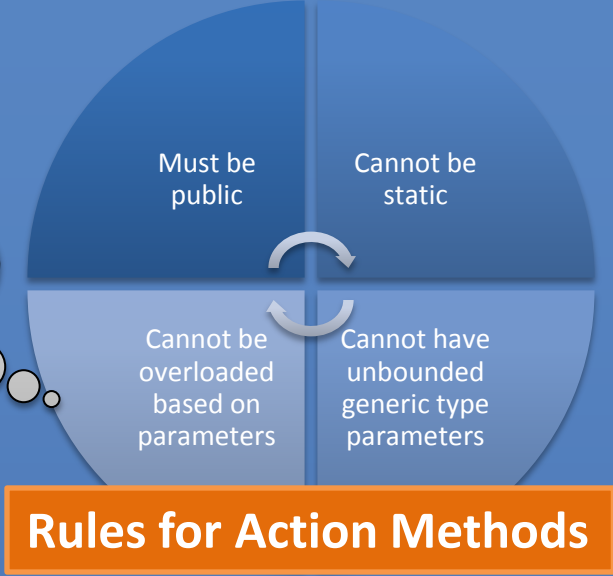


Restricts callers of the action	[Authorize]	Action Filters
Handling of an exception thrown by an action	[HandleError]	
Indicates that the output will be cached	[OutputCache]	
Used to detect if someone is tampering with the request	[ValidateAntiForgeryToken]	
Used to validate input	[ValidateInput]	
Specifies which HTTP verbs to respond to	[AcceptVerbs]	Action Selectors
The name of the action	[ActionName]	
Indicates that it is not an action	[NonAction]	
Provide details of how the model binding should work	[Bind]	Binding
Used to associate a model type to a model builder type	[ModelBinder]	

Action Method Attributes

If you add a action name selector attribute, you can have overloaded methods with the same parameters.



ASP.net | MVC

Controller Cheat Sheet

Example:

- Return view for that action method: `return View();`
- Return a different view: `return View("ViewName");`

Example:

- Return a string: `return Content("A plain string");`
- Return XML: `return Content("<node/>", "text/xml");`

Example:

- Return an object as JSON: `return Json(player);`

Action Result Return Types

Action Result	Helper Method	Description
ViewResult	View	Renders a view as a Web page.
PartialViewResult	PartialView	Renders a partial view, which defines a section of a view that can be rendered inside another view.
RedirectResult	Redirect	Redirects to another action method by using its URL.
RedirectToRouteResult	RedirectToAction or RedirectToRoute	Redirects to another action method.
ContentResult	Content	Returns a user-defined content type.
JsonResult	Json	Returns a serialized JSON object.
JavaScriptResult	JavaScript	Returns a script that can be executed on the client.
FileResult	File	Returns binary output to write to the response.
EmptyResult	(None)	Represents a return value that is used if the action method must return a null result (void).

Need to access HTTP items in the controller?

The **Request** and **Response** objects of the Controller class accept objects that implement the `HttpRequestBase` and `HttpResponseBase` abstract classes. This makes it easy to create mock objects, which in turn makes it easy to create unit tests for controller classes.

TempData
Need to pass data from one call to another, Use the TempData property!

More info: <http://msdn.microsoft.com/en-us/library/dd410269.aspx>