HTTP modules and HTTP handlers in ASP.Net MVC?

Asked 3 years, 2 months ago   Active 1 year, 8 months ago   Viewed 20k times

▲

18

▼

★

6

I was asked a question in interview that how to implement `HTTP module` and `HTTP handler` in `ASP.Net MVC`. I know that they are used in ASP.Net to write pre-processing logic before the aspx page is called. But in ASP.Net MVC we have filters for that so i told them we use Filters for that. Did i gave the right answer?

asp.net   asp.net-mvc   httphandler   httpmodule

edited Nov 4 '16 at 14:10   asked Nov 4 '16 at 13:57

rohit singh
857   2   8   18

Filters are only invoked for requests handled by controllers. To be more general you would have to use IHttpHandler and IHttpModule... unless you're in OWIN in which case it would be an OWIN Middleware class. – Dai Nov 4 '16 at 14:13
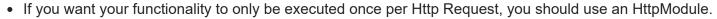
I didn't get you here. I'm not sure about OWIN – rohit singh Nov 4 '16 at 14:23

1 Answer

▲

24

▼

Action Filters allow you to hook into *MVC specific events only*, whereas **HTTP Modules** allow you to hook into *ASP.Net events*. So even in MVC, to implement a HTTP Module and HTTP handler, you will need to implement corresponding interface.

- If you want your functionality to only be executed once per Http Request, you should use an HttpModule.
- ActionFilters may be executed several times in a single trip to the server.

✓ To explain HTTP Modules and HTTP Handlers, HTTP module and HTTP handler are used by MVC to inject pre-processing logic in the request chain.

- HTTP Handlers are extension based pre-processor whereas HTTP Module are event based preprocessor.
  - *For example:* if you want to change how `jpg` files are processed, you will implement custom HTTP handler versus if you want to execute additional logic during processing of the request, you will implement a custom HTTP module. There is always only one HTTP handler for a specific request but there can be multiple HTTP modules.

To Implement an HTTP Handler:

You implement `IHttpHandler` class and implement `ProcessRequest` method and `IsResuable` property. `IsResuable` property determines whether handler can be reused or not.

```
public class MyJpgHandler: IHttpHandler
{

    public bool IsReusable => false;

    public void ProcessRequest(HttpContext context)
    {
        // Do something
    }
}
```

Next we need to specify which kind of request will be handled by our custom handler in `web.config` file:

```
<httpHandlers>
    <add verb="*" path="*.jpg" type="MyJpgHandler"/>
</httpHandlers>
```

To implement an HTTP module:

We need to implement `IHttpModule` and register the required events in `Init` method. As a simple example, if we wanted to log all requests:

```
public class MyHttpModule: IHttpModule
{

```

```csharp
    {
        application.BeginRequest += new EventHandler(this.context_BeginRequest);
        application.EndRequest += new EventHandler(this.context_EndRequest);
    }

    public void context_BeginRequest(object sender, EventArgs e)
    {
        StreamWriter sw = new StreamWriter(@ "C:\log.txt", true);
        sw.WriteLine("Request began at " + DateTime.Now.ToString());
        sw.Close();
    }

    public void context_EndRequest(object sender, EventArgs e)
    {
        StreamWriter sw = new StreamWriter(@ "C:\log.txt", true);
        sw.WriteLine("Request Ended at " + DateTime.Now.ToString());
        sw.Close();
    }

    public void Dispose() {}
}
```

And register our module in `web.config` file:

```xml
<httpModules>
    <add name="MyHttpModule " type="MyHttpModule " />
</httpModules>
```

edited Apr 25 '18 at 1:50　　　　　answered Jan 13 '17 at 14:53

a-man　　　　　　　　　　　　　　　Ujain
489　4　14　　　　　　　　　　　**366**　3　4