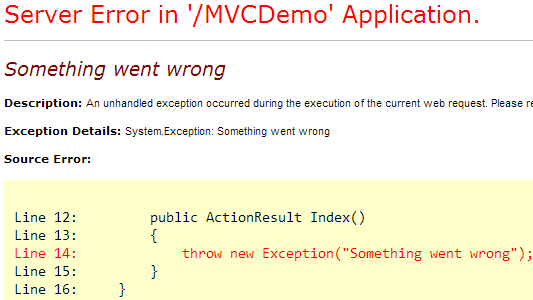
In this video, we will discuss **HandleError**attribute in asp.net mvc. HandleErrorAttribute is used to display friendly error pages to end user when there is an unhandled exception. Let us understand this with an example.   
  
   
  
**Step 1:** Create a blank asp.net mvc 4 application.  
  
**Step 2:** Add a HomeController. Copy and paste the following code.   
public ActionResult Index()  
{  
    throw new Exception("Something went wrong");  
}  
  
   
  
Notice that, the **Index**() action method throws an exception. As this exception is not handled, when you run the application, you will get the default **"yellow screen of death"** which does not make sense to the end user.  
   
  
Now, let us understand replacing this yellow screen of death, with a friendly error page.  
  
**Step 3:** Enable custom errors in web.config file, that is present in the root directory of your mvc application. "customErrors" element must be nested under "<system.web>". For detailed explanation on MODE attribute, please watch [Part 71](http://csharp-video-tutorials.blogspot.com/2012/12/custom-errors-in-aspnet-part-71.html) of [ASP.NET Tutorial](http://www.youtube.com/user/kudvenkat/videos?view=1&flow=grid).  
<customErrors mode="On">  
</customErrors>  
  
**Step 4:** Add **"Shared"** folder under **"Views"** folder. Add **Error.cshtml** view inside this folder. Paste the following HTML in Error.cdhtml view.  
<h2>An unknown problem has occured, please contact Admin</h2>

Run the application, and notice that, you are redirected to the friendly **"Error"** view, instead of the generic **"Yellow screen of death"**.  
  
**We did not apply HandleError attribute anywhere. So how did all this work?**  
**HandleErrorAttribute**is added to the GlobalFilters collection in global.asax. When a filter is added to the GlobalFilters collection, then it is applicable for all controllers and their action methods in the entire application.   
  
Right click on **"RegisterGlobalFilters()"** method in Global.asax, and select **"Go To Definition"** and you can find the code that adds **"HandleErrorAttribute"** to GlobalFilterCollection.  
public static void RegisterGlobalFilters(GlobalFilterCollection filters)  
{  
    filters.Add(new HandleErrorAttribute());  
}  
  
**Is the friendly error page displayed for HTTP status code 404?**  
No, but there is a way to display the friendly error page.  
  
In the HomeController, we do not have List() action method. So, if a user navigates to **/Home/List**, we get an error - The resource cannot be found. HTTP 404.  
  
**To display a friendly error page in this case**  
**Step 1:** Add **"ErrorController"** to controllers folder. Copy and paste the following code.  
public class ErrorController : Controller  
{  
    public ActionResult NotFound()  
    {  
        return View();  
    }  
}  
  
**Step 2:** Right click on **"Shared"** folder and add **"NotFound.cshtml"** view. Copy and paste the following code.  
<h2>Please check the URL. The page you are looking for cannot be found</h2>  
  
**Step 3:** Change **"customErrors"** element in web.config as shown below.  
<customErrors mode="On">  
  <error statusCode="404" redirect="~/Error/NotFound"/>  
</customErrors>