  **View Properties**  @Model.CurrentPage  The IContent model for the current page.  @Model.HomePage  The IContent model for the home page.  **Content Properties**  .Name:String  The name of the page parsed from the folder name. Can be overridden in the content file.  .Slug:String  The URL slug for the content.  .TypeName:String  The type name for the content as parsed from the content file name.  .SortOrder:Int  The sort order for the content.  .Depth:Int  The depth of the content.  .RelativePath:String  The path of the content relative to the content store root.  .RelativeUrl:String  The URL of the content relative to the application root.  .Created:DateTimeOffset  The creation date of the content. | .Modified:DateTimeOffset  The last modified date of the content.  .Data:IDctionary<String, String>  The key value dictionary of parsed values from the content file. If a strongly typed model is used, values will be mapped to model properties and the Data collection will then contain any un-mapped values.  **Content Helper Methods**  .Url():StringGet the absolute URL for the content.  .IsVisible():BoolGets a flag indicating whether the content is visible in the navigation.  .IsOpen():BoolGets a flag indicating whether the content is open in the navigation.  .IsHomePage():BoolGets a flag indicating whether the content is the home page.  .IsChildOf(IContent content):BoolGets a flag indicating whether the content is a child of the supplied content.  .IsAncestorOf(IContent content):BoolGets a flag indicating whether the content is an ancestor of the supplied content.  .IsDescendantOf(IContent content):BoolGets a flag indicating whether the content is a descendant of the supplied content.  .HasPrev([Func<IContent, Bool> filter]):Bool  .HasPrev<TContentType>([Func<TContentType, Bool> filter]):BoolGets a flag indicating whether the content has a previous sibling optionally filter by the type or filter function parameter.  .HasNext([Func<IContent, Bool> filter]):Bool  .HasNext<TContentType>([Func<TContentType, Bool> filter]):BoolGets a flag indicating whether the content has a next sibling optionally filter by the type or filter function parameter. | **Content Traversal**  .Parent():IContent  .Parent<TContentType>:TContentType  Gets the parent content optionally cast to the supplied type parameter  .Parents([Func<IContent, Bool> filter]):IEnumersable<IContent>  .Parents<TContentType>([Func<TContentType, Bool> filter]):IEnumerable<TContentType>  Gets the ancestor content optionality filtered by the type and filter function parameter.  .Closest(Func<IContent, Bool> filter):IContent  .Closest<TContentType>([Func<TContentType, Bool> filter]):TContentType  Gets the closest ancestor content item filtered by the type and / or filter function parameter.  .Children([Func<IContent, Bool> filter]):IEnumersable<IContent>  .Children<TContentType>([Func<TContentType, Bool> filter]):IEnumersable<TContentType>  Gets the child content optionally filtered by the type and filter function parameter.  .Siblings([Func<IContent, Bool> filter]):IEnumersable<IContent>  .Siblings<TContentType>([Func<TContentType, Bool> filter]):IEnumersable<TContentType>  Gets the sibling content optionally filtered by the type and filter function parameter.  .Prev([Func<IContent, Bool> filter]): IContent  .Prev<TContentType>([Func<TContentType, Bool> filter]):TContentType  Gets the previous content item optionally filtered by the type and filter function parameter.  .Next([Func<IContent, Bool> filter]): IContent  .Next<TContentType>([Func<TContentType, Bool> filter]):TContentType  Gets the previous content item optionally filtered by the type and filter function parameter.  .Find(Func<IContent, Bool> filter):IEnumersable<IContent>  .Find<TContentType>([Func<TContentType, Bool> filter]):IEnumersable<TContentType>  Gets the descendant content optionally filtered by the type and filter function parameter. |

Page 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Content Media Access**  .Files([Func<IFile, Bool> filter]):IEnumersable<IFile>  .Files<TFileType>([Func<TFileType, Bool> filter]):IEnumersable<TFileType>  Gets all files optionally filtered by the type and filter function parameter.  .Images([Func<IImageFile, Bool> filter]):IEnumersable<IImageFile>  .Images<TFileType>([Func<TFileType, Bool> filter]):IEnumersable<TFileType>  Gets all image files optionally filtered by the type and filter function parameter. Image files must have one of the following file extensions:   |  |  |  | | --- | --- | --- | | - Jpg  - Png  - Tiff | - Jpeg  - Bmp | - Gif  - Tif |   .Videos([Func<IVideoFile, Bool> filter]):IEnumersable<IVideoFile>  .Videos<TFileType>([Func<TFileType, Bool> filter]):IEnumersable<TFileType>  Gets all video files optionally filtered by the type and filter function parameter. Video files must have one of the following file extensions:   |  |  |  | | --- | --- | --- | | - Ogg  - Mp4  - Flv | - Ogv  - Mov  - Swf | - Webm  - Avi |   .Sounds([Func<ISoundFile, Bool> filter]):IEnumersable<ISoundFile>  .Sounds<TFileType>([Func<TFileType, Bool> filter]):IEnumersable<TFileType>  Gets all sound files optionally filtered by the type and filter function parameter. Sound files must have one of the following file extensions:   |  |  |  | | --- | --- | --- | | - Mp3  - Mid  - Rm | - Wav  - Ra | - Wma  - Ram |   .Documents([Func<IDocumentFile, Bool> filter]):IEnumersable<IDocumentFile>  .Documents<TFileType>([Func<TFileType, Bool> filter]):IEnumersable<TFileType>  Gets all sound files optionally filtered by the type and filter function parameter. Document files must have one of the following file extensions:   |  |  |  | | --- | --- | --- | | - Pdf  - Xls  - Pptx | - Doc  - Xlsx  - Rtf | - Docx  - Ppt | | **File Properties**  .Name:String  The name of the file parsed from the file name. Can be overridden in the meta data file.  .Slug:String  The URL slug for the file.  .Extension:String  The file extension of the file.  .Size:Long  The size of the file in bytes.  .TypeName:String  The type name for the meta data as parsed from the meta data file name.  .SortOrder:Int  The sort order for the file.  .RelativePath:String  The path of the file relative to the content store root.  .RelativeUrl:String  The URL of the file relative to the application root.  .ContentRelativeUrl:String  The relative URL of the associated content.  .Created:DateTimeOffset  The creation date of the file.  .Modified:DateTimeOffset  The last modified date of the file.  .Data:IDctionary<String, String>  The key value dictionary of parsed values from the meta data file. If a strongly typed model is used, values will be mapped to model properties and the Data collection will then contain any un-mapped values.  **File Helper Methods**  .Url():StringGets the absolute URL for the content.  .NiceSize():StringGets the size of the file formatted in a human readable format. | .MimeType():StringGets the mime type of the file.  .IsImage():BoolGets a flag indicating whether the file is an image.    .IsVideo():BoolGets a flag indicating whether the file is a video.  .IsSound():BoolGets a flag indicating whether the file is a sound.  .IsDocument():BoolGets a flag indicating whether the file is a document.  .HasPrev([Func<IFile, Bool> filter]):Bool  .HasPrev<TFileType>([Func<TFileType, Bool> filter]):BoolGets a flag indicating whether the file has a previous sibling optionally filter by the type or filter function parameter.  .HasNext([Func<IFile, Bool> filter]):Bool  .HasNext<TFileType>([Func<TFileType, Bool> filter]):BoolGets a flag indicating whether the file has a next sibling optionally filter by the type or filter function parameter.  .Prev([Func<IFile, Bool> filter]): IFile  .Prev<TFileType>([Func<TFileType, Bool> filter]):TFileType  Gets the previous file optionally filtered by the type and filter function parameter.  .Next([Func<IFile, Bool> filter]): IFile  .Next<TFileType>([Func<TFileType, Bool> filter]):TFileType  Gets the previous file optionally filtered by the type and filter function parameter.  **Image File Properties**  In addition to the standard file properties, image files have the following additional properties:  .Width:Int  The width of the image.  .Height:Int  The height of the image. |

Page 2

|  |  |  |
| --- | --- | --- |
| **Image File Helper Methods**  The following “Fit” methods all make use of the ImageResizing.net library.  .FitWidth(int width):String  Gets the URL of the image resized to fit within the supplied with parameter.  .FitHeight(int height):String  Gets the URL of the image resized to fit within the supplied height parameter.  .Fit(int maxWidthHeight, [FitMode fitMode, ScaleMode scaleMode, AlignMode alignMode, ImageFormat format, int quality, int colors, string bgColor]):String  .Fit(int width, int height, [FitMode fitMode, ScaleMode scaleMode, AlignMode alignMode, ImageFormat format, int quality, int colors, string bgColor]):String  Gets the URL of the image resized to fit within the supplied width / height parameter, optionally constrained by the supplied parameters.  **Controller Base Class**  KarbonController  General base class for custom controllers.  KarbonController<TCurrentPageType>  Base class for custom controllers with a strongly typed current page.  KarbonController<TCurrentPageType, THomePageType>  Base class for controllers with a strongly typed current page and home page.  **Controller Properties**  .CurrentPage  The IContent model for the current page.  .HomePage  The IContent model for the home page.  **Cheat Sheet v1,0**  Please go to <http://karboncms.com/docs> to get the full documentation. |  |  |

Page 3