ActionVerbs: HttpGet, HttpPost, HttpPut

The `ActionVerbs` selector is to handle different type of Http requests. The MVC framework includes HttpGet, HttpPost, HttpPut, HttpDelete, HttpOptions, and HttpPatch action verbs. You can apply one or more action verbs to an action method to handle different HTTP requests. If you don't apply any action verbs to an action method, then it will handle HttpGet request by default.

The following table lists the usage of HTTP methods:

Http method	Usage
GET	To retrieve the information from the server. Parameters will be appended in the query string.
POST	To create a new resource.
PUT	To update an existing resource.
HEAD	Identical to GET except that server do not return the message body.
OPTIONS	It represents a request for information about the communication options supported by the web server.
DELETE	To delete an existing resource.
PATCH	To full or partial update the resource.

Visit W3.org for more information on Http Methods.

The following example shows how to handle different types of HTTP requests in the `Controller` using ActionVerbs:

Example: Handle HTTP Requests in the Controller

Copy

```
public class StudentController : Controller
{
    public ActionResult Index() // handles GET requests by default
    {
        return View();
    }

    [HttpPost]
    public ActionResult PostAction() // handles POST requests by default
    {
        return View("Index");
    }


    [HttpPut]
    public ActionResult PutAction() // handles PUT requests by default
    {
        return View("Index");
    }

    [HttpDelete]
    public ActionResult DeleteAction() // handles DELETE requests by default
    {
        return View("Index");
    }

    [HttpHead]
    public ActionResult HeadAction() // handles HEAD requests by default
    {
        return View("Index");
    }

    [HttpOptions]
    public ActionResult OptionsAction() // handles OPTION requests by default
    {
        return View("Index");
    }

    [HttpPatch]
    public ActionResult PatchAction() // handles PATCH requests by default
    {
        return View("Index");
    }
}
```

You can also apply multiple action verbs using the `AcceptVerbs` attribute, as shown below.

Example: AcceptVerbs

Copy

```
[AcceptVerbs(HttpVerbs.Post | HttpVerbs.Get)]
public ActionResult GetAndPostAction()
{
    return RedirectToAction("Index");
}
```