

Re: filtering DataGrid on the fly

Source:

<http://www.tech-archive.net/Archive/Internet-Server/microsoft.public.inetserver.asp.db/2006-09/msg00136.html>

- *From:* stcheng@xxxxxxxxxxxxxxxxxxxxxx (Steven Cheng[MSFT])
 - *Date:* Thu, 28 Sep 2006 05:02:40 GMT
-

Hello Michael,

From your description, I understand you're developing an ASP.NET web page

which will display some data from database and you want to also add the functionality to allow user input some text on the page to filter the data displayed on the page, correct?

As for this question, I think we need to start from the ASP.NET web page programming's underlying model which is quite different from winform application.

ASP.NET web application is page based and each time the user visit a page, it will got through the following steps:

**user request the page in browser address bar(or invoke a certain link or button on former page)

**browser send request to the server

**server receive the request and use the proper handler to process the request

**ASP.NET runtime load the certain page class and go through the page's server-side lifecycle include page initialization, page viewstate processing, page event processing, page rendering

Here are two good msdn reference introducing the ASP.NET runtime lifecycle:

#ASP.NET Application Life Cycle Overview
<http://msdn2.microsoft.com/en-us/library/ms178473.aspx>

#ASP.NET Page Life Cycle Overview
<http://msdn2.microsoft.com/en-us/library/ms178472.aspx>

** after the page is processed at server-side, the ASP.NET runtime flush

Re: filtering DataGrid on the fly

the response content back to client browser and the client browser display the page as html.

Therefore, you can see that ASP.NET page based application is much different from winform. ASP.NET application is request/response based and the server-side execution/processing only take a little period of time in the whole lifecycle. While for winform, during the application's whole lifetime, the application is executing in memory and the application data and variables are accessible.

Then, for your scenario, you want to display data in ASP.NET page(I assume you're using ASP.NET 2.0), you need to use ASP.NET DataBound control on the page and use ASP.NET databinding to bind data (from database) to the databound control. And in ASP.NET 2.0, the GridView is a good one to use, it support bind data to DataTable/Dataview or our custom class objects(list). And if you want to filter data, you need to regenerate the filtered datasource(DataView) and rebind it to GridView. And if you want to let the user to input the keyword for filtering, you can use a TextBox (or dropdownlist) to display the filter options, and when the user input some data and click a submit button, you'll regenerate the filtered data according to the new input value (in textbox or dropdownlist) and bind the new data to GridView. We can not do it on the fly like winform application because ASP.NET application need to postback the page to server first and then process the data in code.

Also, I suggest you have a look at the samples and tutorials in the WWW.ASP.NET site since there're many good learning resource on ASP.NET there. The following ones contains some example about data accessing in asp.net and data filtering:

#ASP.NET Quickstart Tutorials—Performing Data Access
<http://quickstarts.asp.net/QuickStartv20/aspnet/doc/data/default.aspx>

#Working with Data in ASP.NET 2.0
<http://www.asp.net/learn/dataaccess/default.aspx?tabid=63>

Also, the above info are some general guideline, if you have any detailed questions or meet any specific problem when implementing the page, please feel free to post here.

Sincerely,

Re: filtering DataGrid on the fly

Re: filtering DataGrid on the fly

Steven Cheng

Microsoft MSDN Online Support Lead

=====

Get notification to my posts through email? Please refer to <http://msdn.microsoft.com/subscriptions/managednewsgroups/default.aspx#notifications>.

Note: The MSDN Managed Newsgroup support offering is for non-urgent issues where an initial response from the community or a Microsoft Support Engineer within 1 business day is acceptable. Please note that each follow up response may take approximately 2 business days as the support professional working with you may need further investigation to reach the most efficient resolution. The offering is not appropriate for situations that require urgent, real-time or phone-based interactions or complex project analysis and dump analysis issues. Issues of this nature are best handled working with a dedicated Microsoft Support Engineer by contacting Microsoft Customer Support Services (CSS) at <http://msdn.microsoft.com/subscriptions/support/default.aspx>.

=====

This posting is provided "AS IS" with no warranties, and confers no rights.

.