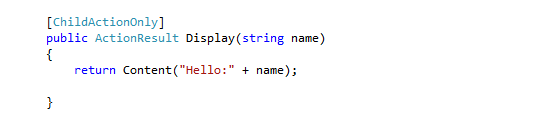
**Child Action Only**

1. A controller is a collection of action methods.
2. Action methods are just like methods and have a public access specifier.
3. Since they are public we can easily call them from a URL request directly from the browser.
4. In our projects, when we want an action method to not be called from a URL request we should make the method a child action method.
5. An action method can be a child or a normal action method, but child actions are action methods invoked from within a view, you cannot invoke a child action method via user request (URL).
6. We can annotate an action method with the [ChildActionOnly] attribute to create a child action. Normally we use child action methods with partial views.
7. Thus [ChildActionOnly] is an attribute when it is preceded over any controller action method. After this, the method could not make a request from the browser, it can only be called from a View.
8. We can call a child action method by using the following two HTML controls.

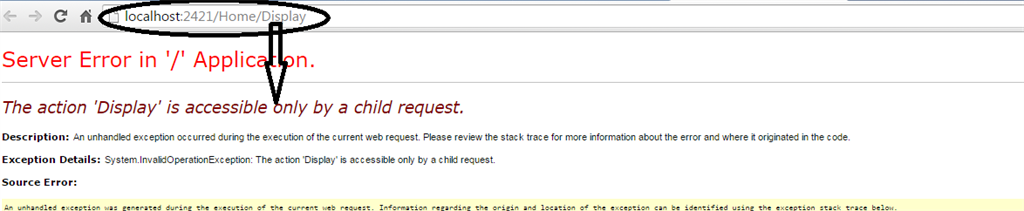
**@html.Action()**

**@html.RenderAction()**

Here in the first example, I will explain how to call a child action method using @html.action. I have a home controller and display child action method.

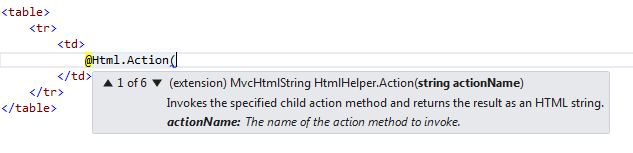
****

Now, accessing this Child action method from the browser request will throw the following error.

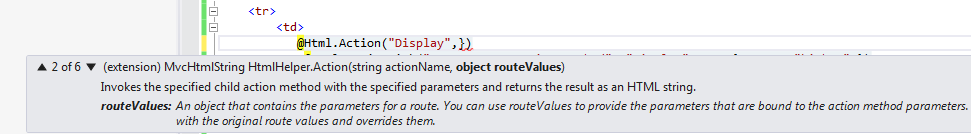
   
  
Thus from this we conclude that a child action method cannot be called directly from a browser request.

Now the question is, how and from where to call a child action method. We can call a child action method from the view.

So, we need to create a view for consuming this child action method as in the following:

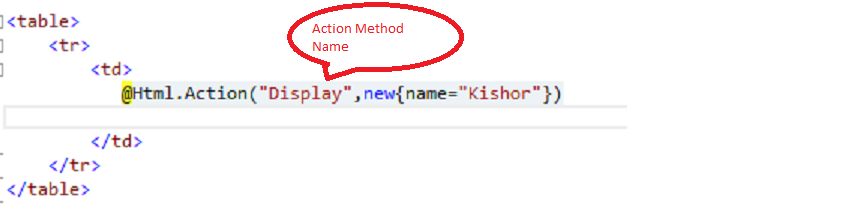


When defining the @Html.Action() method the first parameter should be your child action name.

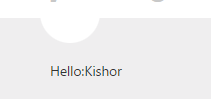


As in the preceding code, the child action method is “Display”.

Now for my second parameter, I will pass a name so that it will be consumed and displayed using the child action method.

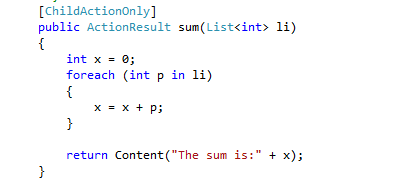


Now when I execute my project it will provide the following output in my view.

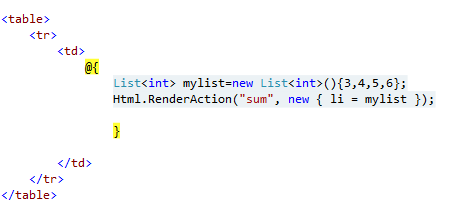


Now I will explain another example, but now accessing using the @html.RenderAction().

Here is the sum child action method with the same controller.



And this is my view using @html.RenderAction().



And my output is:

