ASP.NET - Data Sources Controls

ASP.NET - Data Bound Controls

ASP.NET - Data Sources Controls

- A data source control interacts with the data-bound controls and hides the complex data binding processes.
- These are the tools that provide data to the data bound controls and support execution of operations like insertions, deletions, sorting and updates.
- Each data source control wraps a particular data provider-relational databases, XML documents or custom classes and helps in:

1. Managing connection.
2. Selection of data
3. Managing presentation aspects like paging, caching etc.
4. Manipulation of data

ASP.NET - Data Sources Controls

- There are many data source controls available in ASP.Net for accessing data from SQL Server, from ODBC or OLE DB servers, from XML files and from business objects.
- Based on type of data, these controls could be divided into two categories:

1. hierarchical data source controls
2. table-based data source controls

- The data source controls used for **hierarchical data** are:

1. **XMLDataSource**-allows binding to XML files and strings with or without schema information
2. **SiteMapDataSource**-allows binding to a provider that supplies site map information

ASP.NET - Data Sources Controls

- The data source controls used for tabular data

Data source controls	Description
SqlDataSource	represents a connection to an ADO.Net data provider that returns SQL data, including data sources accessible via OLEDB and ODBC
ObjectDataSource	allows binding to a custom .Net business object that returns data

ASP.NET - Data Sources Controls

- The data source controls used for tabular data

Data source controls	Description
LinqdataSource	allows binding to the results of a Linq-to-SQL query (supported by ASP.Net 3.5 only)
AccessDataSource	represents connection to a Microsoft Access database

ASP.NET - Data Bound Controls

- The BaseDataBoundControl acts as the base class for the DataBoundControl class and the HierarchicalDataBoundControl class.
- The DataBoundControl class is an abstract class used to interact with tabular or list-style data.
- The following classes that derive from DataBoundControl:

1. AdRotator
2. List controls
3. GridView
4. FormView
5. DetailsView

ASP.NET - Data Bound Controls

- **AdRotator**
- The AdRotator control allows you to display a graphic banner on a Web page that is linked to a specific URL.
- The graphic that is displayed is rotated using properties for the control.
- The frequency of a particular ad displaying on a page can be configured using the **Impressions** property and ads can be filtered using keyword filtering.
- AdRotator controls use either an XML file or a table in a database for data. The following attributes are used in XML files to configure the AdRotator control.

ASP.NET - Data Bound Controls

- **ImageUrl** The URL of an image to display for the ad.
- **NavigateUrl** The URL that the user should be taken to when the ad is clicked. This should be URL encoded.
- **AlternateText** The alternate text that is displayed in a tooltip and read by screen readers. Also displays when the image specified by ImageUrl is not available.
- **Height** The height of the ad in pixels.
- **Width** The width of the ad in pixels.

ASP.NET - Data Bound Controls

- **Keyword** Defines a keyword that can be used when using keyword filtering. If specified, only those ads with a keyword matching the keyword filter will be displayed.
- **Impressions** A weighting number that determines how often a particular ad is likely to appear. It is relative to the impression of other ads in the same file. The maximum value of the collective impressions for all ads in an XML file is 2,048,000,000 -1.

ASP.NET - Data Bound Controls

```
<?xml version="1.0" encoding="utf-8" ?>
  <Advertisements xmlns="url">
<Ad>
<ImageUrl>~/images/ad1.gif</ImageUrl>
<NavigateUrl>http://www.mysite.com
</NavigateUrl>
<AlternateText>Ad for MyCompany, Ltd. Web
  site </AlternateText>
  <Impressions>100</Impressions>
</Ad>
```

ASP.NET - Data Bound Controls

```
<Ad> <ImageUrl>~/images/XYZ_ad.gif</ImageUrl>
    <NavigateUrl>http://www.xyz.net</NavigateUrl>
<AlternateText>Ad for XYZ Web site
    </AlternateText>
<Impressions>50</Impressions>
</Ad>
  </Advertisements>
```

ASP.NET - Data Bound Controls

- To display ads from the above XML file, add an AdRotator control to a page and set the **AdvertisementFile** property to point to the XML file as shown below:
- If you choose to use a database table as the data source for your AdRotator control, you will first need to set up a database using the following schema:

```
<asp:AdRotator ID="AdRotator1" runat="server"
    AdvertisementFile="App_Data/Ads.xml" />
```

ASP.NET - Data Bound Controls

Column name	Data type	Description
ID	int	Primary key. This column can have any name.
ImageUrl	nvarchar(*length*)	The relative or absolute URL of the image to display for the ad.
NavigateUrl	nvarchar(*length*)	The target URL for the ad. If you do not provide a value, the ad is not a hyperlink.
AlternateText	nvarchar(*length*)	The text displayed if the image cannot be found.

ASP.NET - Data Bound Controls

Column name	Data type	Description
Keyword	nvarchar(*length*)	A category for the ad on which the page can filter.
Impressions	int(4)	A number that indicates the likelihood of how often the ad is displayed. The larger the number, the more often the ad will be displayed. The total of all impressions values in the XML file may not exceed 2,048,000,000 - 1.
Width	int(4)	The width of the image in pixels.
Height	int(4)	The height of the image in pixels.

ASP.NET - Data Bound Controls

- In cases where you already have a database with a different schema, you can use the **AlternateTextField**, **ImageUrlField**, and **NavigateUrlField** properties to map the AdRotator attributes to your existing database.
- To display the data from the database in the AdRotator control, add a data source control to the page, configure the connection string for the data source control to point to your database, and set the AdRotator control's **DataSourceID** property to the ID of the data source control.

ASP.NET - Data Bound Controls

- In cases where you have a need to configure AdRotator ads programmatically, use the AdCreated event.
- The AdCreated event takes two parameters; one an object, and the other an instance of AdCreatedEventArgs.
- The AdCreatedEventArgs is a reference to the ad that is being created.

ASP.NET - Data Bound Controls

- The following code snippet sets the ImageUrl, NavigateUrl, and AlternateText for an ad programmatically:

```
protected void AdRotator1_AdCreated(object sender,
      AdCreatedEventArgs e)   {
  e.ImageUrl = "images/XYZ_ad.gif";
  e.NavigateUrl = "http://www.contoso-ltd.com/";
  e.AlternateText = "Ad for XYZ.Ltd Web site";
}
```

ASP.NET - Data Bound Controls

- **List Controls**
- List controls include the ListBox, DropDownList, CheckBoxList, RadioButtonList, and BulletedList.
- Each of these controls can be data bound to a data source. They use one field in the data source as the display text and can optionally use a second field as the value of an item.
- Items can also be added statically at design-time, and you can mix static items and dynamic items added from a data source.

ASP.NET - Data Bound Controls

- To data bind a list control, add a data source control to the page.
- Specify a SELECT command for the data source control and then set the DataSourceID property of the list control to the ID of the data source control.
- Use the **DataTextField** and **DataValueField** properties to define the display text and the value for the control.

ASP.NET - Data Bound Controls

- **GridView**
- The GridView control allows for tabular data display and editing using a declarative approach and is the successor to the DataGrid control. The following features are available in the GridView control.
- Binding to data source controls, such as SqlDataSource.
- Built-in sorting capabilities.
- Built-in updating and deleting capabilities.
- Built-in paging capabilities.

ASP.NET - Data Bound Controls

- Built-in row selection capabilities.
- Programmatic access to the GridView object model to dynamically set properties, handle events, and so on.
- Multiple key fields.
- Multiple data fields for the hyperlink columns.
- Customizable appearance through themes and styles.

ASP.NET - Data Bound Controls

- Each column in the GridView control is represented by a DataControlField object.
- By default, the AutoGenerateColumns property is set to **true**, which creates an AutoGenerated Field object for each field in the data source.
- You can also manually control which column fields appear in the **GridView** control by setting the **AutoGenerateColumns** property to **false** and then defining your own column field collection.
- Different column field types determine the behavior of the columns in the control.

ASP.NET - Data Bound Controls

Column field type	Description
BoundField	Displays the value of a field in a data source. This is the default column type of the GridView control.
ButtonField	Displays a command button for each item in the GridView control. This allows you to create a column of custom button controls, such as the Add or the Remove button.
CommandField	Displays predefined command buttons to perform selecting, editing, or deleting operations.

ASP.NET - Data Bound Controls

Column field type	Description
CheckBoxField	Displays a check box for each item in the GridView control. This column field type is commonly used to display fields with a Boolean value.
HyperLinkField	Displays the value of a field in a data source as a hyperlink. This column field type allows you to bind a second field to the hyperlink's URL.
ImageField	Displays an image for each item in the GridView control.
TemplateField	Displays user-defined content for each item in the GridView control according to a specified template. This column field type allows you to create a custom column field.

ASP.NET - Data Bound Controls

- **FormView**
- The FormView control is used to display a single record from a data source.
- It is similar to the DetailsView control, except it displays user-defined templates instead of row fields.
- Creating your own templates gives you greater flexibility in controlling how the data is displayed. The FormView control supports the following features:
- Binding to data source controls, such as SqlDataSource and ObjectDataSource.

ASP.NET - Data Bound Controls

- Built-in inserting capabilities.
- Built-in updating and deleting capabilities.
- Built-in paging capabilities.
- Programmatic access to the FormView object model to dynamically set properties, handle events, and so on.
- Customizable appearance through user-defined templates, themes, and styles.

ASP.NET - Data Bound Controls

- **DetailsView**
- The DetailsView control is used to display a single record from a data source in a table, where each field of the record is displayed in a row of the table. It can be used in combination with a GridView control for master-detail scenarios. The DetailsView control supports the following features:
- Binding to data source controls, such as SqlDataSource.
- Built-in inserting capabilities.

ASP.NET - Data Bound Controls

- Built-in updating and deleting capabilities.
- Built-in paging capabilities.
- Programmatic access to the DetailsView object model to dynamically set properties, handle events, and so on.
- Customizable appearance through themes and styles.