

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

منابع داده / ASP.NET – Data Source

مدرس : مهندس افشین رفوآ

منابع داده / ASP.NET – Data Source

کنترل‌های **Databound** برای نمایش دادن داده به کاربر در برنامه ی تحت وب بکار می روند. با استفاده از کنترل های **Databound** می توان داده ها را به راحتی در اپلیکیشن مدیریت و پردازش (**manipulate**) کرد.

کنترل های **data-bound** به خاصیت **DataSource** متصل هستند.

کنترل های **Databound** در واقع کنترل های مرکبی هستند که دیگر کنترل های **ASP.NET** مثل **textbox** ها، **radio button** و غیره را باهم ترکیب می کنند.

یک کنترل **data source** با کنترل های **Data bound** تعامل و ارتباط برقرار کرده و از این طریق فرایندهای پیچیده ی مقید سازی و اتصال داده ها (**data binding**) را مخفی می سازد. اینها ابزاری هستند که داده ها را در اختیار کنترل های **data bound** قرار می دهند و اجرای عملیاتی همچون درج (**insertion**)، حذف (**deletion**)، مرتب سازی (**sorting**) و بروز رسانی (**update**) را پشتیبانی می کنند.

هر کنترل **data source** یک ارائه دهنده ی داده (**data provider**) معین از جمله پایگاه داده رابطه ای (**relational database**)، **سند XML** یا کلاس سفارشی را دربرمی گیرد و همچنین در انجام عملیات زیر به شما

کمک می کند:

مدیریت اتصال

انتخاب داده

ذخیره سازی داده در حافظه ی پنهان (caching) ، صفحه بندی (paging)

مدیریت و پردازش داده

کنترل های Data source فراوانی در ASP.NET وجود دارد که امکان دسترسی به داده از SQL Server ، سرور

های ODBC و OLE DB ، فایل های XML و غیره .. فراهم می کنند.

می توان این کنترل ها را بر مبنای نوع داده به دو دسته تقسیم کرد:

کنترل های منبع داده سلسله مراتبی (Hierarchical data source controls)

کنترل های منبع داده مبتنی بر جدول (Table-based data source controls)

کنترل های داده ای (data source control) که ویژه ی داده های سلسله مراتبی (Hierarchical data) بکار می

روند به ترتیب زیر می باشند:

XMLDataSource – برای اتصال به فایل ها و رشته های xml با / بدون اطلاعات شیما (schema) . کنترل

XmlDataSource ، دارای قابلیت خواندن و نوشتن در سند های XML می باشد ، از این رو می توان از آن برای

اتصال، دریافت اطلاعات و بروز رسانی سند های XML استفاده کنید . این کنترل در واقع ارائه دهنده ی داده از

منابع XML برای کنترل های **TreeView** و **Menu** است که با ساختار ترتیبی و درختی که ویژه ی نوع داده ای فایل

های XML است ، کار می کنند .

SiteMapDataSource – امکان اتصال به ارائه دهنده ی داده (data provider) که اطلاعات نقشه ی سایت

(site map) را تامین می کنند، فراهم می آورد. کنترل **SiteMapDataSource** برای کار با site map ها در

ASP.NET تعبیه شده است و قابلیت ایجاد و دریافت اطلاعات مورد نیاز برای عملیات پیمایش در سایت را داراست

. site map یک شرح از تمامی صفحه ها و زیر صفحه های موجود در یک سایت است، که با استفاده از زبان نشانه

گذاری XML طراحی می شود . این کنترل توانایی اتصال به این نوع فایل ها را داراست و کنترل های دیگر همانند

Menu و یا **TreeView** می توانند از امکانات و اطلاعات فراهم شده توسط این کنترل ، برای ایجاد منوهای ترتیبی

جهت پیمایش سایت استفاده کنند .

کنترل های داده ای که برای داده های مبتنی بر جدول بکار می روند، ذیل فهرست شده اند:

کنترل های داده	شرح
<p>SqlDataSource</p>	<p>کنترل SqlDataSource برای اتصال به یک پایگاه داده SQL Server، ACCESS، OLB DB و یا Oracle استفاده می شود. این کنترل قادر به دریافت داده از منابع نام برده و نیز حذف، اضافه و یا بهروز رسانی اطلاعات در آنها را می باشد.</p> <p>قابلیت اتصال به یک تامپین کننده ی داده ADO.NET را فراهم می کند که داده های SQL را برمی گرداند. داده های SQL می تواند شامل منابع داده ای باشد که از طریق OLEDB و QDBC قابل دسترسی است.</p>
<p>ObjectDataSource</p>	<p>این کنترل امکان اتصال به یک شیء Net business. سفارشی را بوجود آورده که داده برمی گرداند. به عبارتی دیگر این کنترل نشانگر یک business object است که داده در معماری های برنامه ی تحت وب چند لایه ای در اختیار کنترل های data-bound می گذارد.</p>
<p>LinqdataSource</p>	<p>امکان متصل شدن و دسترسی به نتایج یک query (پرسمان) Linq-to-SQL (از linq به sql) را فراهم می آورد.</p> <p>این کنترل با ارائه ی امکان استفاده از LINQ در صفحات asp.net از طریق متن نشانه گذاری شده (markup text) داده ها را از data object بازمی آید کرده و آن ها را اصلاح می کند.</p>
<p>AccessDataSource</p>	<p>اتصال به پایگاه داده ی Microsoft Access را امکان پذیر می کند. کنترل AccessDataSource یک نسخه ویژه از کنترل SqlDataSource است که برای اتصال به یک پایگاه داده Microsoft Access تعبیه شده است و قادر به برقراری ارتباط با فایل های .mdb می باشد(مشابه کنترل SqlDataSource، می توان دستورات صریح و مستقیمی برای اعمال مختلف این کنترل از جمله Update، Insert، Select به زبان SQL تعیین کرد).</p>

Data Source Views

Data source views در واقع اشیایی از کلاس **DataSourceView** هستند که بیانگر view های (نمایه) سفارشی از عملیات مختلف مربوط به داده همچون مرتب سازی (sort)، فیلتر کردن (**filtering**) و غیره .. می باشد. **DataSourceView** در حقیقت کلاس پایه تمامی کلاس های **data source view** می باشد. کلاس مذکور قابلیت های مختلف کنترل های **data source** را تعریف می کند.

این کلاس خاصیت هایی دارد که در جدول زیر برای شما لیست شده:

خاصیت	شرح
CanDelete	تعیین می کند آیا امکان پاک کردن داده از منبع داده زیرین (underlying data source) وجود دارد یا خیر. این خاصیت مقداری برمی گرداند که نشان می دهد آیا شیء DataSourceView مرتبط با شیء DataSourceControl جاری از عملیات ExecuteDelete پشتیبانی می کند یا خیر.
CanInsert	مقداری برمی گرداند که این مقدار نشان می دهد آیا امکان درج داده در پایگاه داده زیرین (underlying database) وجود دارد یا خیر. مقداری بازیابی می کند که تعیین می کند آیا کنترل منبع داده مربوطه (data source control) با پیکربندی کنونی قابلیت اجرای عملیات درج (insertion operation) را دارد یا خیر.
CanPage	این خاصیت تعیین می کند آیا امکان صفحه بندی (paging) در data source زیرین وجود دارد یا خیر. مقداری برگردانده می شود که این مقدار مشخص می کند آیا شیء DataSourceView مرتبط با شیء DataSourceControl جاری از قابلیت صفحه بندی به وسیله ی داده ای که از متد ExecuteSelect

	بازیابی کرده پشتیبانی می کند یا خیر.
CanRetrieveTotalRowCount	مشخص می کند آیا امکان بازیابی اطلاعات درباره ی تعداد کل سطرهای داده (data row) وجود دارد یا خیر.
CanSort	این خاصیت تعیین می کند آیا امکان مرتب سازی داده وجود دارد یا خیر.
CanUpdate	این property تعیین می کند آیا امکان بروز رسانی داده وجود دارد یا خیر.
Events	این خاصیت فهرستی از delegate های مدیریت کننده ی رخداد(-event handler) برمی گرداند که برای data source view بکار گرفته می شود.
Name	این خاصیت اسم data source view را برمی گرداند.

جدول زیر توابع این کلاس **DataSourceView** را ارائه می دهد:

متد	شرح
CanExecute	تعیین می کند آیا دستور (command) مشخص شده می تواند اجرا شود یا خیر.
ExecuteCommand	این تابع دستور (command) معین را اجرا می کند.
ExecuteDelete	عملیات پاک کردن (deletion) را روی لیست داده ای که متصل به شیء DataSourceView است اجرا می کند.
ExecuteInsert	عملیات درج را روی لیست داده پیاده می کند.

ExecuteSelect	لیستی از داده ها از محل ذخیره سازی داده (data storage) انتخاب کرده و برمی گرداند.
ExecuteUpdate	عمل بروز رسانی داده ها را روی لیست داده که متعلق به شیء DataSourceView می باشد اجرا می کند.
Delete	این تابع عملیات پاک کردن (deletion) را به طور ناهمزمان روی لیست داده (مربوط به شیء DataSourceView) پیاده (اجرا) می کند.
Insert	عملیات درج را روی داده های مربوط به شیء DataSourceView اجرا می کند.
Select	داده ی درخواست (query) شده را باز می گرداند. لیستی از داده ها به طور ناهمگام از محل ذخیره ی داده برمی گرداند.
Update	داده های مربوط به شیء DataSourceView را بروز رسانی می کند / عملیات بروز رسانی را روی داده ی های مرتبط با شیء DataSourceView اجرا می کند.
OnDataSourceViewChanged	این تابع رخداد (event) DataSourceViewChanged را فعال می کند.
RaiseUnsupportedCapabilitiesError	این تابع توسط متد RaiseUnsupportedCapabilitiesError به منظور مقایسه ی قابلیت های مورد نیاز (درخواست شده ی) عملیات ExecuteSelect با قابلیت هایی که توسط view پشتیبانی می شود فراخوانده می شود.

کنترل **SqlDataSource**

کنترل **SqlDataSource** برای اتصال به یک پایگاه داده Oracle ، ACCESS ، OLB DB ، SQL Server و یا استفاده می شود . این کنترل قادر به دریافت داده از منابع نام برده و نیز حذف ، اضافه و یا بهروز رسانی اطلاعات در آنها را می باشد. اتصال و دسترسی به داده با بکارگیری دو خاصیت **ConnectionString** و **ProviderName** امکان پذیر می باشد.

همان طور که مشاهده می کنید تکه کد زیر نحوه ی کلی نگارش (syntax) کنترل مزبور را نمایش می دهد:

```
<asp:SqlDataSource runat="server" ID="MySQLSource"
ProviderName='<%"$ ConnectionStrings:LocalNWind.ProviderName %>'
ConnectionString='<%"$ ConnectionStrings:LocalNWind %>'
SelectionCommand= "SELECT * FROM EMPLOYEES" />
<asp:GridView ID="GridView2" runat="server" DataSourceID="MySQLSource" />
```

پیکربندی عملیات مختلف که روی داده های زیرین (underlying data) پیاده می شود به مجموعه خاصیت های کنترل **data source** بستگی دارد.

مجموعه خاصیت های کنترل **SqlDataSource** که رابط برنامه نویسی (programming interface) این کنترل را فراهم می کنند، در جدول زیر مشاهده می کنید:

Property Group (مجموعه خاصیت)	شرح
DeleteCommand, DeleteParameters, DeleteCommandType	نوع، پارامتر و دستور SQL ویژه ی پاک کردن سطرها را خوانده یا تنظیم می کند.
FilterExpression, FilterParameters	پارامترها و رشته های filtering (فیلتر) داده را بازیابی کرده یا مقداردهی می کند.
InsertCommand, InsertParameters,	نوع (type)، پارامترها و دستور SQL ویژه ی درج (insert) سطر (row) در پایگاه داده را برگردانده یا تنظیم می کند.

InsertCommandType	
SelectCommand, SelectParameters, SelectCommandType	نوع (type)، پارامتر و دستور SQL ویژه ی بازبایی سطرها از پایگاه داده را بازگردانی کرده یا مقداردهی می کند.
SortParameterName	اسم پارامتر ورودی (input parameter) که رویه ی آماده (stored procedure) دستور (command) مورد نظر از آن به منظور مرتب سازی داده ها کمک می گیرد، بازبایی کرده یا تنظیم می کند.
UpdateCommand, UpdateParameters, UpdateCommandType	نوع (type)، پارامترها و دستور SQL ویژه ی بروز رسانی سطرهای (row) پایگاه داده را بازگردانده یا مقداردهی می کند.

در تکه کد (code snippet) زیر مشاهده می کنید که یک کنترل **data source** (منبع داده) به منظور دستکاری و مدیریت داده (**data manipulation**) فعال سازی شده:

```
<asp:SqlDataSource runat="server" ID="MySQLSource"
ProviderName='<%"$ ConnectionStrings:LocalNWind.ProviderName %>'
ConnectionString=' <%"$ ConnectionStrings:LocalNWind %>'
SelectCommand="SELECT * FROM EMPLOYEES"
UpdateCommand="UPDATE EMPLOYEES SET LASTNAME=@lname"
DeleteCommand="DELETE FROM EMPLOYEES WHERE EMPLOYEEID=@eid"
FilterExpression="EMPLOYEEID > 10">.....
.....
</asp:SqlDataSource>
```

کنترل **ObjectDataSource**

کنترل **ObjectDataSource** با فعال سازی کلاس های **user-dfined** (کلاس های تعریف و ارائه شده توسط کاربر) خروجی (**output**) توابع کلاس های مذکور را به کنترل های **data bound** مربوط ساخته و متصل می کند. رابط برنامه سازی (**programming interface**) این کلاس تقریباً با کنترل **SqlDataSource** یکسان است.

زیر دو جنبه ی بسیار مهم متصل یا مقید سازی **business object** ها را مشاهده می کنید:

کلاس مورد نظر باید یک سازنده (**constructor**) پیش فرض داشته باشد، ناپایدار (**stateless**) بوده و متدهایی داشته باشد که بتوان برای انتخاب، بروز رسانی، درج و حذف (**semantics** لغات و معانی) نگاهت کرد.

شئ مورد نظر باید آیتم ها را به صورت تکی بروز رسانی کند، عملیات دسته ای و گروهی پشتیبانی نمی شوند.

در این مرحله برای آشنایی هر چه بیشتر شما یک مثال کاربردی ارائه می دهیم. کلاس **Student** کلاسی است که باید با یک **object data source** بکار گرفته شود. کلاس ذکر شده دارای سه خاصیت (**property**) می باشد:

student id، **name** و **city**. این کلاس همچنین دارای یک سازنده (**constructor**) پیش فرض و متد

GetStudents برای بازیابی داده است:

کلاس **Student**:

```
using System.Data;
public class Student
{
    public int StudentID { get; set; }
    public string Name { get; set; }
    public string City { get; set; }

    public Student()
    { }

    public DataSet GetStudents()
    {
        DataSet ds = new DataSet();
        DataTable dt = new DataTable("Students");

        dt.Columns.Add("StudentID", typeof(System.Int32));
        dt.Columns.Add("StudentName", typeof(System.String));
        dt.Columns.Add("StudentCity", typeof(System.String));
        dt.Rows.Add(new object[] { 1, "M. H. Kabir", "Calcutta" });
        dt.Rows.Add(new object[] { 2, "Ayan J. Sarkar", "Calcutta" });
        ds.Tables.Add(dt);

        return ds;
    }
}
```

با دنبال کردن مراحل زیر شئ مورد نظر را به **object data source** متصل ساخته و داده ها را بازیابی کنید:

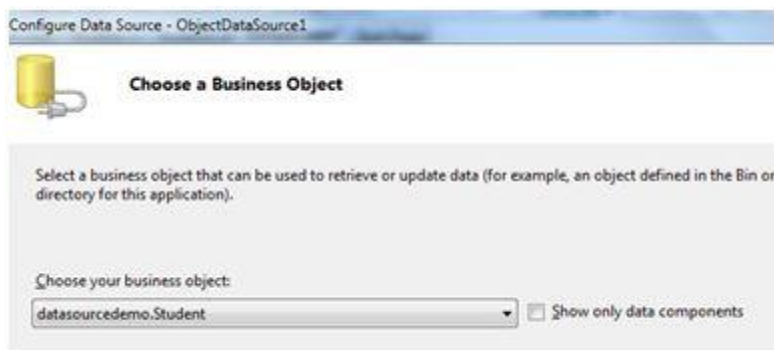
ابتدا یک وب سایت جدید ایجاد کنید.

حال یک کلاس (**Students.cs**) به وب سایت خود اضافه کنید. برای این منظور در پنجره ی **solution explorer** روی پروژه راست کلیک کرده، یک قالب کلاس (**class template**) اضافه کنید، سپس کد فوق را در آن جای گذاری کنید.

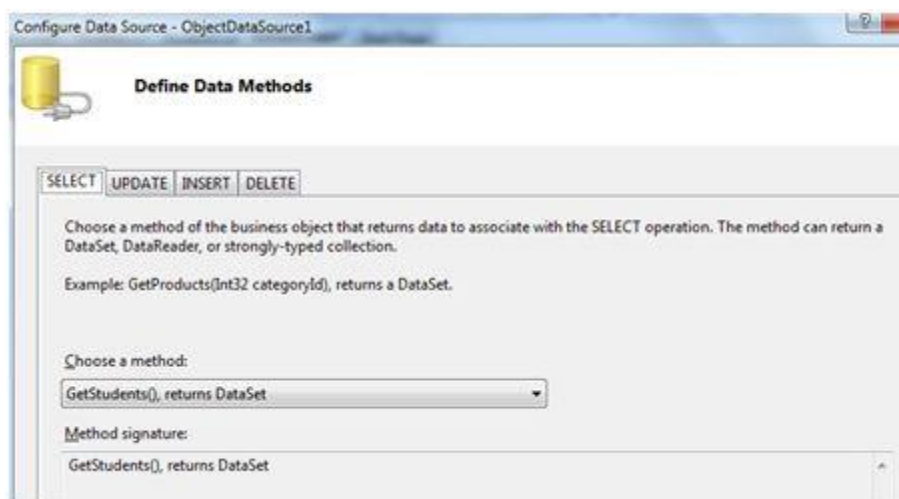
Solution را ساخته تا **application** مورد نظر بتواند از ارجاع (**reference**) به کلاس استفاده کند.

یک کنترل **object data source** در فرم وب (**web form**) قرار دهید.

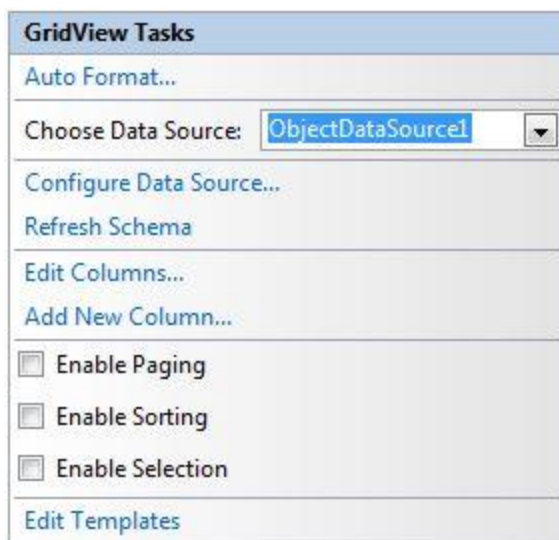
با انتخاب **object**، **data source** را پیکربندی (**configure**) کنید.



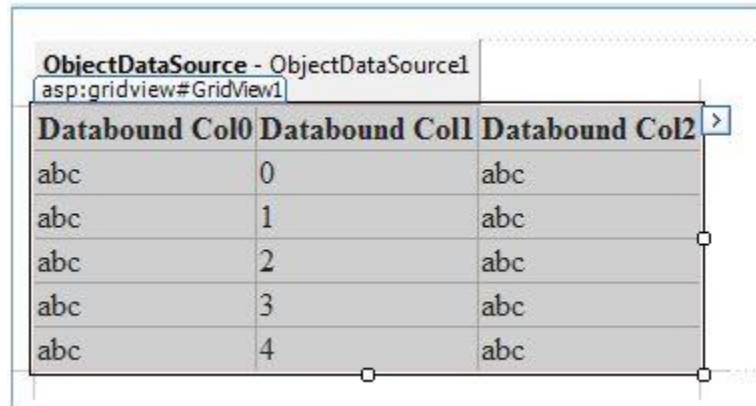
متد یا متدهایی را برای اجرای عملیات مختلف روی داده انتخاب کنید. در این مثال تنها یک متد بکار رفته است.



اکنون یک کنترل **data bound** همچون **grid view** روی صفحه قرار داده و **object data source** را به عنوان منبع داده ی زیرین (**underlying datasource**) انتخاب کنید.

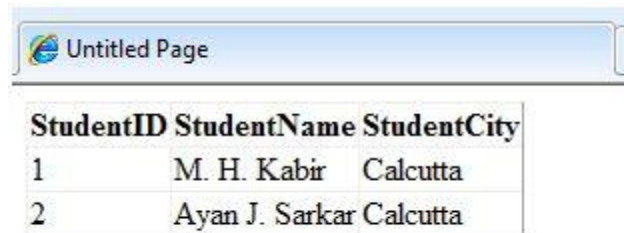


در این مرحله **design view** ظاهری همچون تصویر زیر خواهد داشت:



Databound Col0	Databound Col1	Databound Col2
abc	0	abc
abc	1	abc
abc	2	abc
abc	3	abc
abc	4	abc

پروژه را اجرا کنید، خواهی دید که **tuple** های (مجموعه ای از مقادیر مرتبط با هم) **hard code** شده از کلاس **students** بازیابی می شود.



StudentID	StudentName	StudentCity
1	M. H. Kabir	Calcutta
2	Ayan J. Sarkar	Calcutta

کنترل **AccessDataSource**

اتصال به پایگاه داده ی **Microsoft Access** را امکان پذیر می کند. کنترل **AccessDataSource** یک نسخه ویژه از کنترل **SqlDataSource** است که برای اتصال به یک پایگاه داده **Microsoft Access** تعبیه شده است و قادر به برقراری ارتباط با فایل های **mdb** می باشد. این کنترل در واقع مبتنی بر پایگاه داده ی **SqlDataSource** بوده و رابط برنامه نویسی (**programming interface**) ساده تری نسبت به آن ارائه می دهد.

تکه کد زیر دستور نگارش کنترل مزبور را به نمایش می گذارد:

```
<asp:AccessDataSource ID="AccessDataSource" runat="server"
DataFile="~/App_Data/ASPDotNetStepByStep.mdb" SelectCommand="SELECT * FROM
[DotNetReferences]">
</asp:AccessDataSource>
```

توجه داشته باشید که کنترل **AccessDataSource** فقط پایگاه داده را در حالت **read-only** اجرا می کند. با این وجود می توان از آن برای اجرای عملیاتی همچون درج (**insertion**)، بروز رسانی (**update**) یا حذف (**delete**) استفاده کرد. این کار با استفاده از دستورات **ADO.NET** و مجموعه ای از پارامترها (**parameter collection**) امکان پذیر می باشد.

بروز رسانی برای پایگاه داده ی **Access** به خصوص از داخل یک اپلیکیشن **ASP.NET** مشکل ساز است زیرا که پایگاه داده ی **Access** تنها یک فایل ساده است و همچنین حساب (**account**) پیش فرض **ASP.NET** ممکن است مجوز نوشتن را در فایل پایگاه داده نداشته باشد.