Mitch Meyer

Assignment 9.3

ListView and Pager Controls

ListView is a control function available to developers in ASP.Net that allows the developer to bind data that is returned from a database and display that data. ListView gives a developer freedom to display this data individually or in groups. DataPager controls are used to enable paging on ListView objects but is defined as a separate object. These two relatively new features of ASP.Net show what the future may hold in store for future iterations of ASP.Net.

ListView is very useful for displaying data in a repeating structure, similar to a DataList or GridView function. What sets ListView apart from DataList and GridView is that command buttons can be enabled in ListView without code and it provides for automatic sorting, paging, and formatting. This makes ListView a very powerful choice to displayed bound data. There are two ways to bind a data source to a ListView control. The first way is by using the DataSourceID property. Binding a data source to a ListView control using this method enables ListView to take advantage of the Data Source control and enables it to take advantage of built-in sorting, paging, inserting, deleting, and updating controls. The second method for binding a data source to a ListView control is by using the DataSource property. This method allows a developer to bind the ListView to various objects including ADO.Net datasets, data readers, and in-memory structures but additional code must be written to enable the controls enabled by the DataSourceID method.

Templates can be used to define how data is displayed by a ListView control. This data can be displayed individually or in groups. ListView comes with a set of prebuilt templates including LayoutTemplate, ItemTemplate, ItemSeparatorTemplate, GroupTemplate, GroupSeparatorTemplate, EmptyItemTemplate, EmptyDataTemplate, SelectedItemTemplate, AlternatingItemTemplate, EditItemTemplate, and InsertItemTemplate. The LayoutTemplate is the root template that defines the main layout of the control. This usually contains a placeholder object that will be replaced with content that is defined in the ItemTemplate at run time. The LayoutTemplate may also contain a DataPager Object. The ItemTemplate is used to define the data-bound content to be displayed for a single item. The ItemSeparatorTemplate is used to define content to be displayed between individual items. GroupTemplate is used to define how data will be displayed in a group layout and the GroupSeparatorTemplate defines the separation between group items. EmptyItemTemplate is used in conjunction with a GroupTemplate to define what would be displayed for a situation where part of the group display would be empty, such in a situation where there is an odd number of data items to be displayed. EmptyDataTemplate defines what is to be rendered if the data source returns no data. The SelectedItemTemplate defines how the ListView would render data that has been selected to differentiate that data item from other displayed items. AlternatingItemTemplate is used to help a user differentiate between a list of consecutive items. The EditItemTemplate defines how items will be rendered when the data for an item is being edited. InsertItemTemplate identifies the content that will be displayed when an item is being inserted into the data source. These templates allow the developer to deploy their data with a large amount of functionality quickly. Using these templates, the code-behind file only needs to provide data validation and error handling for the database operations.

Paging data while using the ListView control requires the use of a DataPager control. The DataPager control is a control object that can be used within a Listview control or can be used outside of the ListView by defining the PagedControlID with the ID of the ListView control that is being paged. The DataPager Control has attributes that allow the developer to define what kind of paging controls will be rendered. It allows for button controls, numbered controls, and a combination of both. The developer can determine how many numbered pages will be displayed as well. The default setting of the DataPager control is to display a maximum of 5 paging buttons but this setting is easily adjusted to suit the needs of the application. DataPager controls can only be used with controls that implement the IPageableItemContainer interface, and currently, only the ListView control fits that requirement. This limits how useful the DataPager control is, but future versions of ASP.Net will most likely grant other controls the ability to use the DataPager control.

ListView is an incredibly powerful tool available to ASP.Net developers for displaying bound data items. It really streamlines the process of formatting and displaying the information and allows the developer to focus their coding time on data validation and handling exceptions. The DataPager control makes it very simple for a developer to implement an integral feature, especially when dealing with large data sets, with relative ease. The only major downside to the ListView and DataPager controls is that the DataPager is only useable with ListView at this time.

Sources

Delamater, M., & Boehm, A. (2013). *Murach's ASP.NET 4.5 Web programming with C# 2012* (5th ed.). Fresno, CA: Mike Murach & Associates.

ListView Web Server Control Overview. (n.d.). Retrieved October 12, 2015, from https://msdn.microsoft.com/en-us/library/bb398790.aspx