Web API Exception Handling

<https://docs.microsoft.com/en-us/aspnet/web-api/overview/error-handling/exception-handling>

Other links

Global - <https://docs.microsoft.com/en-us/aspnet/web-api/overview/error-handling/web-api-global-error-handling>

M Jones <https://www.exceptionnotfound.net/the-asp-net-web-api-exception-handling-pipeline-a-guided-tour/>

Web API - <https://docs.microsoft.com/en-us/aspnet/web-api/overview/security/>

<https://docs.microsoft.com/en-us/aspnet/web-api/overview/error-handling/exception-handling>

Stack overflow - <https://stackoverflow.com/questions/38025305/best-practice-for-error-handling-with-asp-net-web-api>

**HttpResponseException:**

if (item == null)

{

var resp = new HttpResponseMessage(HttpStatusCode.NotFound)

{

Content = new StringContent(string.Format("No product with ID = {0}", id)),

ReasonPhrase = "Product ID Not Found"

};

throw new HttpResponseException(resp);

}

**Exception Filters**

You can customize how Web API handles exceptions by writing an *exception filter*. An exception filter is executed when a controller method throws any unhandled exception that is *not* an **HttpResponseException** exception. The **HttpResponseException** type is a special case, because it is designed specifically for returning an HTTP response.

Exception filters implement the **System.Web.Http.Filters.IExceptionFilter** interface. The simplest way to write an exception filter is to derive from the **System.Web.Http.Filters.ExceptionFilterAttribute** class and override the **OnException** method.

using System.Web.Http.Filters;

public class NotImplExceptionFilterAttribute : ExceptionFilterAttribute

{

public override void OnException(HttpActionExecutedContext context)

{

if (context.Exception is NotImplementedException)

{

context.Response = new HttpResponseMessage(HttpStatusCode.NotImplemented);

}

}

}

The **Response** property of the **HttpActionExecutedContext** object contains the HTTP response message that will be sent to the client.

[NotImplExceptionFilter]

To apply the filter globally to all Web API controllers, add an instance of the filter to the **GlobalConfiguration.Configuration.Filters** collection. Exception filters in this collection apply to any Web API controller action.

C#Copy

GlobalConfiguration.Configuration.Filters.Add(

new ProductStore.NotImplExceptionFilterAttribute());