.NET Framework 4 - ASP.NET

**ASP.NET Data Access Overview**

Web applications commonly access data sources for storage and retrieval of dynamic data. You can write code to access data using classes from the [System.Data](http://msdn.microsoft.com/en-us/library/system.data.aspx) namespace (commonly referred to as ADO.NET) and from the [System.Xml](http://msdn.microsoft.com/en-us/library/system.xml.aspx) namespace. This approach was common in previous versions of ASP.NET.

However, ASP.NET also enables you to perform data binding declaratively. This requires no code at all for the most common data scenarios, including:

* Selecting and displaying data.
* Sorting, paging, and caching data.
* Updating, inserting, and deleting data.
* Filtering data using run-time parameters.
* Creating master-detail scenarios using parameters.

ASP.NET includes several types of server controls that participate in the declarative data binding model, including data source controls, data-bound controls, and the query extender control. These controls manage the underlying tasks that are required by the stateless Web model in order to display and update data in ASP.NET Web pages. The controls let you add data-binding behavior to a page without having to understand details of the page request life cycle.

http://i.msdn.microsoft.com/Global/Images/clear.gif Data Source Controls

Data source controls are ASP.NET controls that manage the tasks of connecting to a data source and reading and writing data. Data source controls do not render any user interface, but instead act as an intermediary between a particular data store (such as a database, business object, or XML file) and other controls on the ASP.NET Web page. Data source controls enable rich capabilities for retrieving and modifying data, including querying, sorting, paging, filtering, updating, deleting, and inserting. Data source controls derive from [ContextDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.contextdatasource.aspx), which is the base class that provides the context type that data source controls use. This base class enables you to create data source controls that support data models such as Entity Framework and WCF Data Services.

ASP.NET includes the following data source controls:

|  |  |
| --- | --- |
| **Data source control** | **Description** |
| [AccessDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.accessdatasource.aspx) | Enables you to work with a Microsoft Access database.  For more information, see [AccessDataSource Web Server Control Overview](http://msdn.microsoft.com/en-us/library/b277ts6z.aspx). |
| [LinqDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.linqdatasource.aspx) | Enables you to use Language-Integrated Query (LINQ) in an ASP.NET Web page through declarative markup in order to retrieve and modify data from a data object. Supports automatic generation of select, update, insert, and delete commands. The control also supports sorting, filtering, and paging. |
| [ObjectDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.objectdatasource.aspx) | Enables you to work with a business object or other class and create Web applications that rely on middle-tier objects to manage data.  For more information, see [ObjectDataSource Web Server Control Overview](http://msdn.microsoft.com/en-us/library/9a4kyhcx.aspx). |
| [SiteMapDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.sitemapdatasource.aspx) | Used with ASP.NET site navigation. For more information, see [ASP.NET Site Navigation](http://msdn.microsoft.com/en-us/library/e468hxky.aspx). |
| [SqlDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.sqldatasource.aspx) | Enables you to work with ADO.NET managed data providers, which provide access to Microsoft SQL Server, OLE DB, ODBC, or Oracle databases.  For more information, see [SqlDataSource Web Server Control Overview](http://msdn.microsoft.com/en-us/library/dz12d98w.aspx). |
| [EntityDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.entitydatasource.aspx) | Enables you to bind to data that is based on the Entity Data Model (EDM). Supports automatic generation of update, insert, delete, and select commands. The control also supports sorting, filtering and paging.  For more information, see [EntityDataSource Web Server Control Overview](http://msdn.microsoft.com/en-us/library/cc488502.aspx). |
| [XmlDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.xmldatasource.aspx) | Enables you to work with an XML file, which is especially useful for hierarchical ASP.NET server controls such as the [TreeView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.treeview.aspx) or [Menu](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.menu.aspx) control.  For more information, see [XmlDataSource Web Server Control Overview](http://msdn.microsoft.com/en-us/library/494y92bs.aspx). |

Data-source controls can also be extended to support additional data access storage providers.

For more information on data source controls, see [Data Source Controls Overview](http://msdn.microsoft.com/en-us/library/ms227679.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif QueryExtender Control

Filtering data in a data-driven Web site typically requires that you create a **Where** clause in the query that is associated with the data source control. This method of filtering data can be difficult, and in some cases it does not expose the full functionality of the underlying data source. In order to make it easier to filter data in a page, ASP.NET provides the [QueryExtender](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.queryextender.aspx) control.

You can use the [QueryExtender](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.queryextender.aspx) control to do the following:

* Provide string-search capabilities in a Web page.
* Define a range of values to search.
* Compare a specified value to a property value in a table.
* Provide sorting capabilities for the filtered data.
* Define custom filtering.

Using the [QueryExtender](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.queryextender.aspx) control for filtering data is more efficient than using a **Where** clause because you do not have to learn the query language for the data provider that is used by the query extender control.

The [EntityDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.entitydatasource.aspx) and [LinqDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.linqdatasource.aspx) controls support the [QueryExtender](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.queryextender.aspx) control.

For more information, see [QueryExtender Web Server Control Overview](http://msdn.microsoft.com/en-us/library/dd537671.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Data-Bound Controls

Data-bound controls render data as markup to the requesting browser. A data-bound control can bind to a data source control and automatically fetch data at the appropriate time in the page request lifecycle. Data-bound controls can take advantage of the capabilities provided by a data source control including sorting, paging, caching, filtering, updating, deleting, and inserting. A data-bound control connects to a data source control through its [DataSourceID](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.basedataboundcontrol.datasourceid.aspx) property.

ASP.NET includes the data-bound controls described in the following table.

List controls

Renders data in a variety of lists format. List controls include the [BulletedList](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.bulletedlist.aspx), [CheckBoxList](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.checkboxlist.aspx), [DropDownList](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.dropdownlist.aspx), [ListBox](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.listbox.aspx), and [RadioButtonList](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.radiobuttonlist.aspx) controls.

[AdRotator](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.adrotator.aspx)

Renders advertisements on a page as an image that users can click to go to a URL associated with the advertisement.

For more information, see [AdRotator Web Server Control Overview](http://msdn.microsoft.com/en-us/library/edx4dac3.aspx).

[DataList](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.datalist.aspx)

Renders data in a table. Each item is rendered using an item template that you define.

For more information see the [DataList Web Server Control Overview](http://msdn.microsoft.com/en-us/library/es4e4e0e.aspx).

[DetailsView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.detailsview.aspx)

Displays one record at a time in a tabular layout and enables you to edit, delete, and insert records. You can also page through multiple records.

For more information see the [DetailsView Web Server Control Overview](http://msdn.microsoft.com/en-us/library/s3w1w7t4.aspx).

[FormView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.formview.aspx)

Similar to the [DetailsView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.detailsview.aspx) control, but enables you to define a free-form layout for each record. The [FormView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.formview.aspx) control is like a [DataList](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.datalist.aspx) control for a single record.

For more information, see [FormView Web Server Control Overview](http://msdn.microsoft.com/en-us/library/ms227992.aspx).

[GridView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.gridview.aspx)

Displays data in a table and includes support for editing, updating, deleting, sorting, and paging data without requiring code.

|  |
| --- |
| **NoteNote** |
| The [GridView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.gridview.aspx) control supersedes the ASP.NET [DataGrid](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.datagrid.aspx) control available in previous versions of ASP.NET. |

For more information, see [GridView Web Server Control Overview](http://msdn.microsoft.com/en-us/library/2s019wc0.aspx).

[ListView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.listview.aspx)

Enables you to define the data layout by using templates. Supports automatic sort, edit, insert, and delete operations. You can also enable paging by using an associated [DataPager](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.datapager.aspx) control.

For more information, see [ListView Web Server Control Overview](http://msdn.microsoft.com/en-us/library/bb398790.aspx).

[Menu](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.menu.aspx)

Renders data in a hierarchical dynamic menu that can include submenus.

For more information, see [Menu Control Overview](http://msdn.microsoft.com/en-us/library/ecs0x9w5.aspx).

[Repeater](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.repeater.aspx)

Renders data in a list. Each item is rendered using an item template that you define.

For more information, see [Repeater Web Server Control Overview](http://msdn.microsoft.com/en-us/library/x8f2zez5.aspx).

[TreeView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.treeview.aspx)

Renders data in a hierarchical tree of expandable nodes.

For more information, see [TreeView Web Server Control Overview](http://msdn.microsoft.com/en-us/library/e8z5184w.aspx).

|  |
| --- |
| **NoteNote** |
| The [DataGrid](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.datagrid.aspx) control available in ASP.NET version 1.0 and version 1.1 has been superseded by the [GridView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.gridview.aspx) control, which includes expanded capabilities for sorting, paging, and modifying data. Existing pages that use the [DataGrid](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.datagrid.aspx) control will continue to function. As with other data controls, the [DataGrid](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.datagrid.aspx) control has been enhanced to interact with data source controls. |
| **NoteNote** |
| The [ListView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.listview.aspx) control supersedes the [Repeater](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.repeater.aspx) control and the [DataList](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.datalist.aspx) control. Existing pages that use those controls will continue to function. The [ListView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.listview.aspx) control simplifies the implementation of many common scenarios. |

For more information, see [ASP.NET Data-Bound Web Server Controls Overview](http://msdn.microsoft.com/en-us/library/ms228214.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif LINQ

Language-Integrated Query (LINQ) provides a unified programming model for querying and updating data from different types of data sources, and extends data capabilities directly into the C# and Visual Basic languages. LINQ applies the principles of object-oriented programming to relational data. To work with LINQ, you can use the [LinqDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.linqdatasource.aspx) control. You can also create LINQ queries directly in order to access data from a Web page. For more information, see [Using LINQ with ASP.NET](http://msdn.microsoft.com/en-us/library/bb907622.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Dynamic Data

ASP.NET Dynamic Data is a framework that enables you to create data-driven ASP.NET Web applications quickly. Dynamic Data automatically discovers the data model at run time and determines UI behavior from the data model. A scaffolding framework instantly provides a functional Web site for displaying and editing data. This scaffolding can then be customized using metadata, templates, or by creating standard ASP.NET pages to override the default behavior. Existing ASP.NET Web applications can easily integrate pieces of the scaffolding logic into their web pages. For more information about Dynamic Data, see [ASP.NET Dynamic Data Content Map](http://msdn.microsoft.com/en-us/library/cc488545.aspx)

You can add Dynamic Data features to an ASP.NET Web Application without using scaffolding. When Dynamic Data is enabled in an ASP.NET Web application, the Web application can take advantage of the following features:

* Run-time data field validation.
* The ability to use in-memory page templates.
* The ability to apply custom metadata to various data fields by using attributes from the [System.ComponentModel.DataAnnotations](http://msdn.microsoft.com/en-us/library/system.componentmodel.dataannotations.aspx) namespace.