

ASP.NET MVC 4 Cheat Sheet



Application Folders

App_Start

It has configuration classes to reduce clutter code in the Global.asax Application_Start method. It contains following files:

- AuthConfig.cs

Related to security settings, including sites for OAuth login.

- BundleConfig.cs

Related to register bundles for the bundling and minification

- FilterConfig.cs

To register global MVC filters.(by default HandleErrorAttribute is registered)

- RouteConfig.cs

Related to MVC Route configuration

- WebApiConfig.cs

Used to register Web API routes and configuration settings

Areas

An MVC structure inside the application to partition our web application into smaller parts.

Content

To put CSS and other static contents

Controllers

Contains the controller classes

Images

For images, It is being referenced from the Content\Site.css by default.

Models

Contains view model and domain model classes.

Scripts

Stores all JavaScript libraries and files

Views

Related to the display of the application (UI).

Html Helpers

used to render (modify and output) HTML form elements

Html.ActionLink

Output: ...

Html.BeginForm

Output: <form ...>

Html.CheckBox

Output: <input type="checkbox" .../>

Html.DropDownList

Output: <select ...> ... </select>

Html.EndForm

Output: </form>

Html.Hidden

Output: <input type="hidden" ... />

Html.ListBox

Output: <select multiple ...>...</select>

Html.Password

Output: <input type="password" ... />

Html.RadioButton

Output: <input type="radio" ... />

Html.TextArea

Output: <textarea>...</textarea>

Html.TextBox

Output: <input type="text" ... />

Html.ValidationSummary

Returns an unordered list (ul element) of validation messages that are in the ModelStateDictionary object.

Html.ValidationMessage

To display an error for a particular field in the ModelState dictionary

Html.RouteLink

To generate a URL to a specific route

View Related & Others

@{ Layout = "..."}

To define layout page Equivalent to asp.net master-page

@model <StrongModelType>

To define strongly-typed model

@section <name> { }

To define named section

@RenderBody

Used in layout as a placeholder for view's entire content.

@RenderPage

Renders the content of one page within another page.

@RenderSection

In layout pages, renders the content of a named section.

IsSectionDefined

Returns a value that indicates whether the specified section is defined in the page.

DefineSection

To create named content sections in content page.

@helper

To define a helper which create a reusable code snippet.

@functions

To define a function (pure server-side methods)that can be reused within the view.

@ViewBag.<myData>

Dynamic property, used to pass data between a Controller and a View

@ViewData[myData]

To pass data between a Controller and a View via special Dictionary class

@tempdata

To pass State Between Action Methods

Action Method Results

ActionResult

generic return value for an action used to perform a framework-operation on behalf of the action

ContentResult

To return a user-defined content is the result of an action method

EmptyResult

To return nothing from a Controller

FileContentResult

Sends the contents of a binary response.

FilePathResult

Sends the contents of a file to response.

FileResult

To return the content of a file to browser

FileStreamResult

Sends binary content to the response using a Stream instance.

HttpNotFoundResult

To return 404 status-code to indicate the requested resource was not found

HttpStatusCodeResult

To return specific HTTP response code and description.

HttpUnauthorizedResult

To return 401 status code to indicate unauthorized HTTP request.

JsonResult

To return a serialized JSON object

JavaScriptResult

To return a script that can be executed by the client.

PartialViewResult

To render a partial view (fx. just a control)

RedirectResult

To redirect to a specified URL.

RedirectToRouteResult

to redirect to a route

ValueProviderResult

Represents the result of binding (such as from a form post or query string) to an action-method argument or to the argument itself.

ViewEngineResult

Represents the result of locating a view.

ViewResult

To Render a view as a Web page

Attributes

ActionFilterAttribute

Represents the base class for filter attributes.

ActionMethodSelectorAttribute

Represents an attribute that is used to influence the selection of an action method.

ActionNameSelectorAttribute

Represents an attribute that affects the selection of an action method.

CustomModelBinderAttribute

It invokes a custom model binder.

FilterAttribute

base class for action and result filter attributes.

Attributes

[AcceptVerbs(...)]

To specify HTTP verbs an action method will respond to.

[ActionName(...)]

To define the name of an action

[AdditionalMetadata(...)]

to populate the ModelMetadata.AdditionalValues dictionary for a model property.

[AllowHtml]

To allow HTML markup during model binding by skipping request validation for the property.

[AsyncTimeout(...)]

To set the timeout value, in milliseconds, for an asynchronous method.

[Authorize(...)]

To restrict access by callers to an action method.

[ChildActionOnly]

Data Annotations for Model

[Bind(...)]

Lists fields to exclude or include when binding parameter or form values to model properties

[Compare("...")]

To compare two properties.

[CreditCard]

To Specify that a data field value is a credit card number.

[DataType(DataType.<type>)]

used for formatting purposes, not for validation.

[DisplayName (...)]

Defines the text we want used on form fields and validation messages

[EmailAddress]

To validate an email address

Html Helpers

@Html.AntiForgeryToken

It generates a hidden form field (anti-forgery token) that is validated when the form is submitted.

To indicate that an action method should be called only as a child action.

[HandleError(...)]

to handle an exception that is thrown by an action method.

[HiddenInput(...)]

To render a property or field value as a hidden input element.

[HttpDelete]

to restrict an action method to handle only HTTP DELETE requests.

[HttpGet]

to restrict an action method to handle only HTTP Get requests.

[HttpPost]

to restrict an action method to handle only HTTP Post requests.

[HttpPut]

to restrict an action method to handle only HTTP Put requests.

[ModelBinder(...)]

to associate a model type to a model-builder type.

[NoAsyncTimeout]

To set the timeout to infinite (never).

[NonAction]

To indicate that a controller method is not an action method.

[OutputCache(...)]

To mark an action method whose output will be cached.

[Remote(...)]

It is used to invoke the server-based validation from client side.

[RequireHttps]

It forces an unsecured HTTP request to be re-sent over HTTPS.

[SessionState(...)]

Specifies the session state of the controller.

[ValidateInput(...)]

To mark action methods whose input must be validated.

[FileExtensions(Extensions = "jpg,png...")]

To Validate file name extensions.

[key]

To denote one or more properties that uniquely identify an entity.

[MaxLength(...)]

To define the maximum length of array or string data allowed in a property.

[MinLength(...)]

To define the minimum length of array or string data allowed in a property.

[Phone..]

To Specify for phone number using a regular expression.

[Range (...)]

Gives a maximum and minimum value for a numeric field

[RegularExpression(...)]

To specify that a data field value must match the specified regular expression.

[Required (...)]

Indicates that the property is a required field

[StringLength(...)]

Defines a maximum length for a string field

[ScaffoldColumn(...)]

Allows hiding fields from editor forms

[Url]

It provides URL validation.

@Html.AttributeEncode

To convert the specified attribute to an HTML-encoded string.

@Html.Encode

To convert the specified value to an HTML-encoded string.

@{ Html.EnableClientValidation }

To enables or disables client validation.

Html.EnableUnobtrusiveJavaScript

To enables or disables unobtrusive JavaScript.

@Html.FormatValue

To format value

@Html.Raw

To return markup that is not HTML-encoded.

@Html.Partial

To render a partial view into a page.

@{Html.RenderPartial(..)}

writes directly to the response stream instead of returning a string.

@Html.Action

executes a separate controller action and returns as string

Html.RenderAction

executes a separate controller action and render the result directly to the response stream.

[Contact Us](#) | [Disclaimer](#)

Copyright © 2014 TechBrij. Content cannot be reproduced without permission from the author.