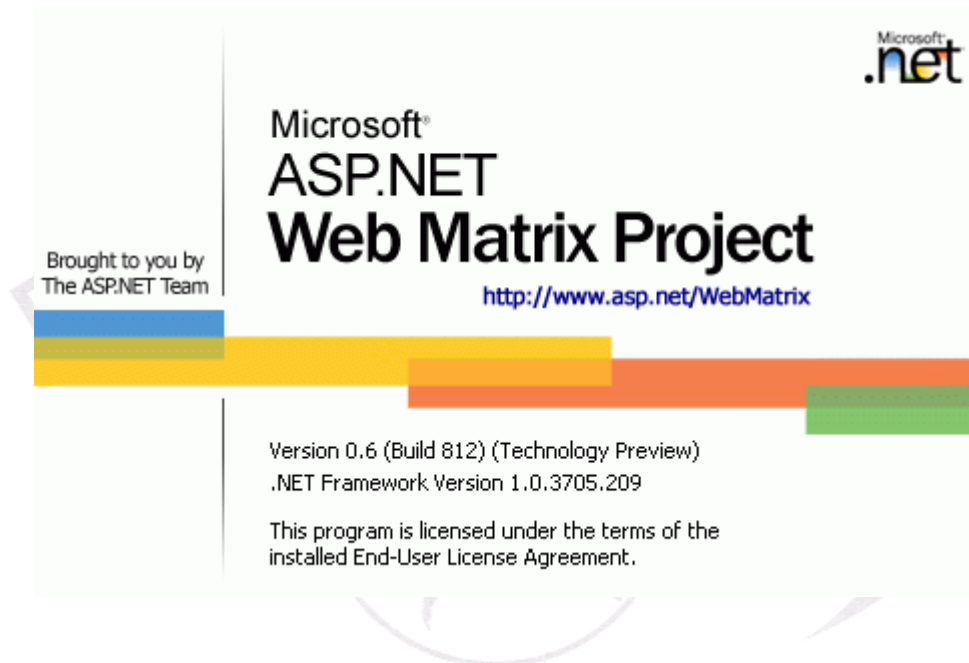


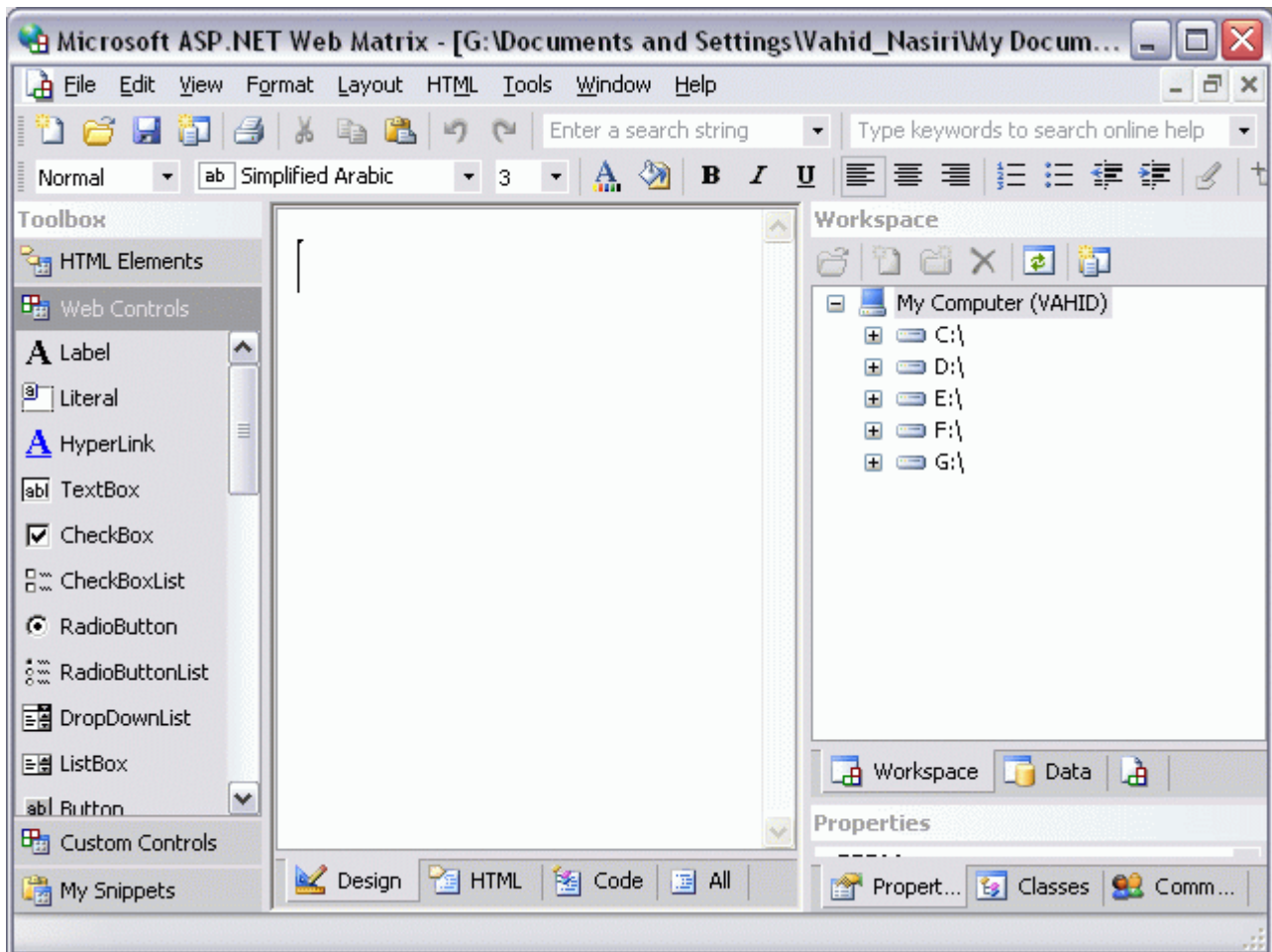
## فصل هفتم :

### آشنایی با کنترل های WebMatrix و بکارگیری آنها



#### مقدمه :

در نگاه اول WebMatrix شبیه به محیط توسعه ای ساده شده برای ساخت برنامه های ASP.NET به نظر می رسد. اما قابلیت های آن بسیار بیشتر از این مورد است. این مجموعه علاوه بر توانایی تولید صفحات وب ، توانایی ایجاد کنترل های کاربر، فایل های کلاس ، وب سرویس ها و غیره را دارا است. همچنین این مجموعه دارای Wizard هایی متعدد و یک وب سرور مخصوص خودش نیز می باشد. بعلاوه سایر موارد افزودنی (Add-ins) برای آن مهیا است و شما نیز می توانید اینگونه برنامه های افزودنی را برای آن تولید نمایید.



شکل ۱- نمایشی از محیط وب ماتریکس.

## تفاوت های WebMatrix و VS.NET :

شاید این سوال مطرح شود که چرا مایکروسافت دو محیط را برای توسعه ی برنامه های ASP.NET ارائه داده است (شکل ۱). به صورت خلاصه تفاوت های این دو محیط به شرح زیر هستند:

- در VS.NET با یک برنامه به صورت پروژه برخورد می شود و می توان به آن انواع فایلها و منابع را افزود. در وب ماتریکس هر فایل موجودیتی مستقل دارد.



- وب ماتریکس از روش inline-code برای صفحات خودش استفاده می نماید در حالیکه VS.NET از روش Code-behinde سود می جوید.
- وب ماتریکس حدود یک مگابایت و بسیار سریع می باشد.
- برخلاف VS.NET ، وب ماتریکس به صورت خودکار فایل های کلاس های برنامه را به صورت اسمبلی کامپایل نمی کند. اینکار باید از طریق خط فرمان صورت گیرد.
- در وب ماتریکس امکان استفاده از راهنمای دات نت فریم ورک مهیا نیست.
- وب ماتریکس اساسا پروژه ای است که در انجمن <http://asp.net/webmatrix/> توسعه و تکامل پیدا می کند.
- قیمت : وب ماتریکس مجانی است!

و در حالت کلی این مجموعه تا بحال ۸۰ درصد توانایی های ساخت یک برنامه ی ASP.NET را کسب کرده است. در ادامه گوشه ای از امکانات مهم آنرا با هم مرور خواهیم کرد ( البته بدیهی است که اصول کار در اینجا با VS.NET فرقی نمی کند و صرفا هدف از این فصل آشنا شدن با یک محیط جدید و با آتیه است ) :

مکان تهیه ی وب ماتریکس در اینترنت :

book

[http://asp.net/webmatrix/web%20matrix\\_doc.pdf](http://asp.net/webmatrix/web%20matrix_doc.pdf)

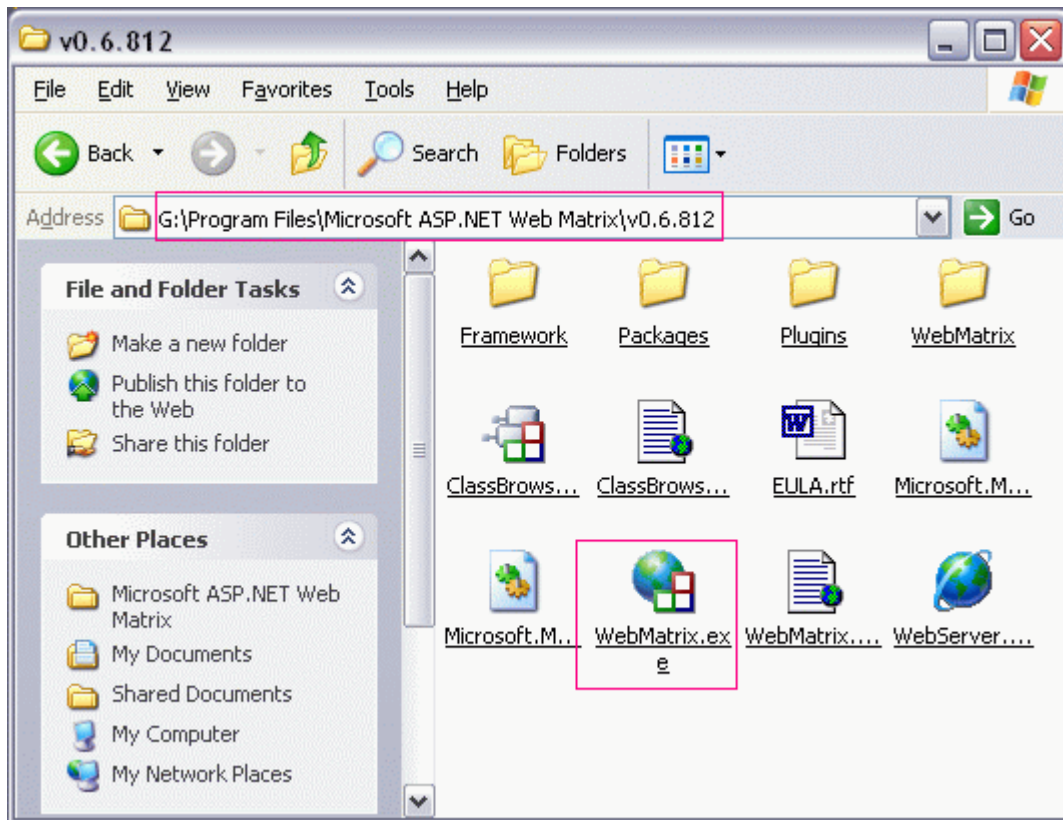
home page

<http://asp.net/webmatrix/default.aspx?tabindex=4&tabid=46>

:download

<http://asp.net/webmatrix/download.aspx?tabindex=4>

دقت داشته باشید که پس از نصب باید به دایرکتوری آن برای اجرا مراجعه نمایید (شکل ۲).

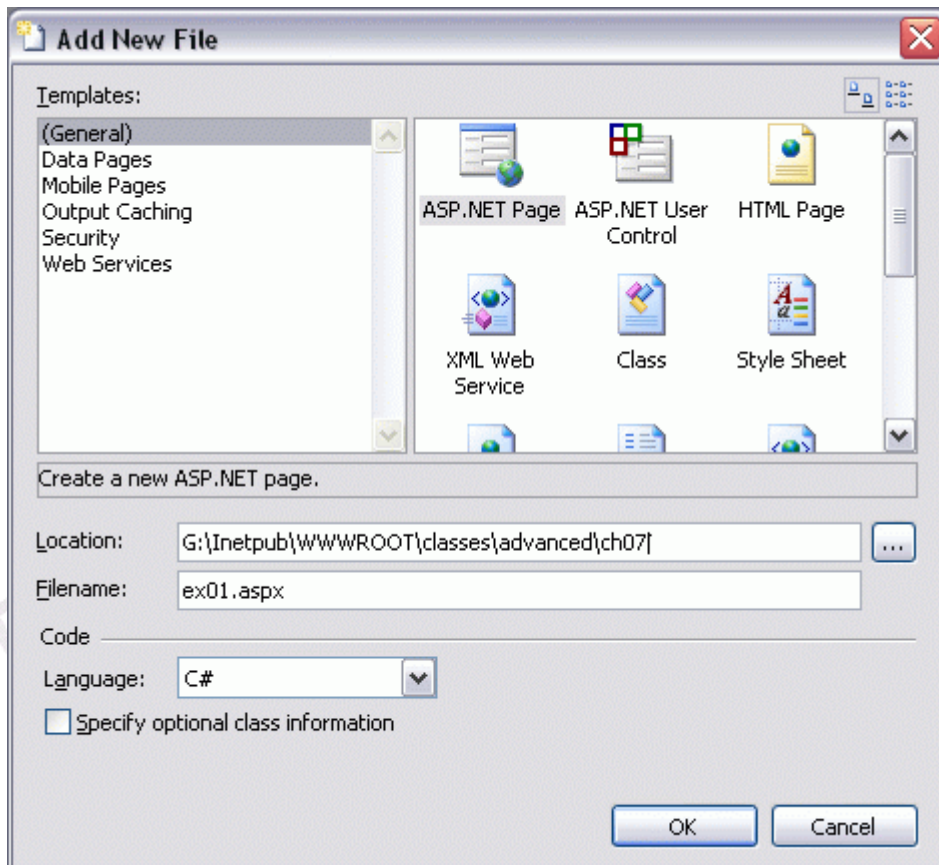


شکل ۲- در بعضی از نگارش های وب ماتریکس باید به فولدر مربوطه برای اجرای آن مراجعه کرد.

### ایجاد و اتصال به یک دیتابیس :

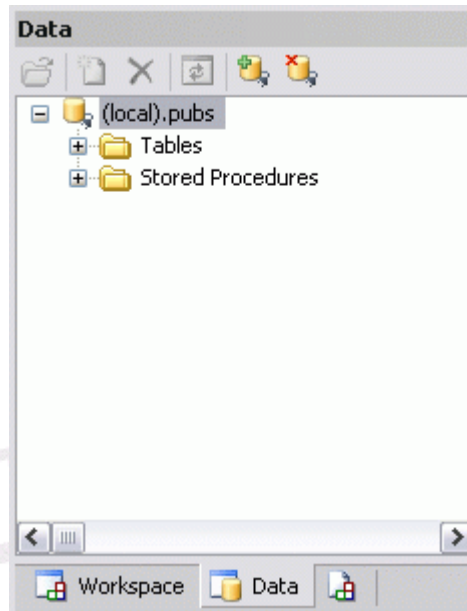
یک فایل جدید ASP.NET page را به نام ex01.aspx در وب ماتریکس ایجاد نمایید (شکل ۳). در سمت راست صفحه در قسمت workspace روی tab مربوط به data کلیک نمایید ( فرض بر این است که SQL-Server روی سیستم شما نصب است ) ( شکل ۴).

در اینجا با توجه به دکمه های روی toolbar می توان یک Connection را به دیتابیس موجود ایجاد نمود و یا کلا دیتابیس جدید را خلق کرد. اگر اتصال بر قرار شده باشد و دیتابیس pubs اس کیوال سرور را قبلا انتخاب نموده باشید ، با باز کردن treeview آن ، می توان جدول مربوطه را مشاهده نمود و یا حتی داده های آنرا ویرایش کرد (شکل ۵).

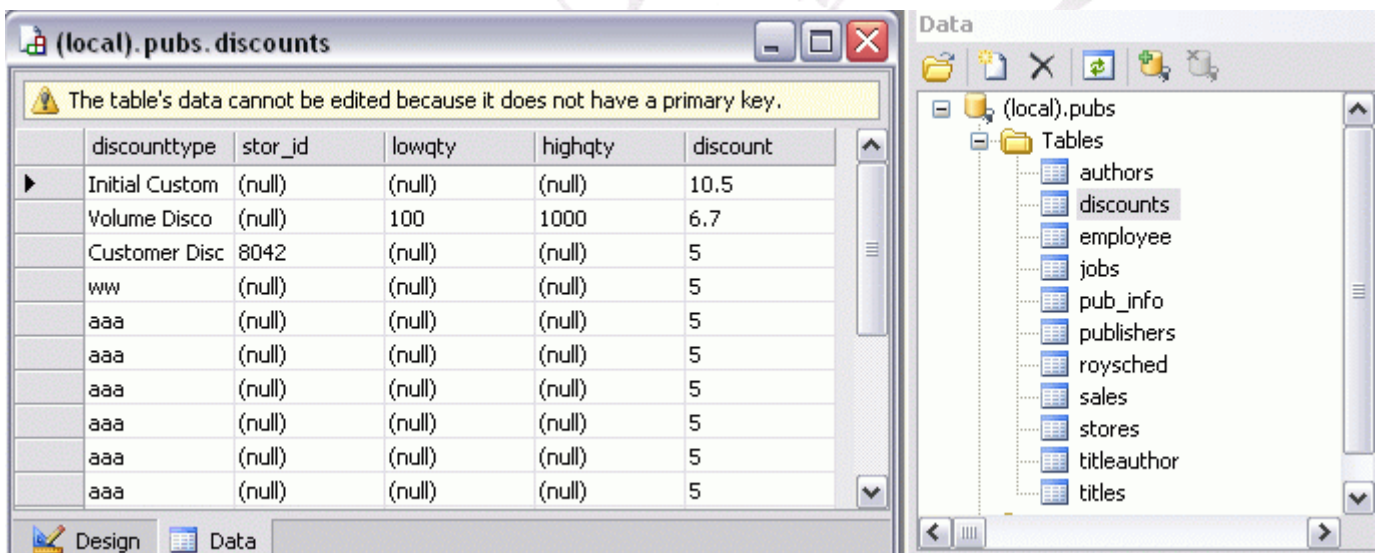


شکل ۳- صفحه ایجاد فایل‌های جدید در وب ماتریکس.

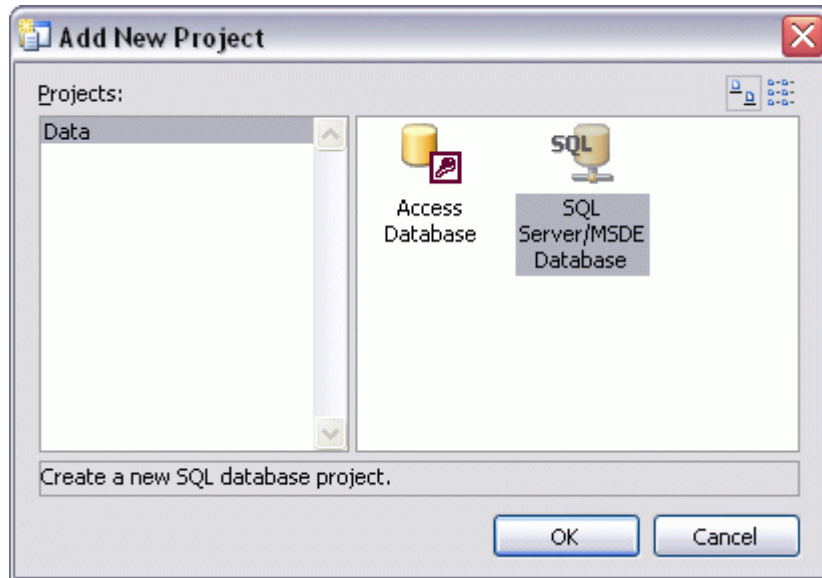
برای ایجاد یک کانکشن جدید روی دکمه ی add database connection در نوار ابزار برگه ی data کلیک نمایید تا صفحه ی Add new project ظاهر شود (شکل ۶). سپس SQL-Server/MSDE Database را انتخاب نمایید. در ادامه صفحه ی اتصال به SQL-Server را که حتما هنگام کار با Query analyzer مشاهده نموده اید ظاهر می گردد. در اینجا با وارد کردن ID و Password کاربر دیتابیس ، دیتابیس Northwind را انتخاب نمایید (شکل ۷). در این حالت این دیتابیس به لیست دیتابیس های برگه ی data اضافه می گردد. اگر نیاز به استفاده از آن در پروژه نیست می توانید روی close database connection در نوار ابزار آن کلیک نمایید (فعلا دست نگه دارید!).



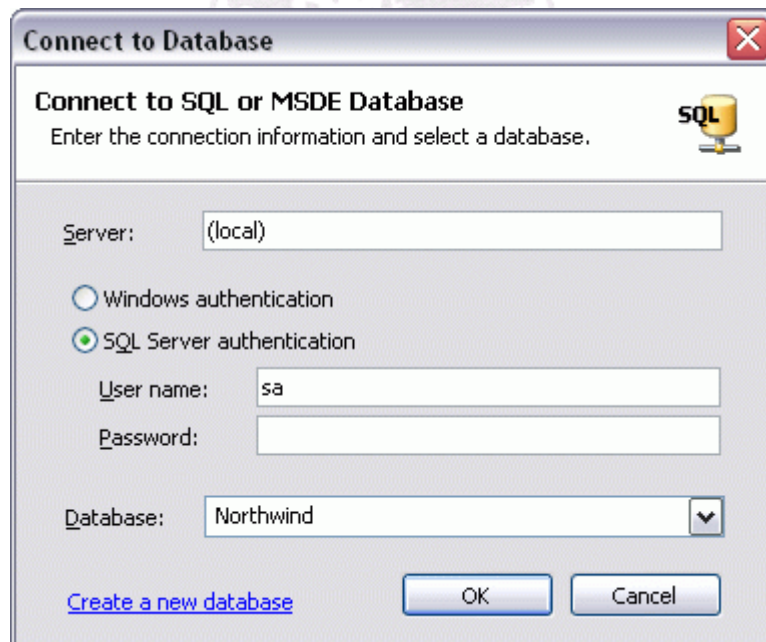
شکل ۴- برگه ی دیتا مخصوص ایجاد اتصال به دیتابیس ها.



شکل ۵- پس از برقراری ارتباط با بانک اطلاعاتی می توان به سادگی آنرا ویرایش کرد.



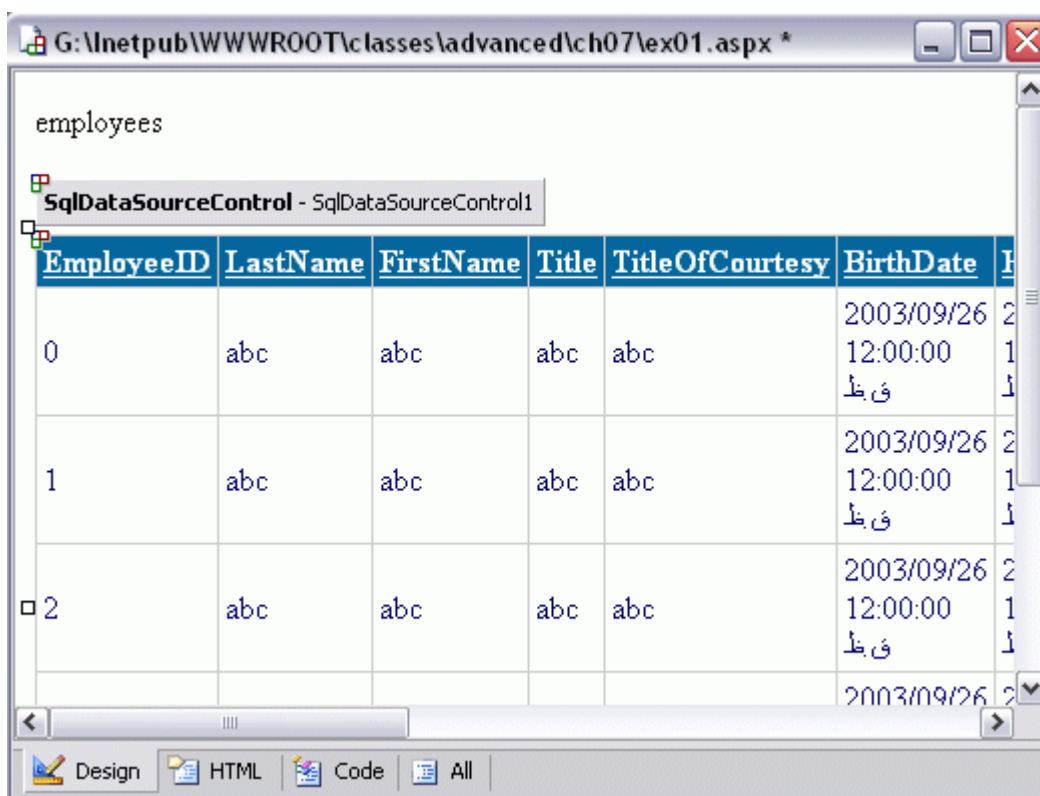
شکل ۶- انتخاب نوع اتصال به دیتابیس های مختلف.



شکل ۷- وارد کردن نام کاربر و کلمه ی عبور برای اتصال به بانک SQL-Server.

نمایش داده ها در صفحات وب :

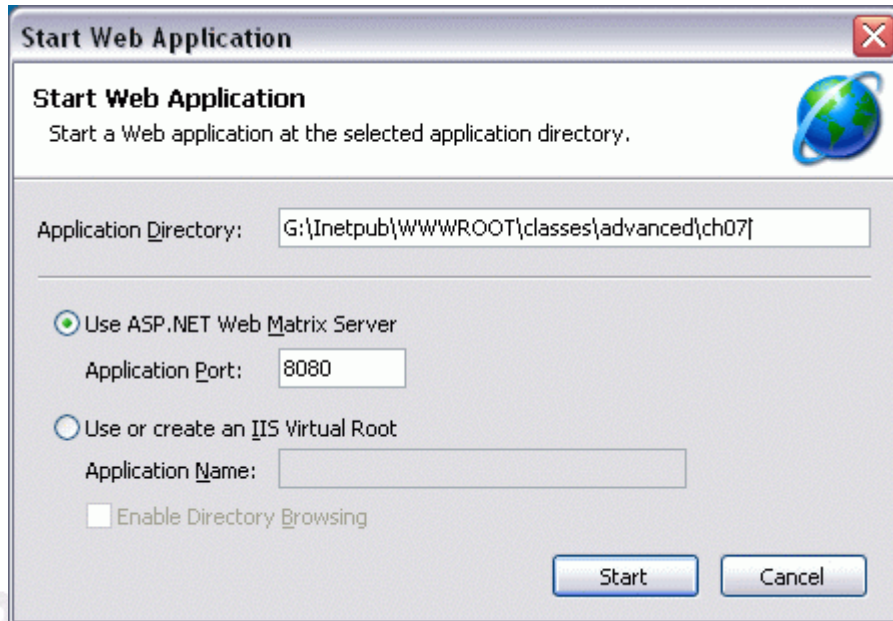
در حالیکه صفحه ی ex01.aspx را که در قسمت قبل ایجاد نموده ایم در محیط وب ماتریکس گشوده است ، از برگه ی data در workspace ، جدول employees را روی صفحه ی وب ، drag کنید. به صورت خودکار SqlDataAdapter و دیتاگرید مربوطه ایجاد می شوند (شکل ۸) .



شکل ۸- با دراگ کردن یک جدول روی فرم خودبخود دیتاگرید آن ایجاد می شود.

برای اجرای برنامه دکمه ی F5 را بفشارید. صفحه ی Start web application ظاهر خواهد شد (شکل ۹). در اینجا می توان خیلی راحت بدون ایجاد دایرکتوری مجازی با استفاده از سرور مخصوص WebMatrix که هنگام نصب آن روی کامپیوتر نصب خواهد گردید ، برنامه را اجرا کرد. روی دکمه ی start کلیک نمایید تا برنامه اجرا شود.





شکل ۹- وب ماتریکس به همراه یک وب سرور مخصوص ارائه می شود.

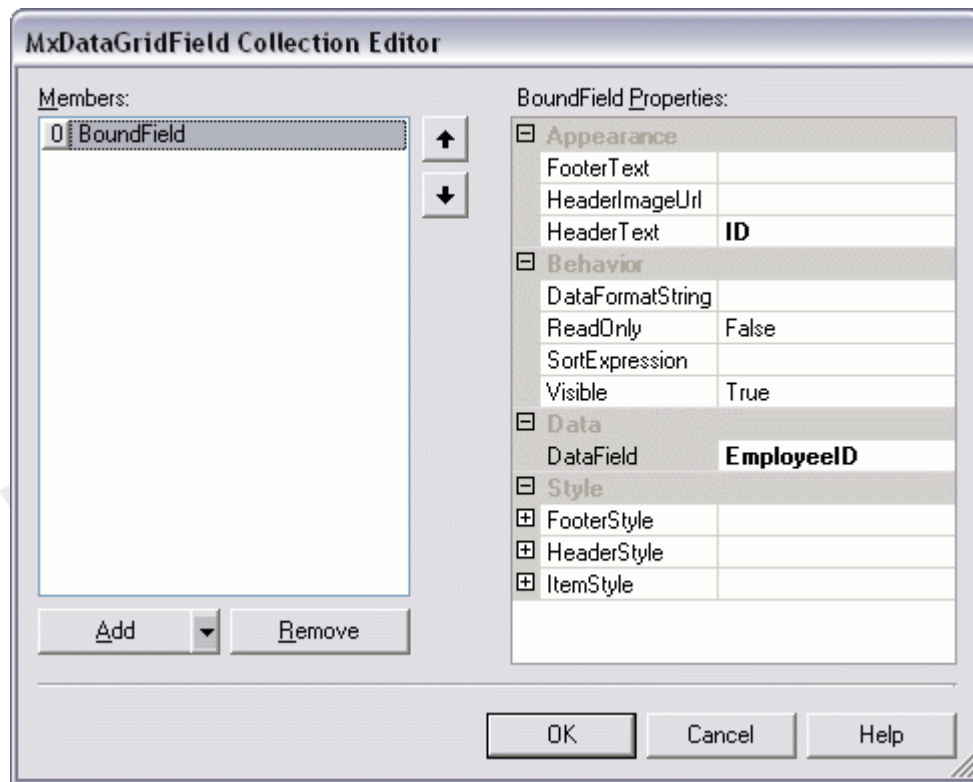
نکته :

اکثر مثالهای روی اینترنت با فرمت وب ماتریکس یعنی inline-code نوشته شده اند. اجرای آنها در این محیط فقط با یک کلیک میسر است!

فرمت کردن نحوه ی نمایش داده ها :

در برگه ی خواص MxDataGrid که به صورت خودکار برای ما ایجاد گشت ، خاصیت AutoGenerateFields آنرا false نمایید. در این حالت می خواهیم تعداد و سایر مشخصات فیلدها را خودمان کنترل نماییم. در برگه ی خواص کنترل دیتاگرید ، روی خاصیت Fields کلیک نمایید تا صفحه ی Collection editor آن ظاهر شود (شکل ۱۰). اکنون روی دکمه ی Add در آن صفحه کلیک کنید تا یک

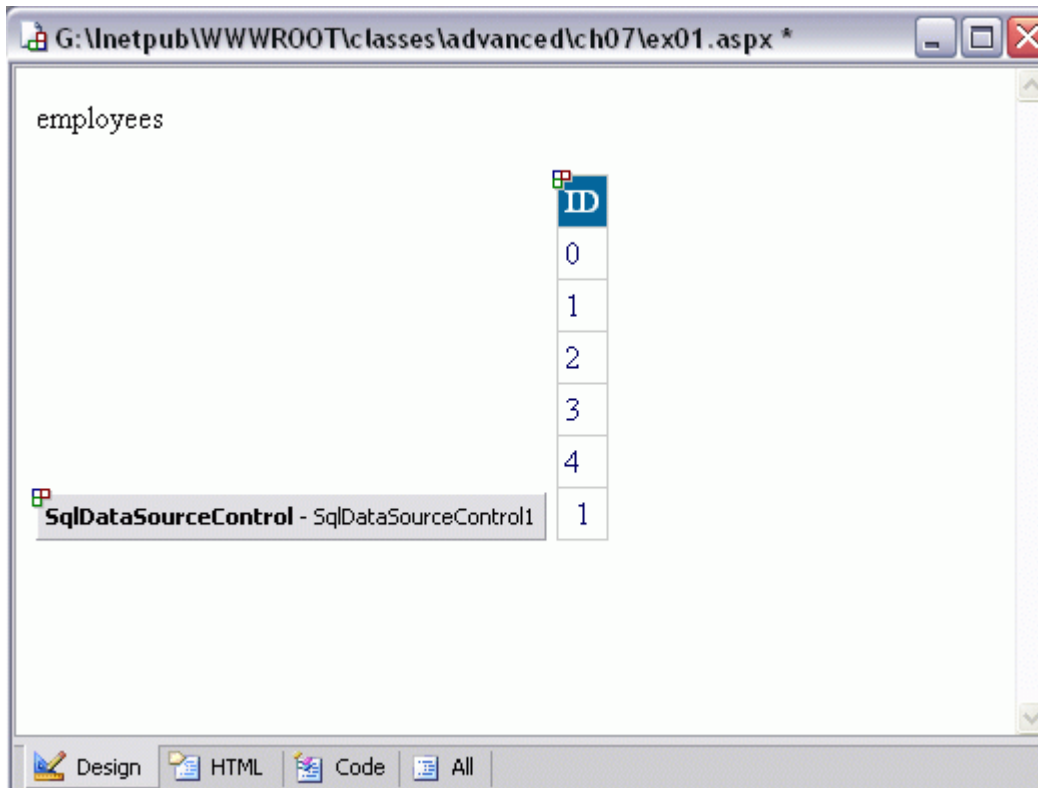
BoundField به صورت اتوماتیک ایجاد شود. HeaderText آنرا یک نام دلخواه مانند ID و DataField آنرا EmployeeID وارد نمایید.



شکل ۱۰- در اینجا به سادگی می توان ستون های سفارشی را ایجاد نمود.

اکنون روی دکمه ی Ok کلیک نمایید تا نتیجه را در صفحه ملاحظه کنید (شکل ۱۱). به این صورت هر تعداد ستون سفارشی را می توان اضافه نمود.

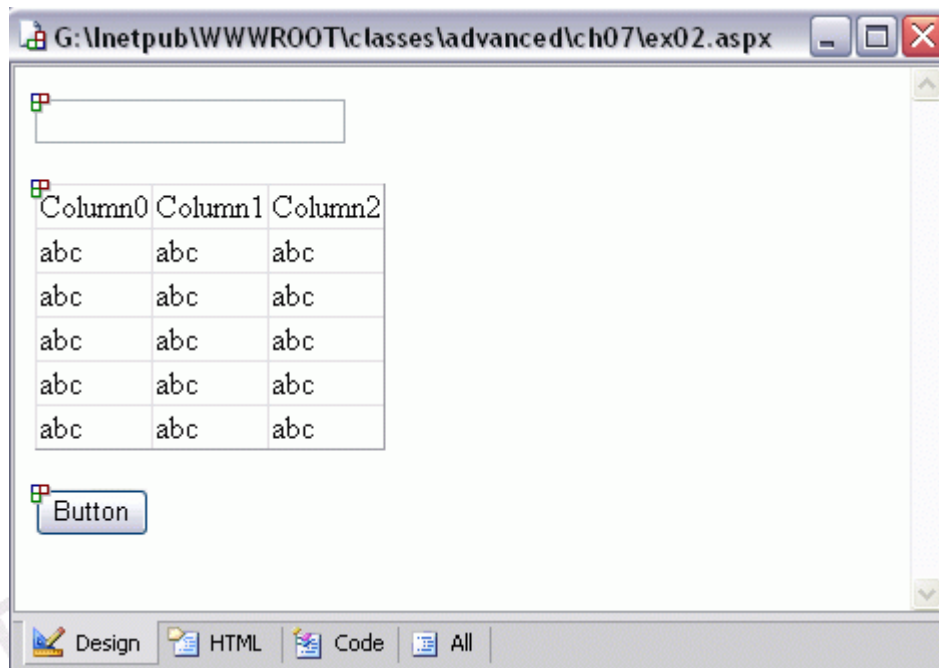
بعنوان تمرین دو ستون LastName و FirstName را هم به گرید اضافه نمایید.



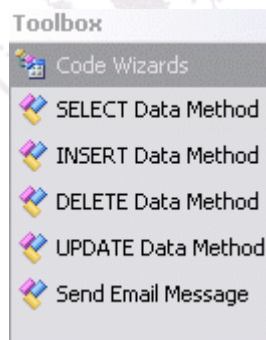
شکل ۱۱- ایجاد ستون های سفارشی .

نمایش داده های حاصل از کوئری های پیچیده ی ایجاد شده :

در قسمت قبل نحوه ی اتصال به دیتابیس را به صورت ساده ملاحظه نمودید. اما در عمل نیاز به انجام پرسجوهای پیشرفته تری روی جداول می باشد که در این قسمت به آن خواهیم پرداخت. یک فایل جدید ASP.NET page را به نام ex02.aspx ایجاد نمایید. مطابق شکل یک TextBox ، دیتاگرید و دکمه را روی فرم قرار دهید (شکل ۱۲). از منوی View گزینه ی Code را انتخاب نمایید از Code wizard ظاهر شده ، Select data method را روی فرم وب Drag نمایید (شکل ۱۳).

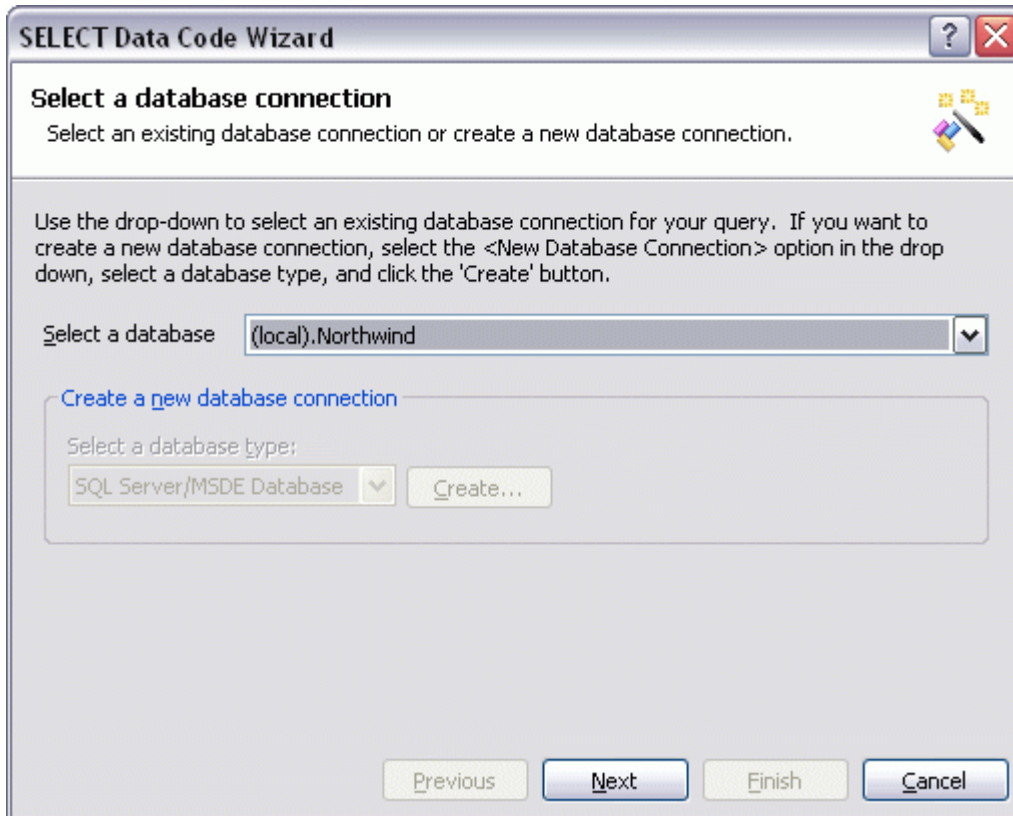


شکل ۱۲- نمایی از مثال دوم فصل در حال طراحی.



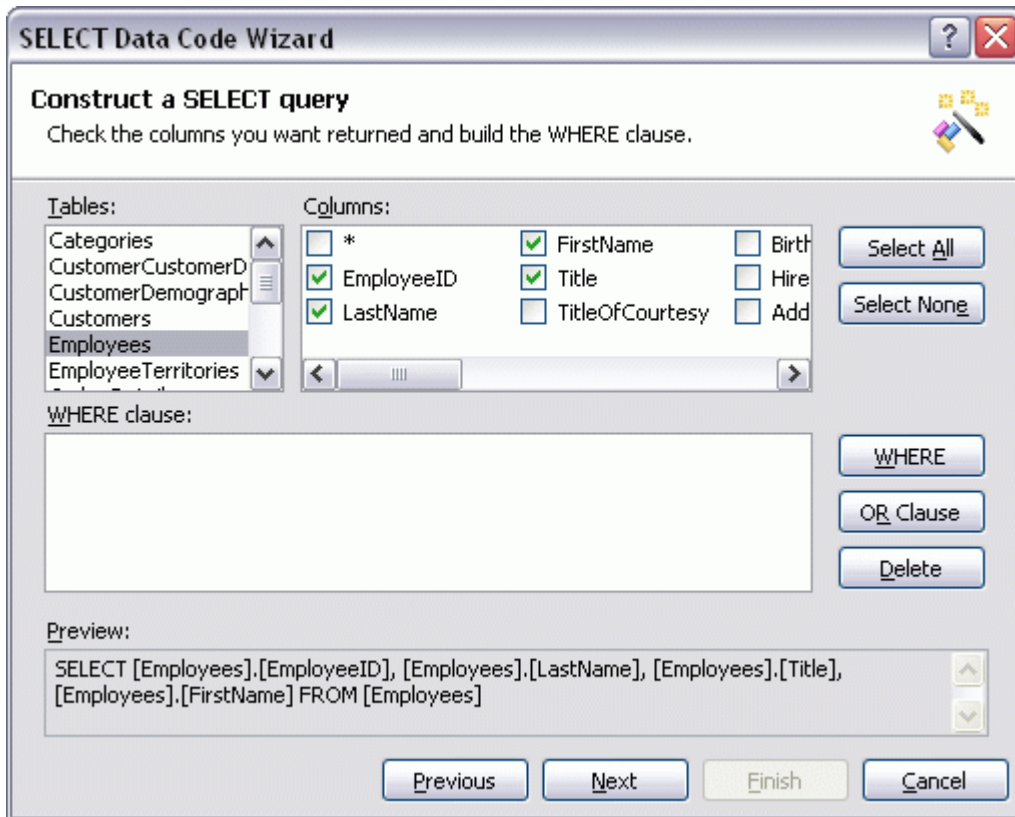
شکل ۱۳- جعبه ابزار ایجاد کوئری های مختلف برای کار با دیتابیس ها.

در صفحه ی ظاهر شده (شکل ۱۴) ، یا می توان از کانکشن هایی که قبلا ایجاد نموده ایم استفاده نمود و یا کانکشن جدیدی را به دیتابیس های موجود خلق کرد. روی Next کلیک کنید تا صفحه ی ساخت کوئری Select ظاهر شود (شکل ۱۵). در اینجا همانند محیط SQL-Server می توان کوئری های پیچیده ای را ایجاد نمود و در برنامه استفاده کرد.

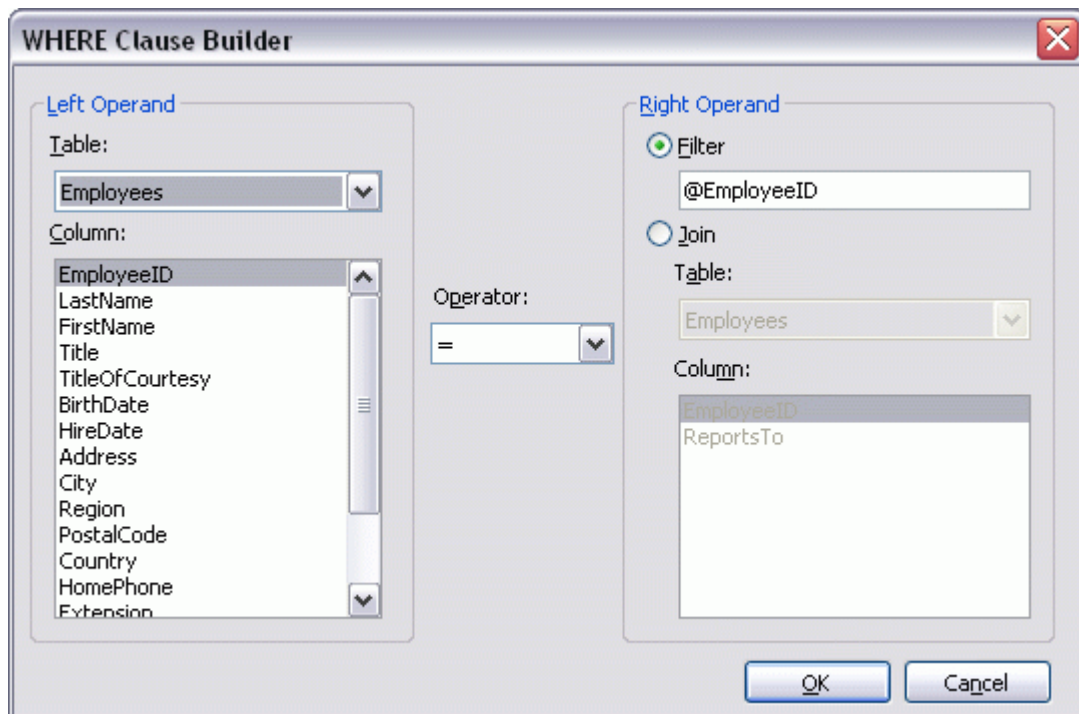


شکل ۱۴- انتخاب یک دیتابیس برای آغاز ساختن کوئری بر روی آن.

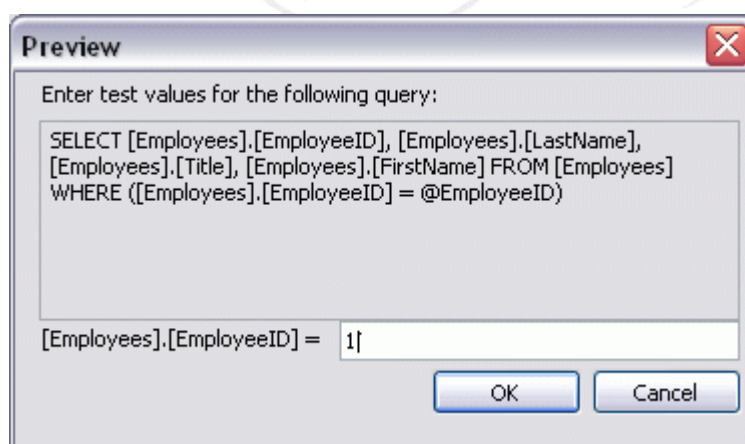
همانند شکل ۱۴، فیلدهای EmployeeID، LastName، FirstName و Title را علامت بزنید و سپس روی دکمه ی where کلیک نمایید. در اینجا می خواهیم یک عنوان را بر اساس EmployeeID پیدا کنیم بنابراین صفحه به صورت شکل ۱۶ خواهد بود. روی Ok کلیک نمایید. همانطور که ملاحظه می نمایید دکمه های AND و OR ظاهر شده اند که در صورت لزوم می توان از آنها نیز استفاده نمود. روی Next کلیک نمایید در صفحه ی ظاهر شده می توان Test query را انتخاب و حاصل را مشاهده نمود (شکل ۱۷). در صفحه ی بعدی پیش فرض ها پذیرفته و روی Finish کلیک نمایید (شکل ۱۸). به صورت خودکار کد این قسمت در محیط وب ماتریکس ایجاد می شود که صرفه جویی زمانی زیادی را پدید می آورد.



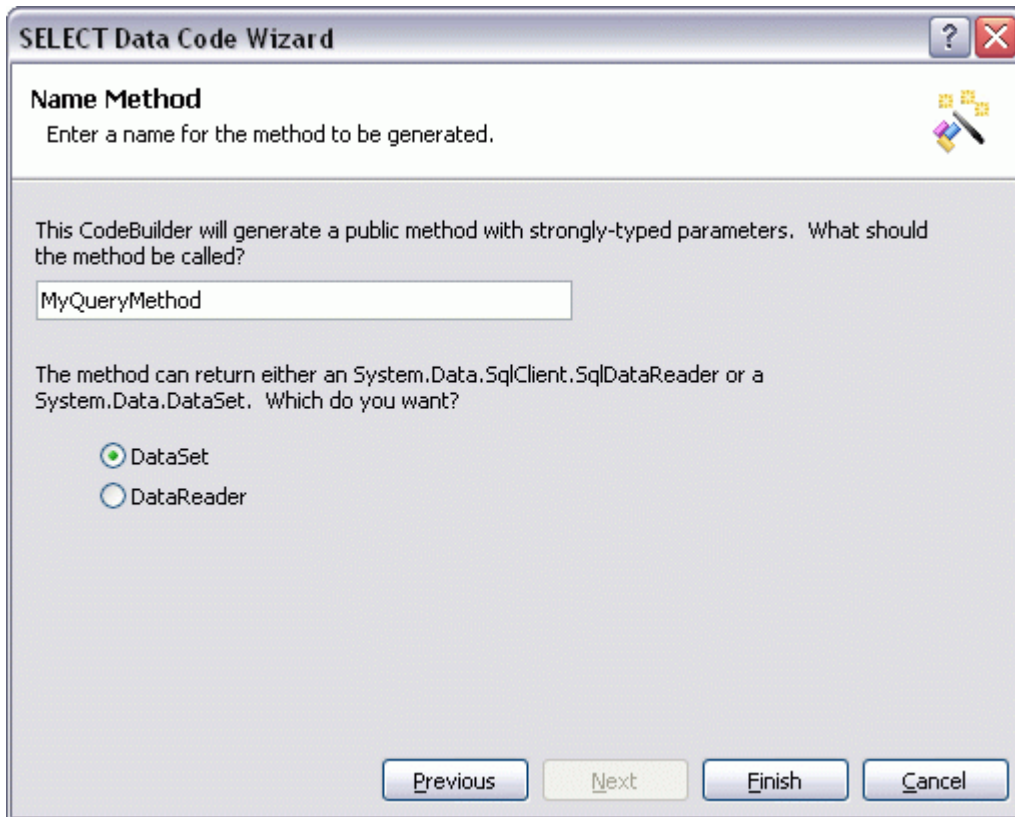
شکل ۱۵- صفحه ی ایجاد کوئری ها.



شکل ۱۶- ساختن قسمت where در عبارات SQL.



شکل ۱۷- آزمودن صحت کوئری ایجاد شده.

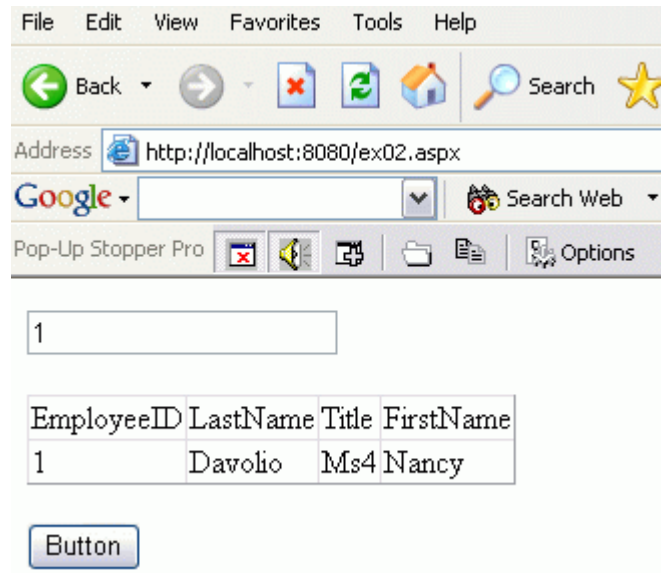


شکل ۱۸- مرحله ی پایانی ساخت یک کوئری و ایجاد متد مخصوص آن.

اکنون برگه ی Design مربوط به وب فرم را انتخاب نموده و دوبار روی دکمه کلیک نمایید تا روال رخداد آن ایجاد شود. سپس کد زیر را در آن بنویسید:

```
void Button1_Click(object sender, EventArgs e) {  
    MxDataGrid2.DataSource = MyQueryMethod(Convert.ToInt32(TextBox1.Text));  
    MxDataGrid2.DataBind();  
}
```



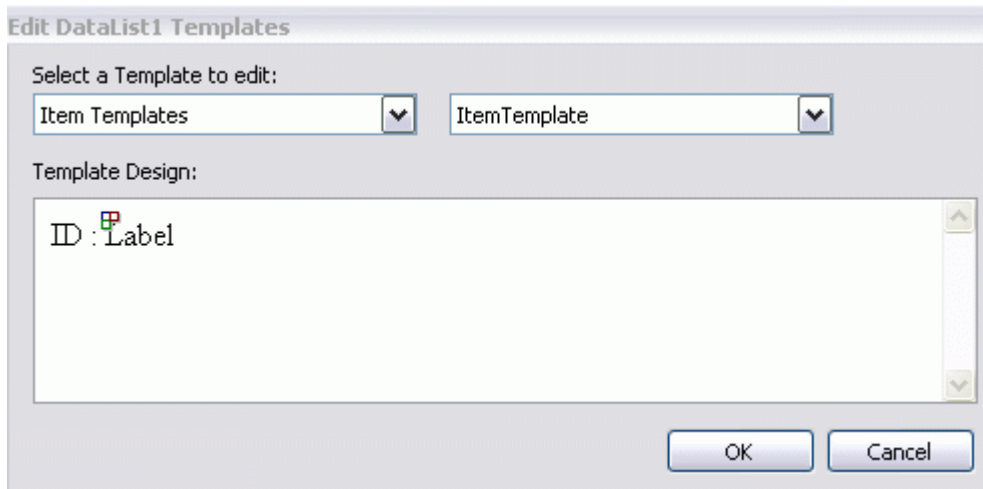


شکل ۱۹- نمایی از اجرای مثال دوم فصل.

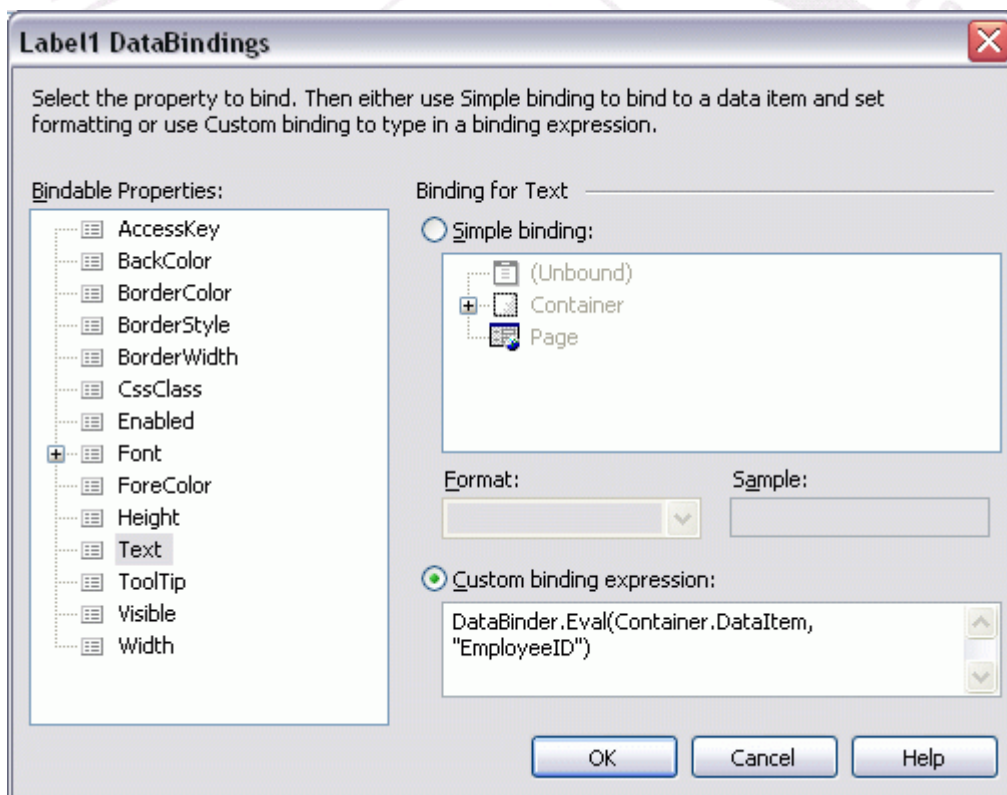
## استفاده از DataList و Templates :

یک فایل جدید ASP.NET page را به نام ex03.aspx ایجاد نمایید. یک TextBox و یک دکمه را روی فرم قرار دهید. روی دیتالیست کلیک راست نموده و سپس گزینه ی Edit templates را انتخاب نمایید. دیالوگ باکس Edit datalist templates ظاهر می شود. از منوی پایین افتادنی در هر دو قسمت Item template را برگزینید. سپس در تکست باکس این دیالوگ باکس ، ID را تایپ کرده و یک Label را روی آن Drag کنید (شکل ۲۰).

این label را انتخاب نمایید تا در پنجره ی خواص ، خواص مربوط به آن ظاهر شود. گزینه DataBinding را در پنجره ی خواص این Label انتخاب کنید تا Label data binding ظاهر شود (شکل ۲۱). از قسمت Bindable properties گزینه ی Text را انتخاب نمایید و از پنل سمت راست گزینه ی Custom dataBinding را .



شکل ۲۰- ایجاد Templates برای دیتالیست ها.

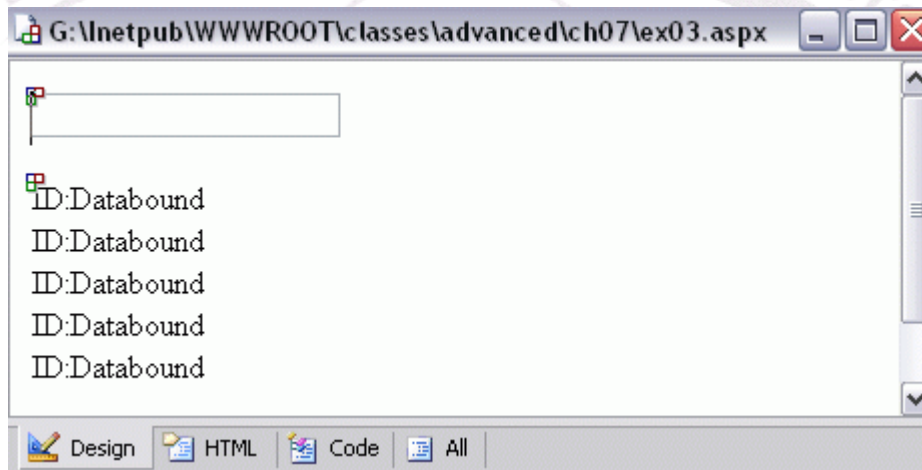


شکل ۲۱- بایندینگ سفارشی به دیتالیست.

سپس در تکست باکس آن عبارت قابل مشاهده در شکل ۲۱ را تایپ نمایید. دکمه ی Ok را فشرده و هر دو دیالوگ باکس را ببندید.  
در ادامه مانند مثال قبل همان Select data method wizard را انتخاب کرده و کوئری مناسب را ایجاد نمایید (روی جدول Employees).  
سپس روی دکمه دوبار کلیک کرده و کد زیر را بنویسید :

```
void Button1_Click(object sender, EventArgs e) {  
    DataList1.DataSource = MyQueryMethod(Convert.ToInt32(TextBox1.Text));  
    DataList1.DataBind();  
}
```

اکنون برنامه را اجرا نمایید.



شکل ۲۲- نمایی از فرم مثال سوم فصل در حالت طراحی.



تمرین :

۱- نحوه ی بایند کردن اطلاعات را به دراپ داون لیست در این محیط جدید تمرین کنید.

