link
@Html.ActionLink("View", "details", new { id = employee.Id })

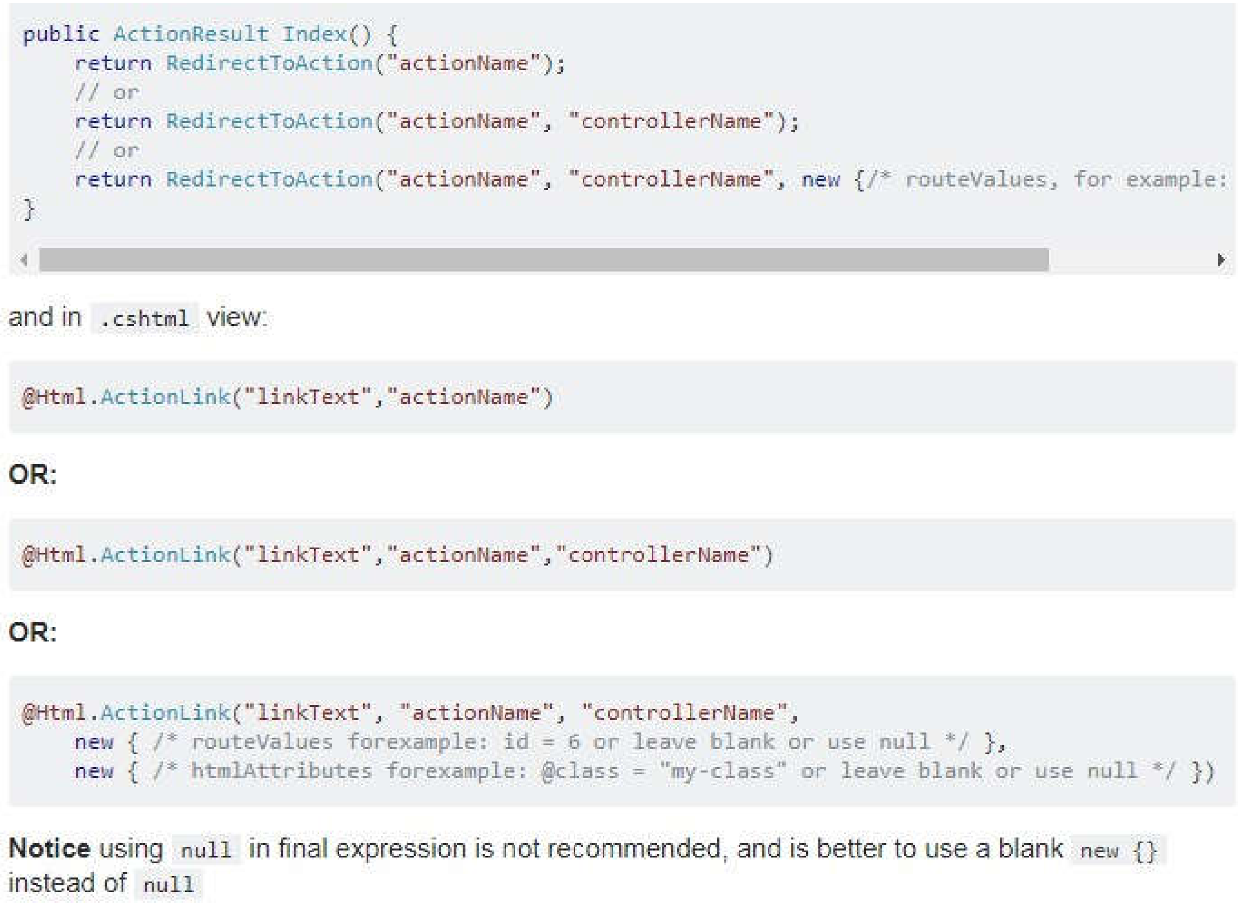
```
@html.BeginForm()
```

It's not a stupid question. `@html.BeginForm()` works like this. It has some parameters you could add to it like `Action` `Controller` `FormType` `htmlAttributes`. The way it works is that if you leave it empty it will look for a post action with the same name that on the page you are now, for example if you are in on the login page, it will look for a login post action. I always write what `action` and `controller` I want it to access.

```
@Html.BeginForm("Action", "Controller", FormMethod.Post, new { @class = "my_form"}) {

}
```

by using `@using (Html.BeginForm("Action", "Controller", FormMethod.Post))`
. you don't need to care about closing your form tag and it prevents accidental issue if somebody forgets to close the form.



```
public ActionResult Index() {
    return RedirectToAction("actionName");
    // or
    return RedirectToAction("actionName", "controllerName");
    // or
    return RedirectToAction("actionName", "controllerName", new {/* routeValues, for example:
}
```

and in `.cshtml` view:

```
@Html.ActionLink("linkText","actionName")
```

OR:

```
@Html.ActionLink("linkText","actionName","controllerName")
```

OR:

```
@Html.ActionLink("linkText", "actionName", "controllerName",
    new { /* routeValues forexample: id = 6 or leave blank or use null */ },
    new { /* htmlAttributes forexample: @class = "my-class" or leave blank or use null */ })
```

Notice using `null` in final expression is not recommended, and is better to use a blank `new {}` instead of `null`