

Tutorial #1: Getting Started with ASP.NET

This is the first of a series of tutorials that will teach you how to build useful, real-world websites with dynamic content in a fun and easy way, using ASP.NET and C#. ASP.NET pages can display HTML, collect user input, and interact with databases.

Note that these tutorials are not meant to provide an in-depth training in ASP.NET programming. The goal is to walk you through the development steps of a simple project, which you may later follow in designing an interface to your own final database project for this class.

If you are interested in further study of web development with ASP.NET, I recommend the book by Scott Mitchell, *SAMS Teach Yourself ASP.Net in 24 Hours*, Pearson Education, 2010, ISBN 978-0-672-33305-7.

Step 1. Installing the Software Environment

Prior to using ASP.NET, you will need to install the following software:

- SQL Server 2008 R2 for the database engine.
- SQL Server Management Studio Express for the client (SSMS)
- Visual Web Developer Express 2010 for the IDE
- The .NET framework 4 environment

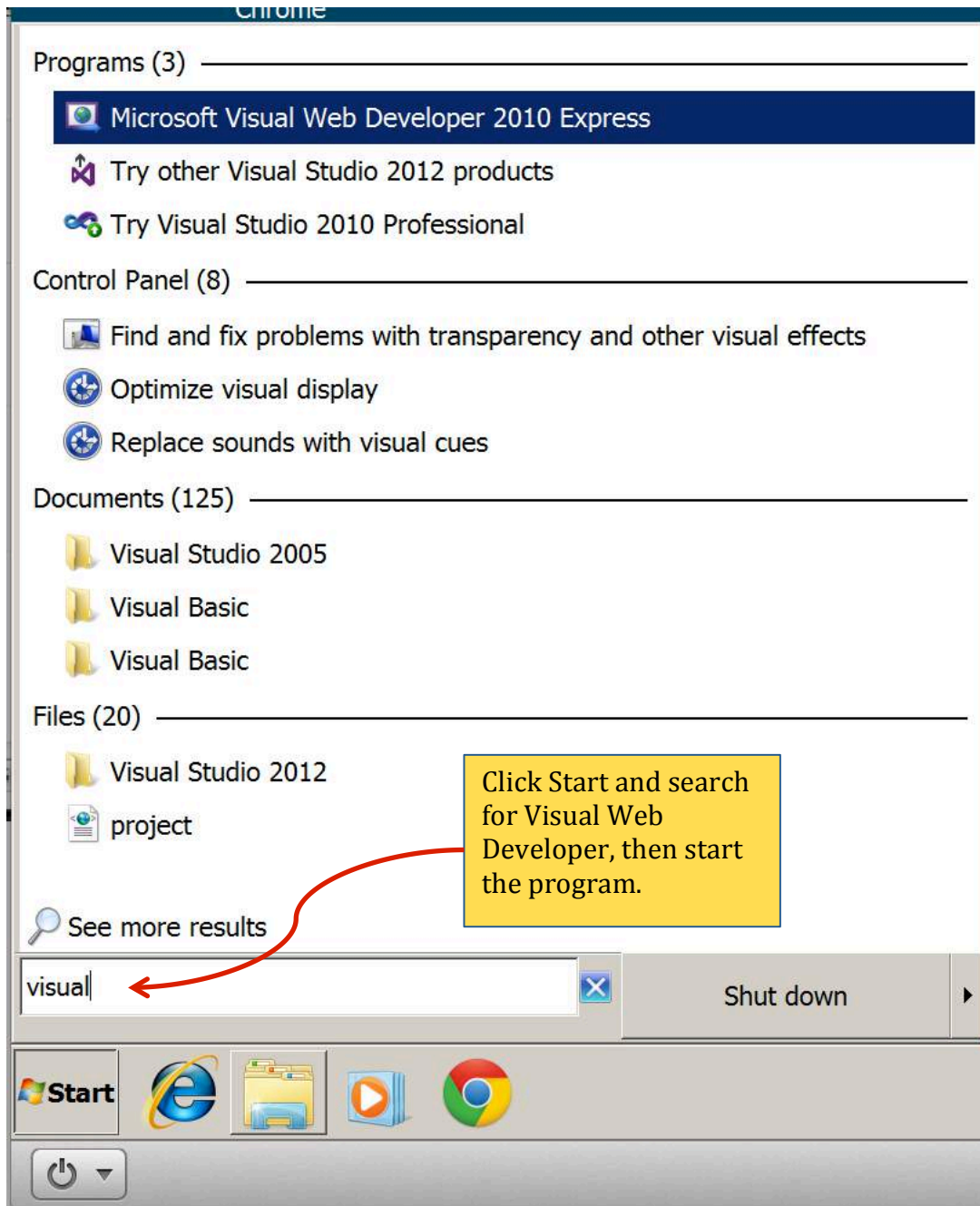
To install this software, visit the following website:

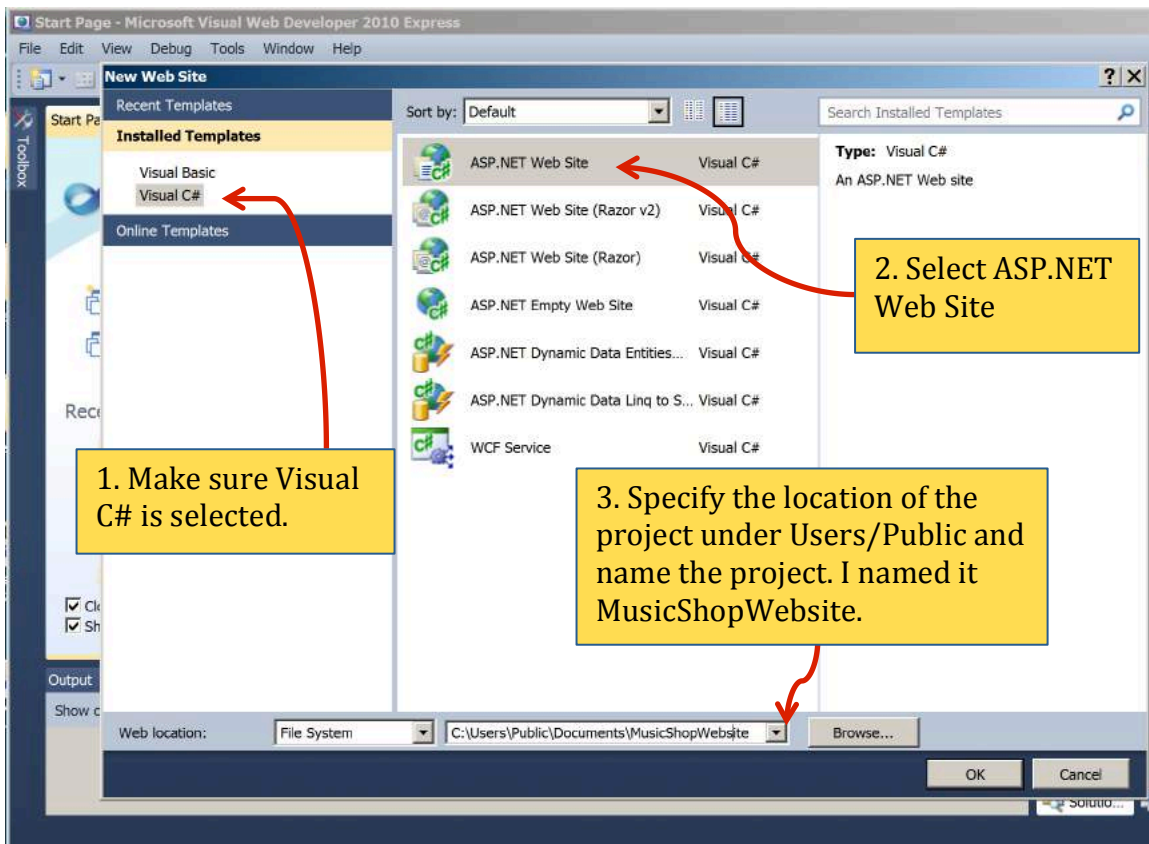
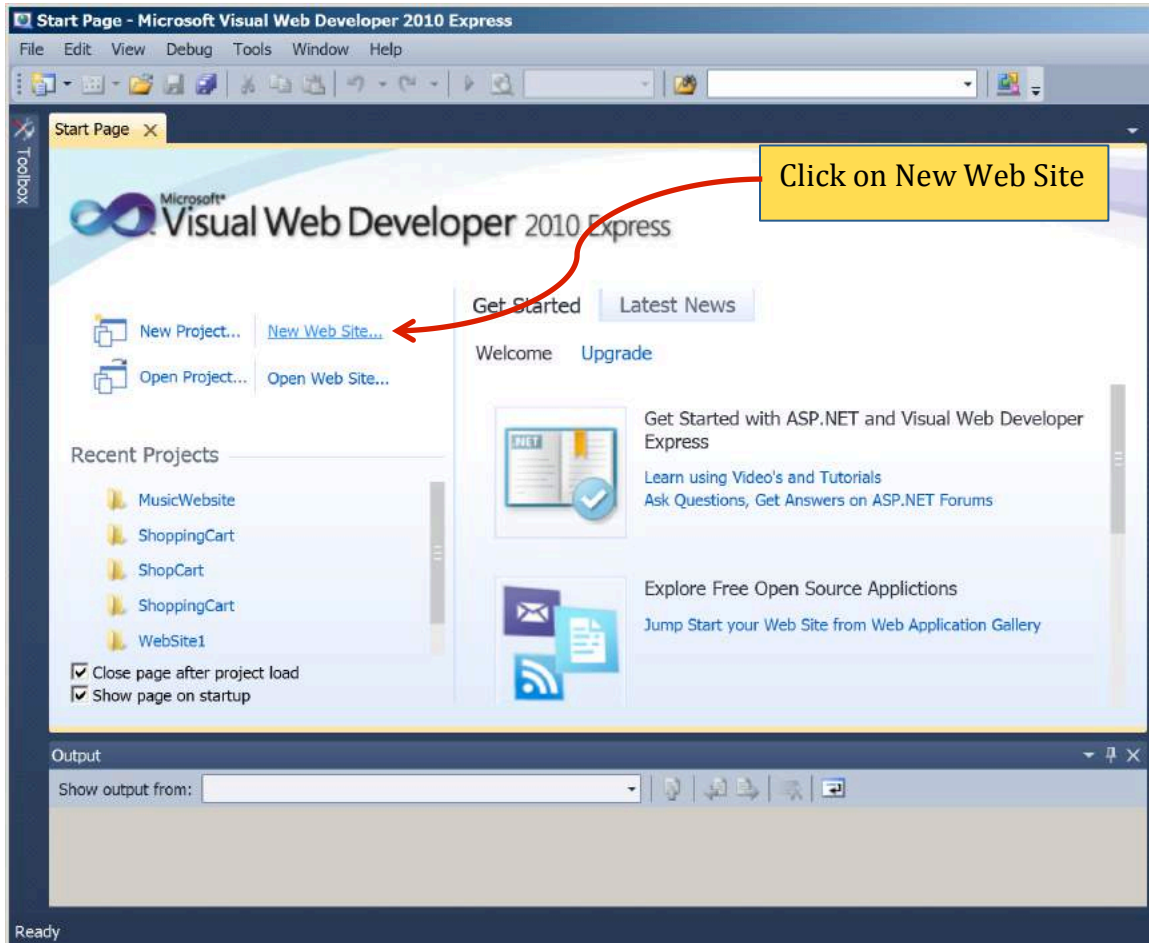
<http://www.microsoft.com/express/Downloads/>

Click on Download then scroll down to 2010 Express products and click on Visual Web Developer 2010 Express. Because Visual Web Developer is designed for developing ASP.NET websites, installing it automatically installs the .NET Framework and other required ASP.NET tools. You can also optionally install SQL Server 2008 (just checkmark “Microsoft SQL Server 2008 Express Service Pack 1” when prompted). If you forget to checkmark this option, you’ll need to install Microsoft SQL Server 2008 Express separately.

Alternately, you may download and execute the Web Platform Installer 4.6, and checkmark the 4 products listed above to be installed.

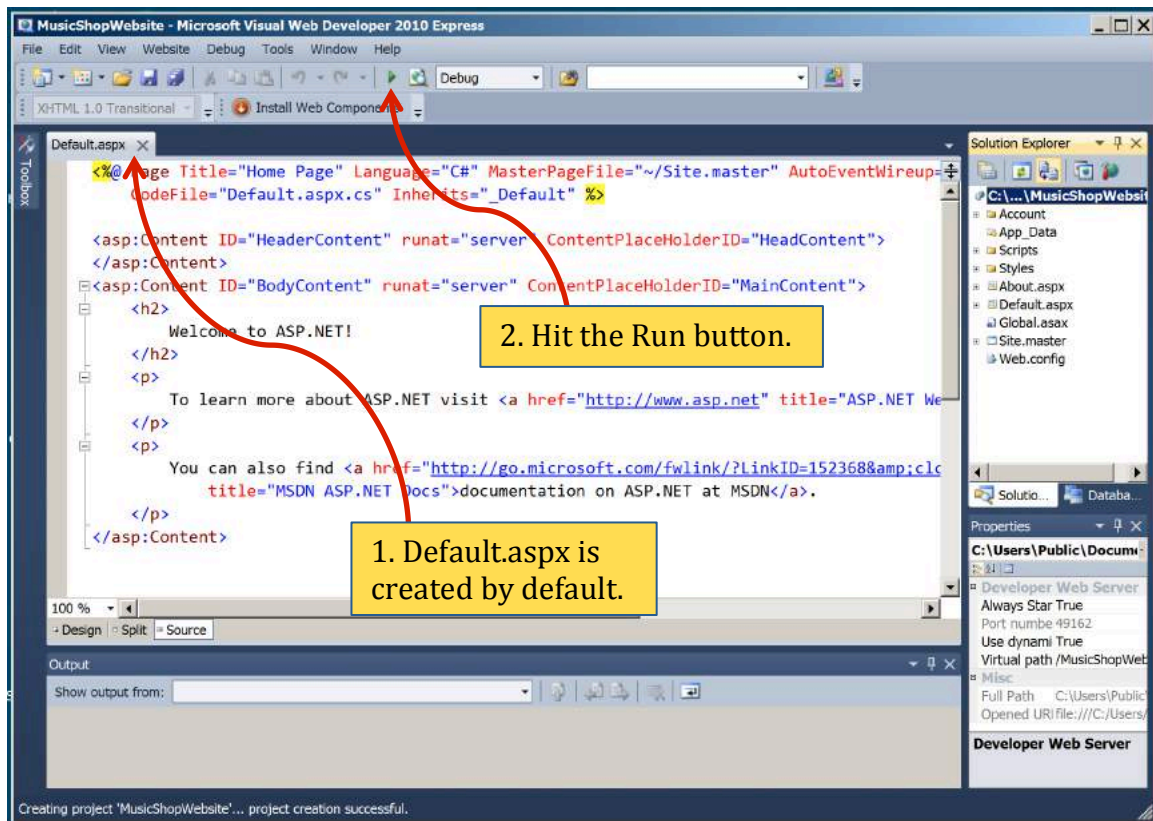
Step 2. Getting started with Visual Web Developer



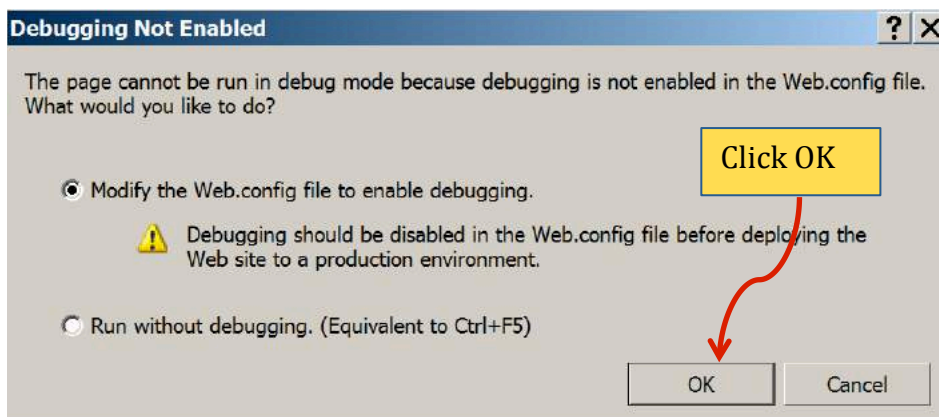


Pay attention to the directory where you store the project. The project should be in **C:\Users\Public\...** here I have named the project folder MusicShopWebsite. You might want to name it WebsiteExample, or any other name you like.

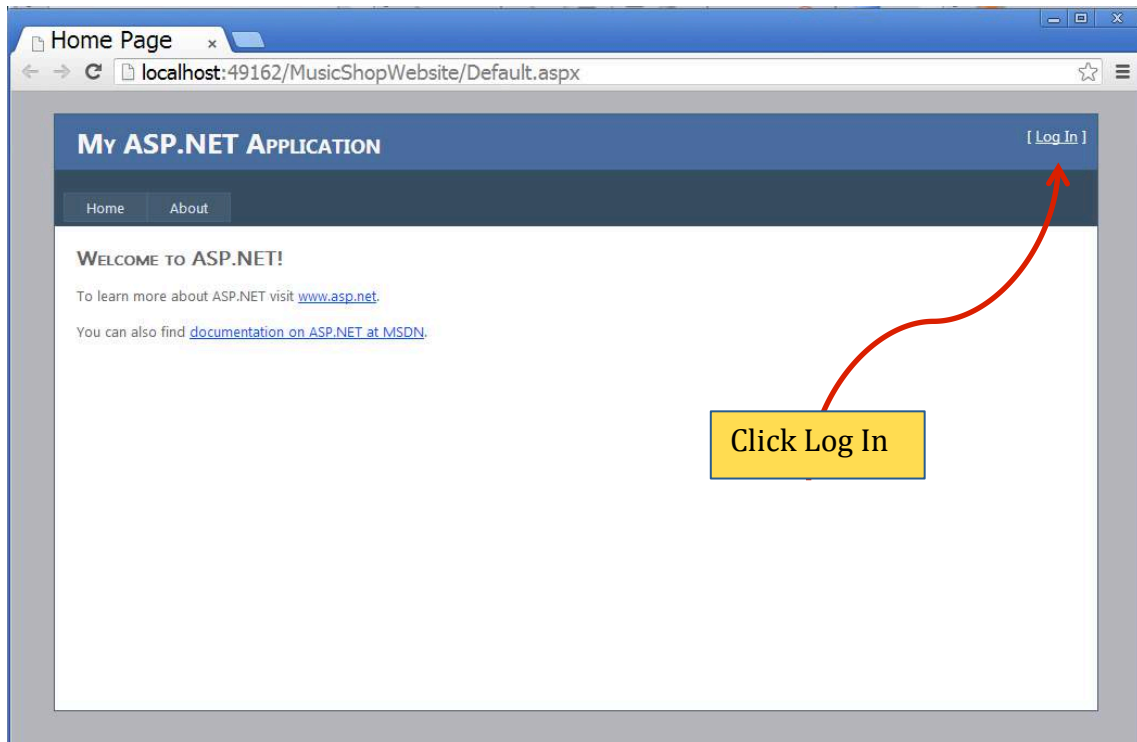
Visual web developer creates a skeleton aspx file named **Default.aspx**, which looks like the one below. This file contains only static code, but we will shortly see that we can *Web controls* to this page to support dynamic content.



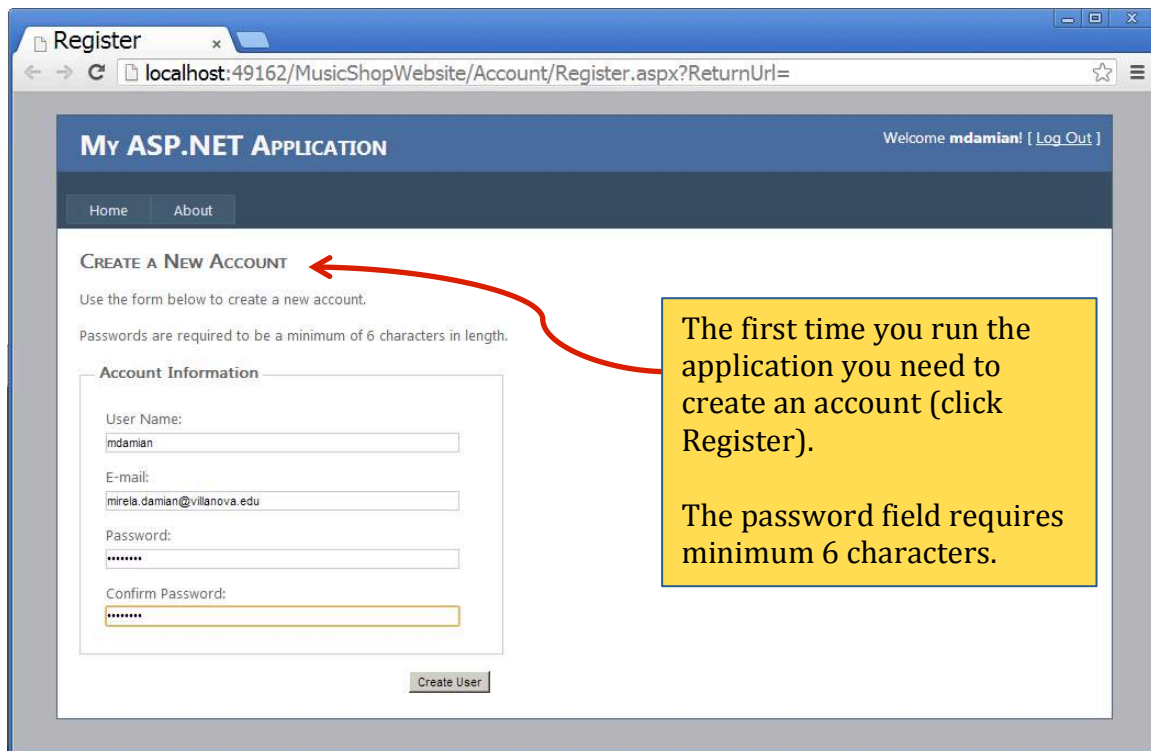
With Default.aspx open and your project folder in the Solution Explorer highlighted, hit the green triangular button at the top toolbar or F5. You may get a window like the one below. Go ahead and click OK.

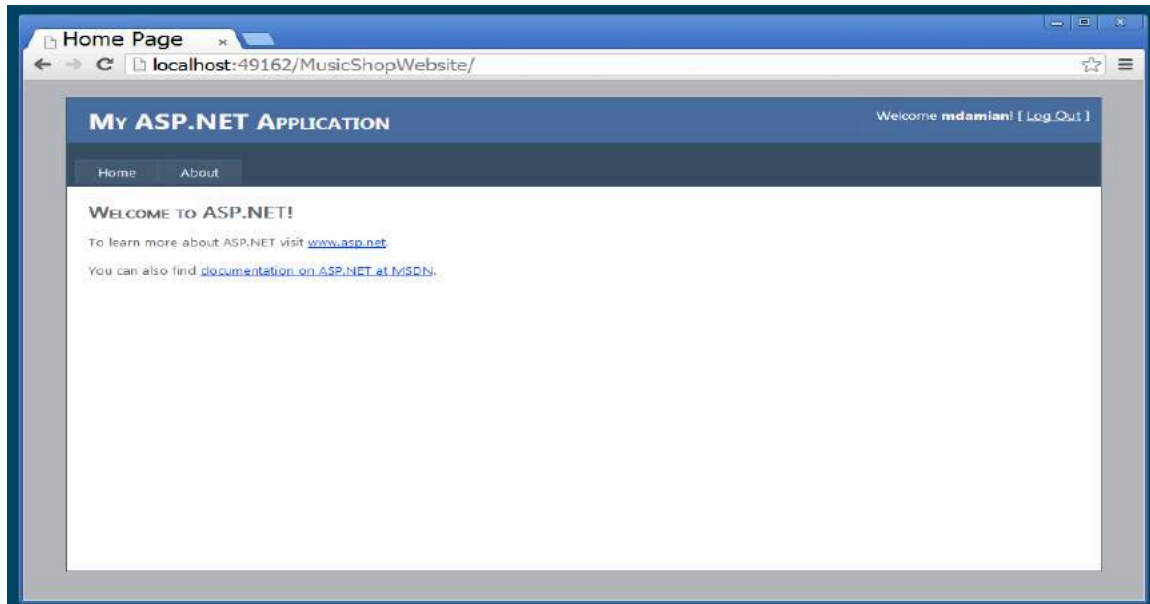


The default system browser opens and displays the application running.



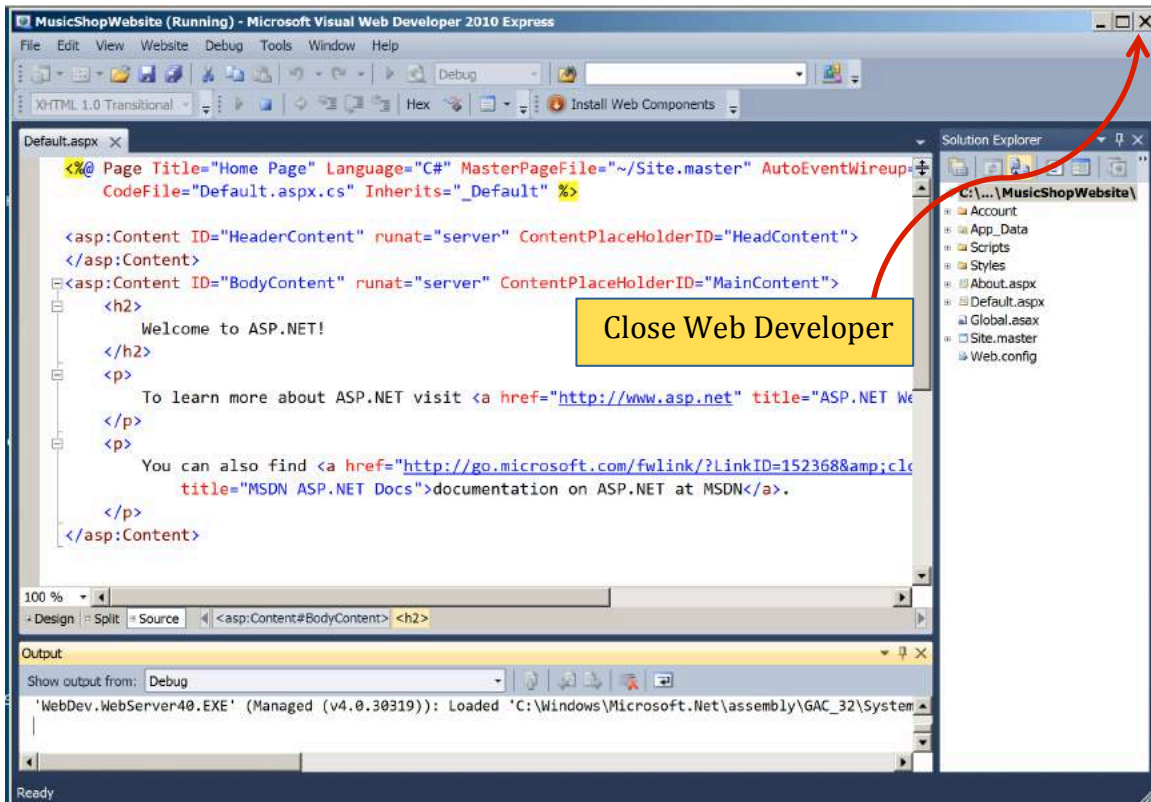
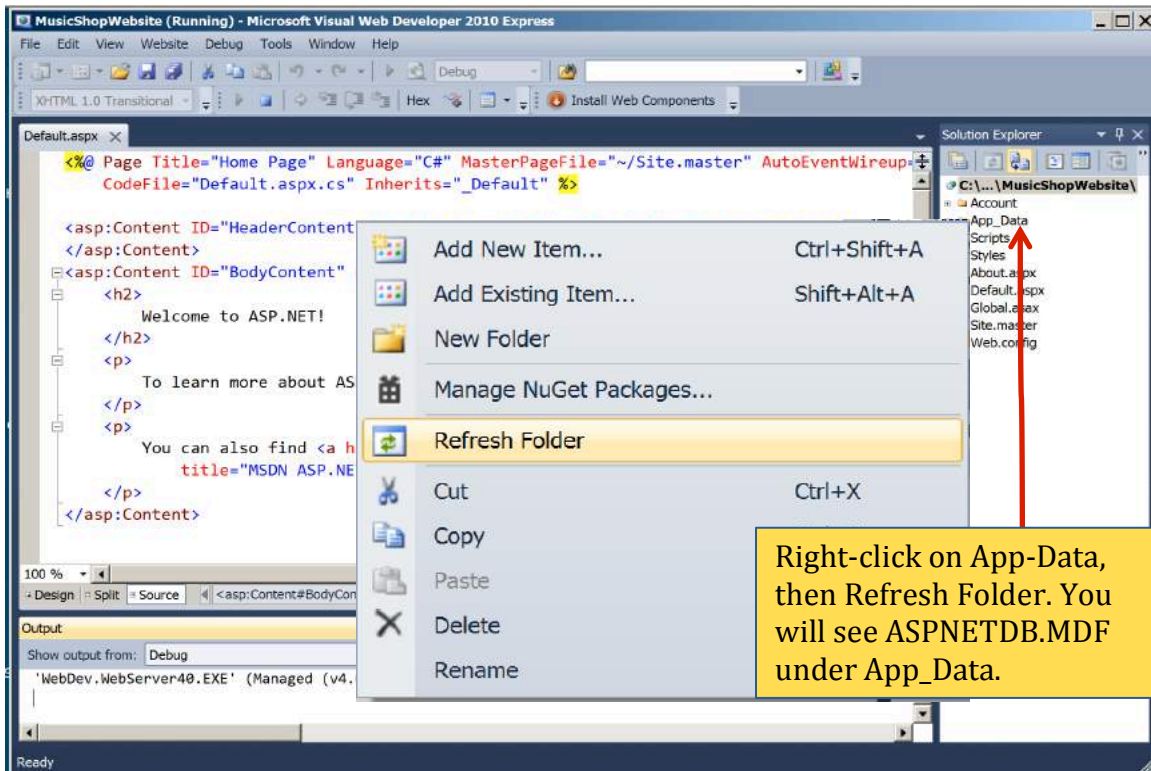
Many web applications require users to create an account before they can use certain features of the site. Visual Web Developer automatically includes Login and Register controls. Try logging in or create a new user.





Close your browser and return to Visual Web Developer. For the first user registered with the website, Visual Web developer automatically creates a SQL Server 2008 Express Edition database named **ASPNETDB** in the **App_Data** folder with predesigned tables to support user accounts.

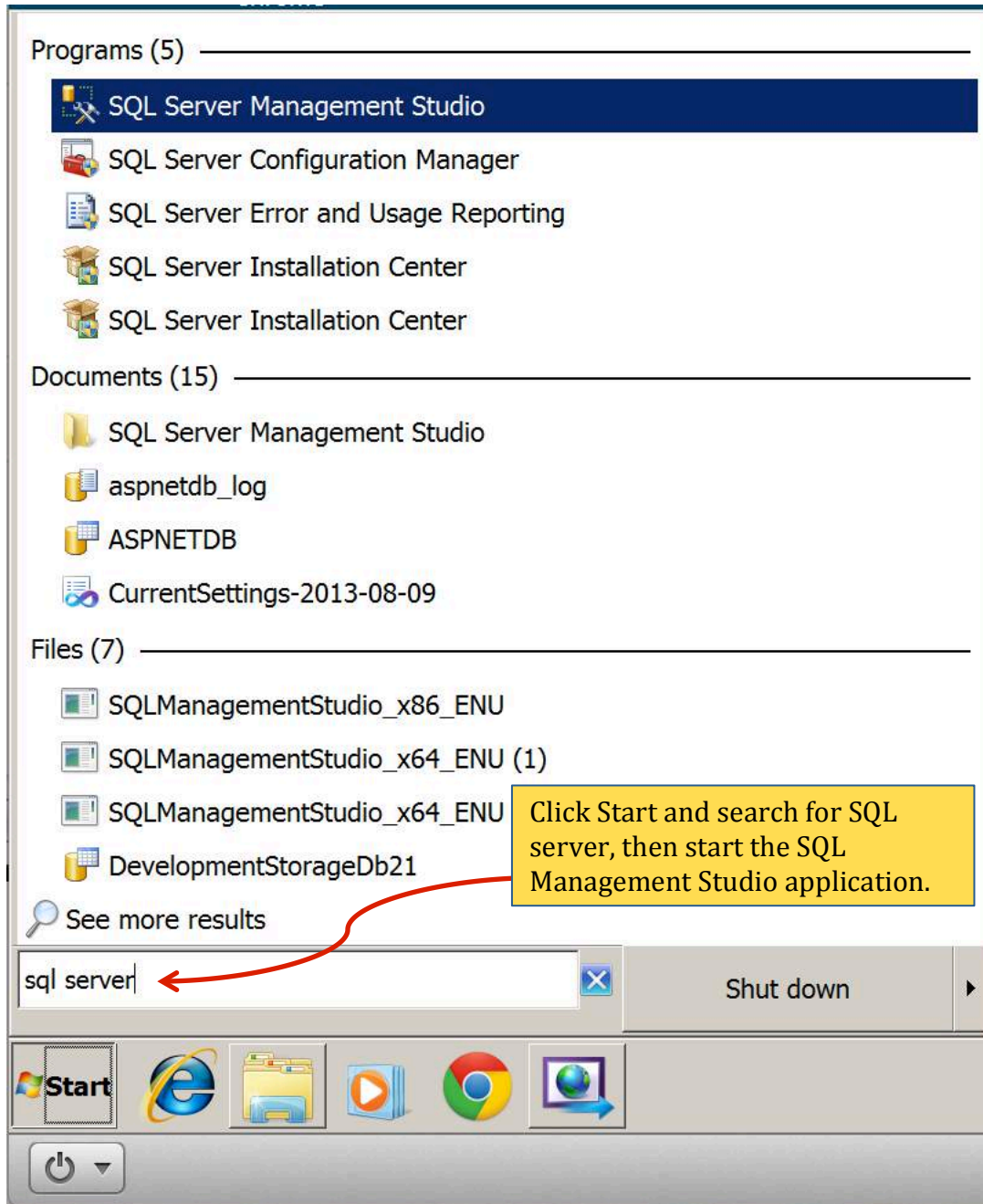
To verify this, right-click on the App_Data folder in the Solution Explorer window and select **Refresh Folder**. Expand App_Data by clicking on the plus sign. You will see the database file ASPNETDB.MDF used by your website. By default it includes 11 tables, providing functionality that extends beyond simple user accounts.

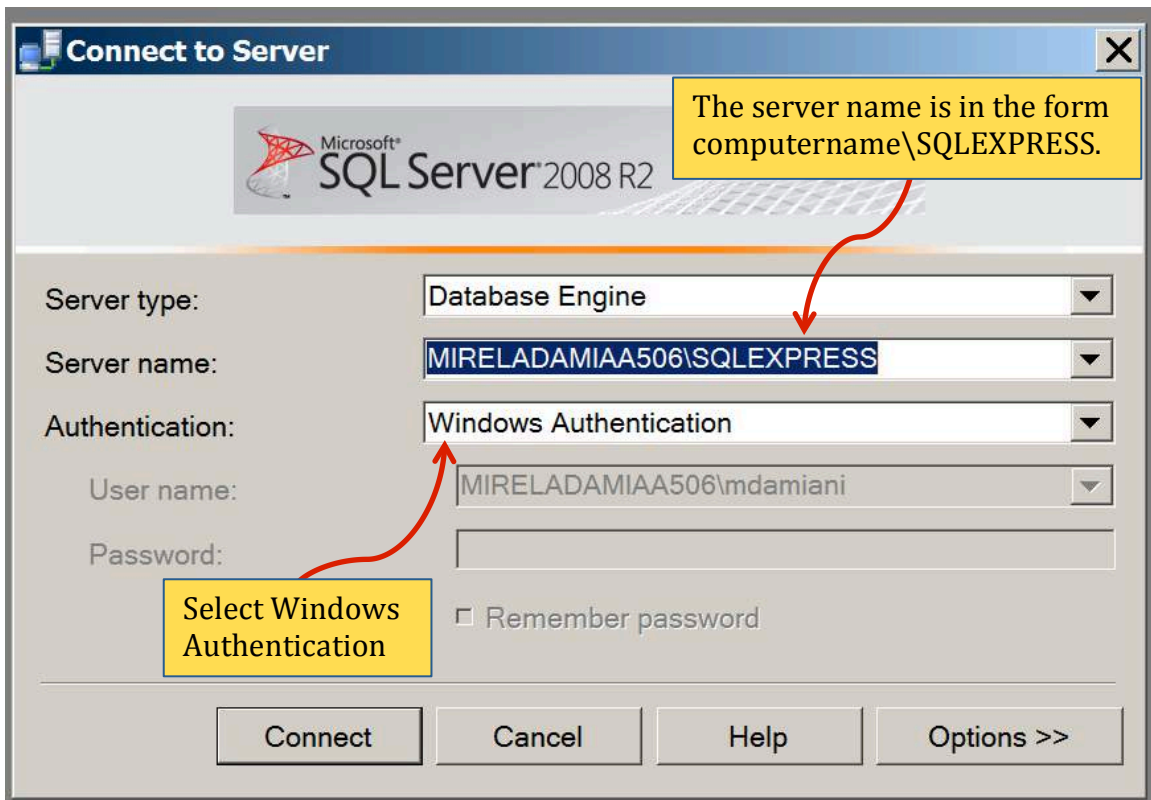


Now that you are a new user of this application you are ready to create the tables in the database. Save all changes and make sure you close Visual Web Developer.

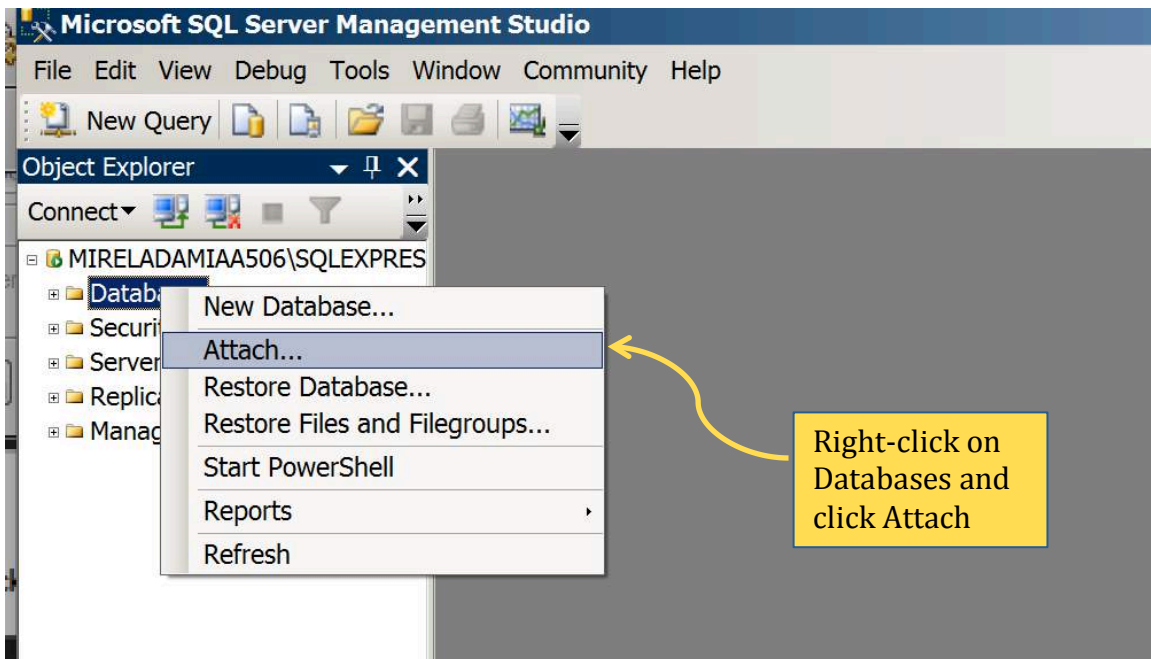
Step 3. Adding Tables to Your Database

After closing the Visual Web Developer, open SQL Server Management Studio.

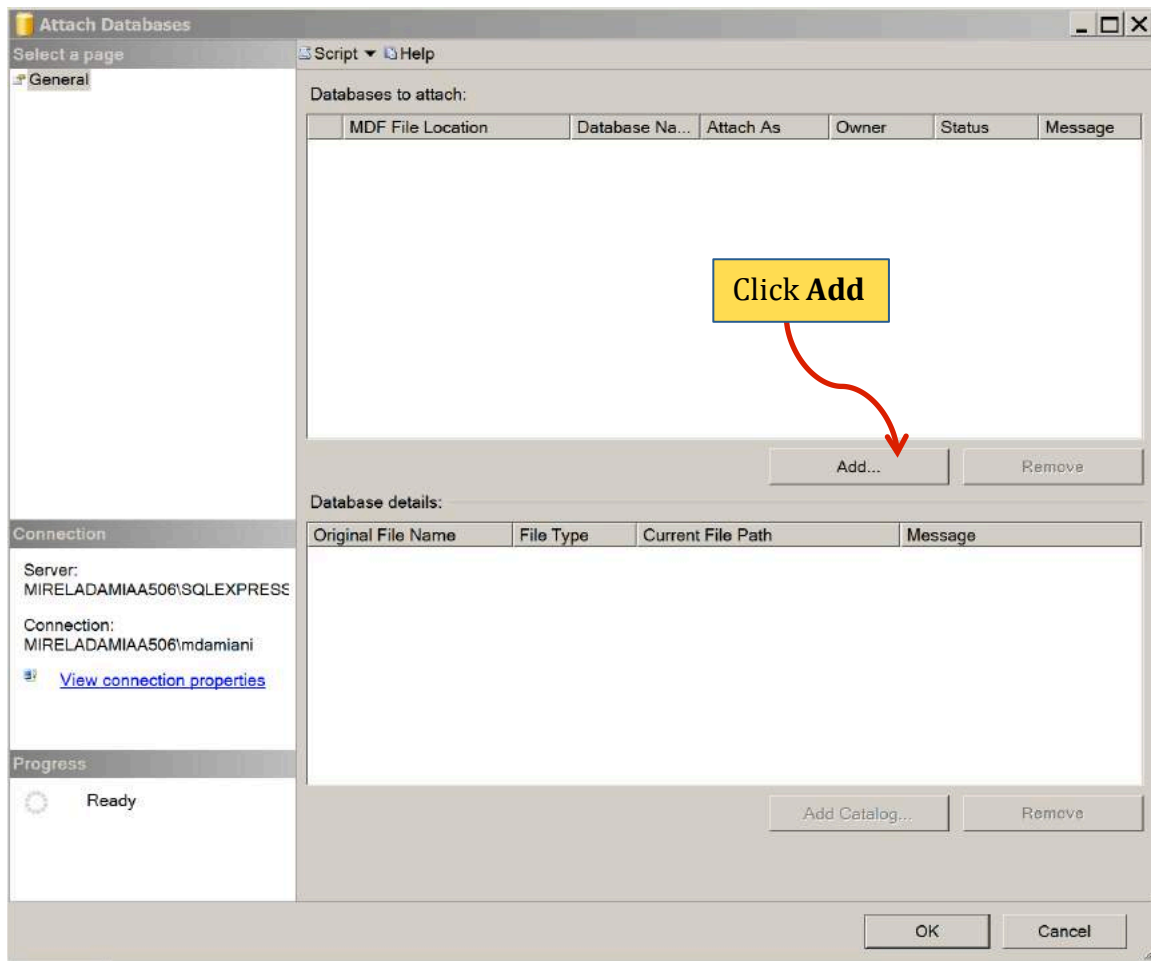




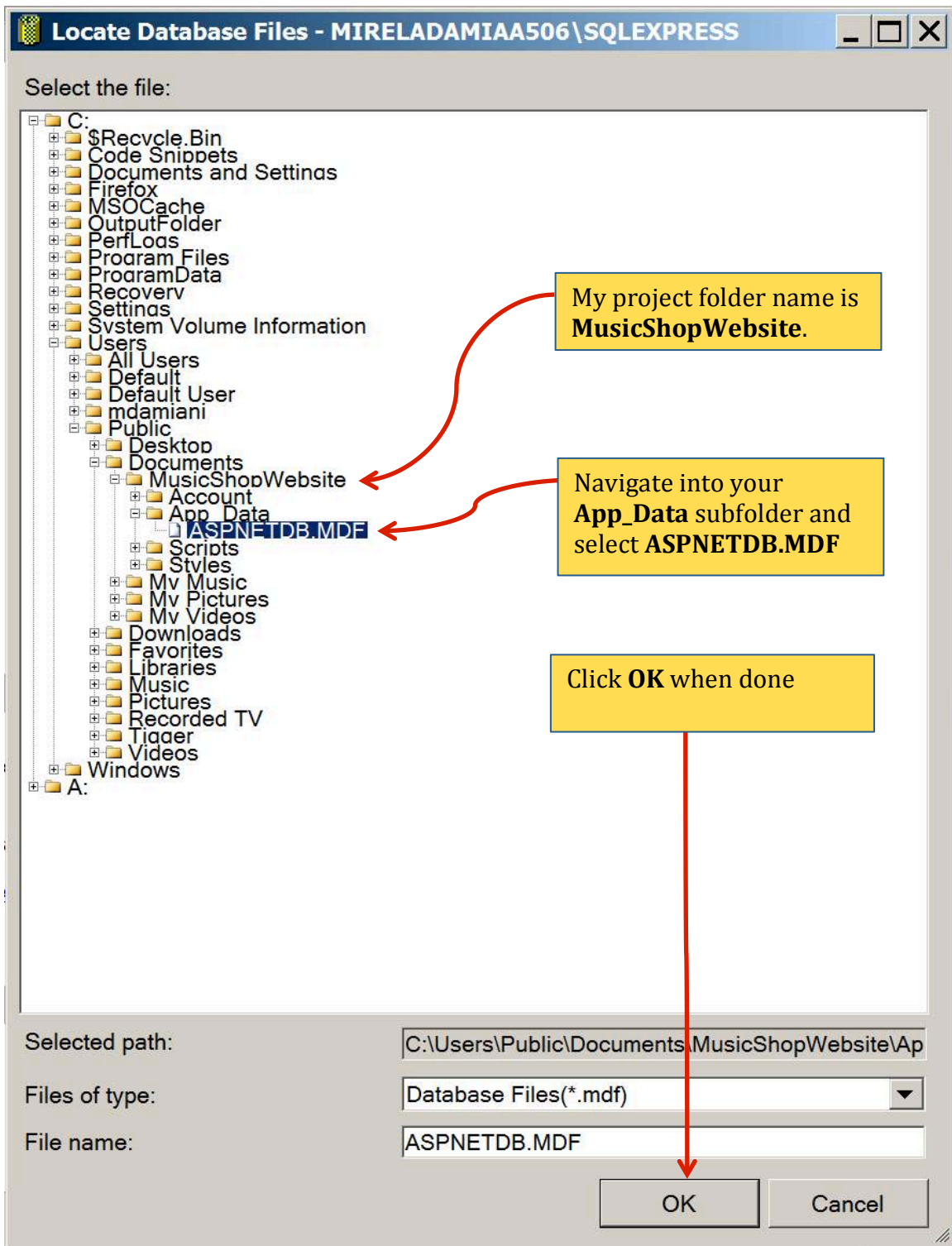
Once you are connected right click on **Databases** and click **Attach**.

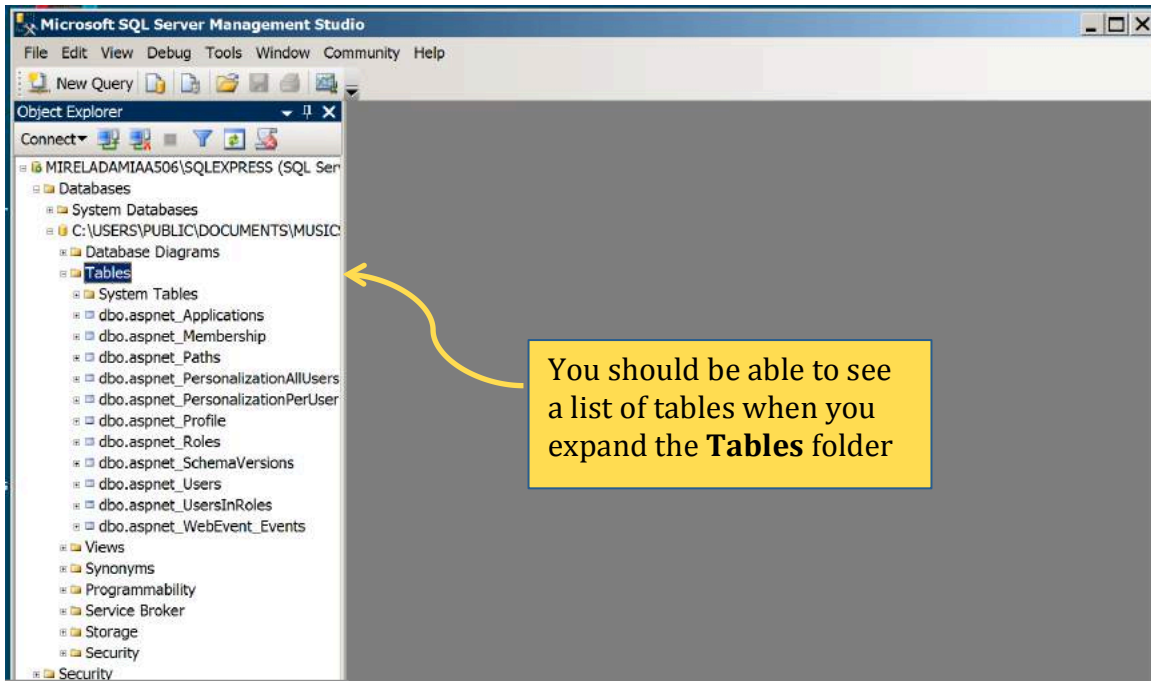
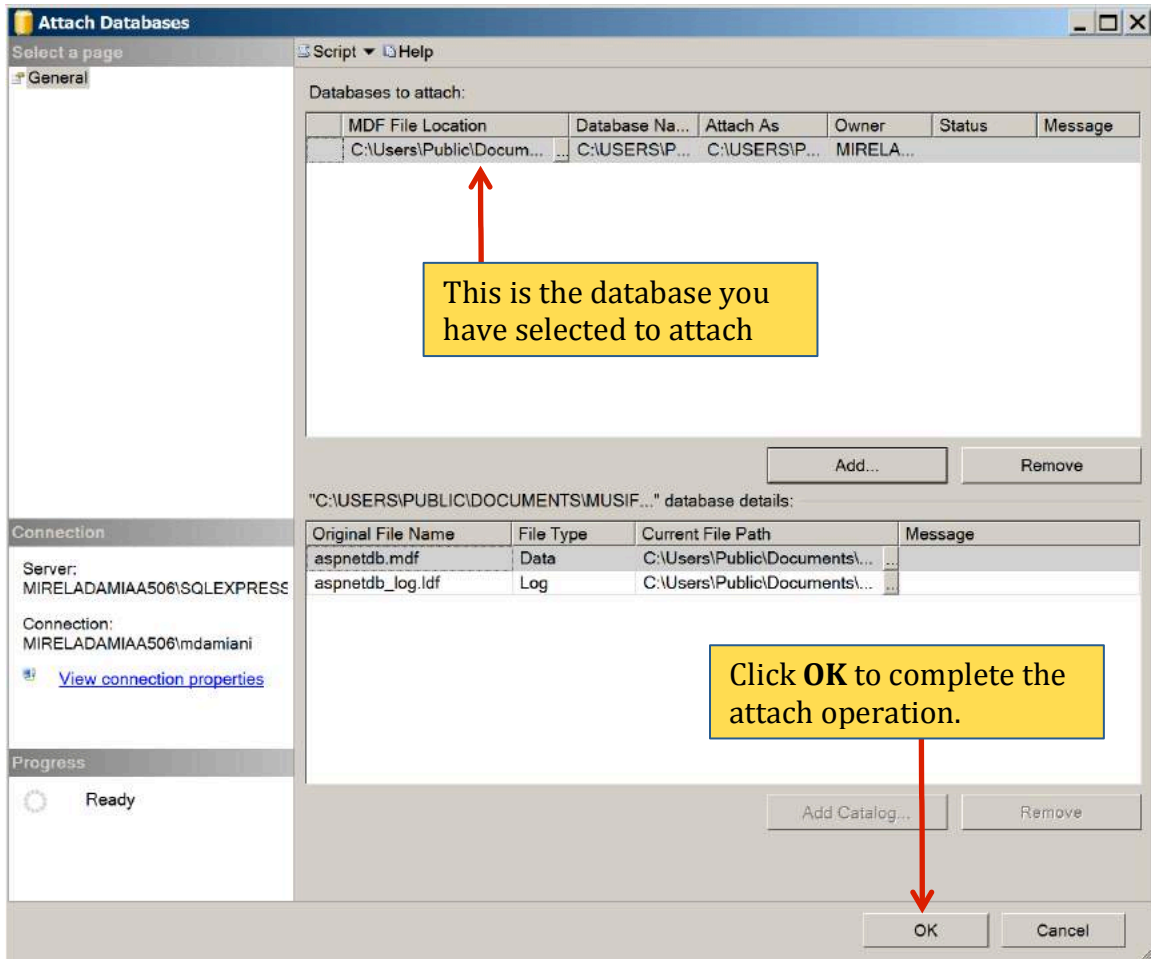


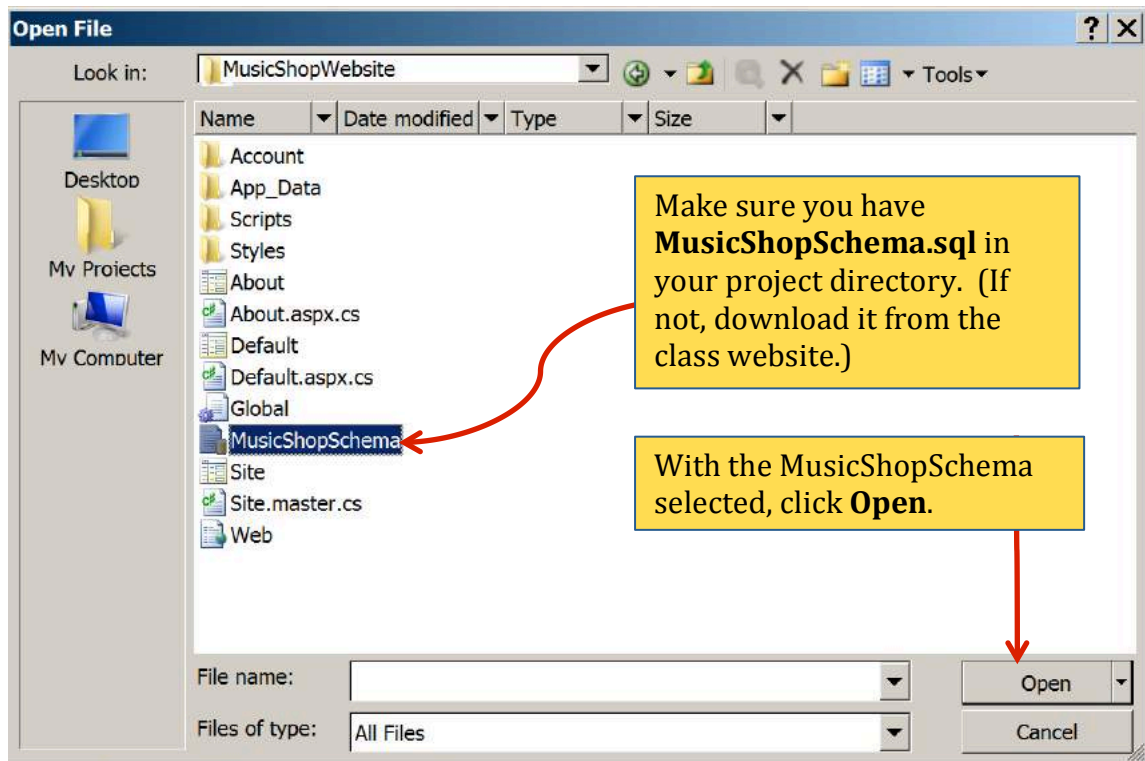
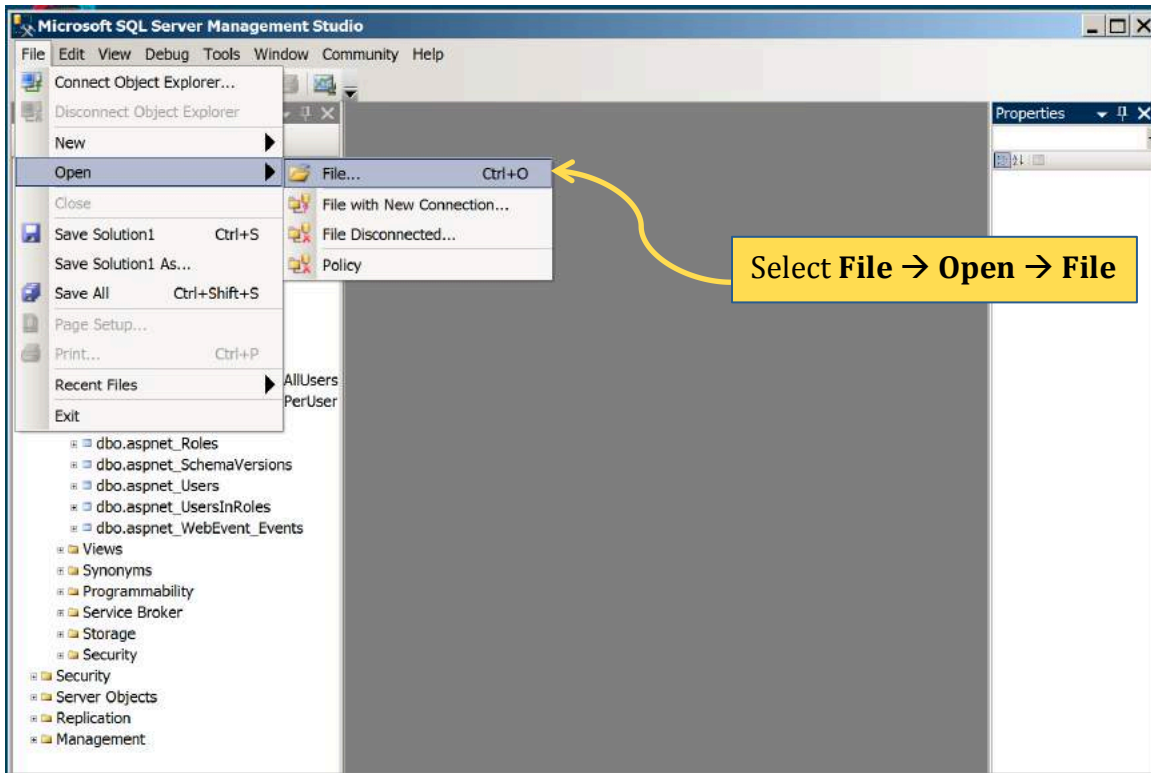
After clicking **Attach**, you will be presented with the following window:

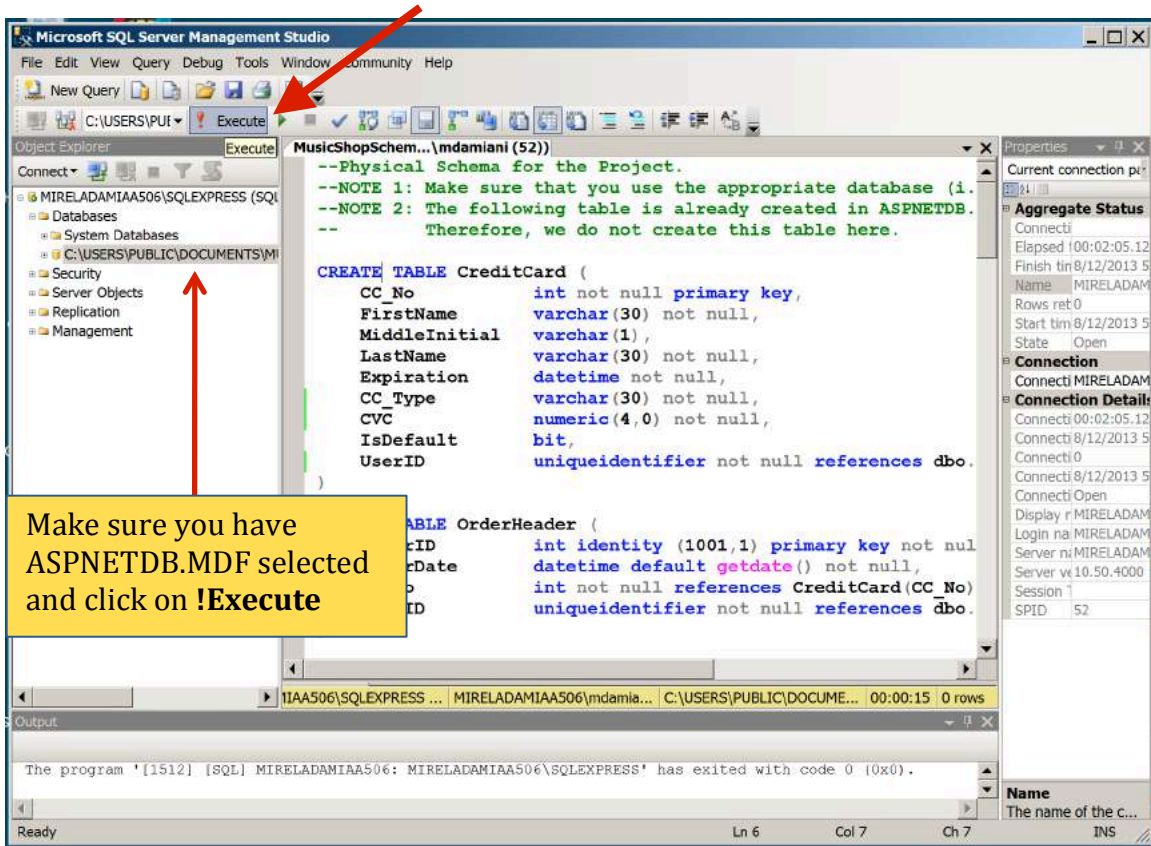


Navigate to your project folder under C:\Users\Public... and select ASPNETDB.MDF. (This file was created when you added yourself as a new user of the Visual Web Development Application).

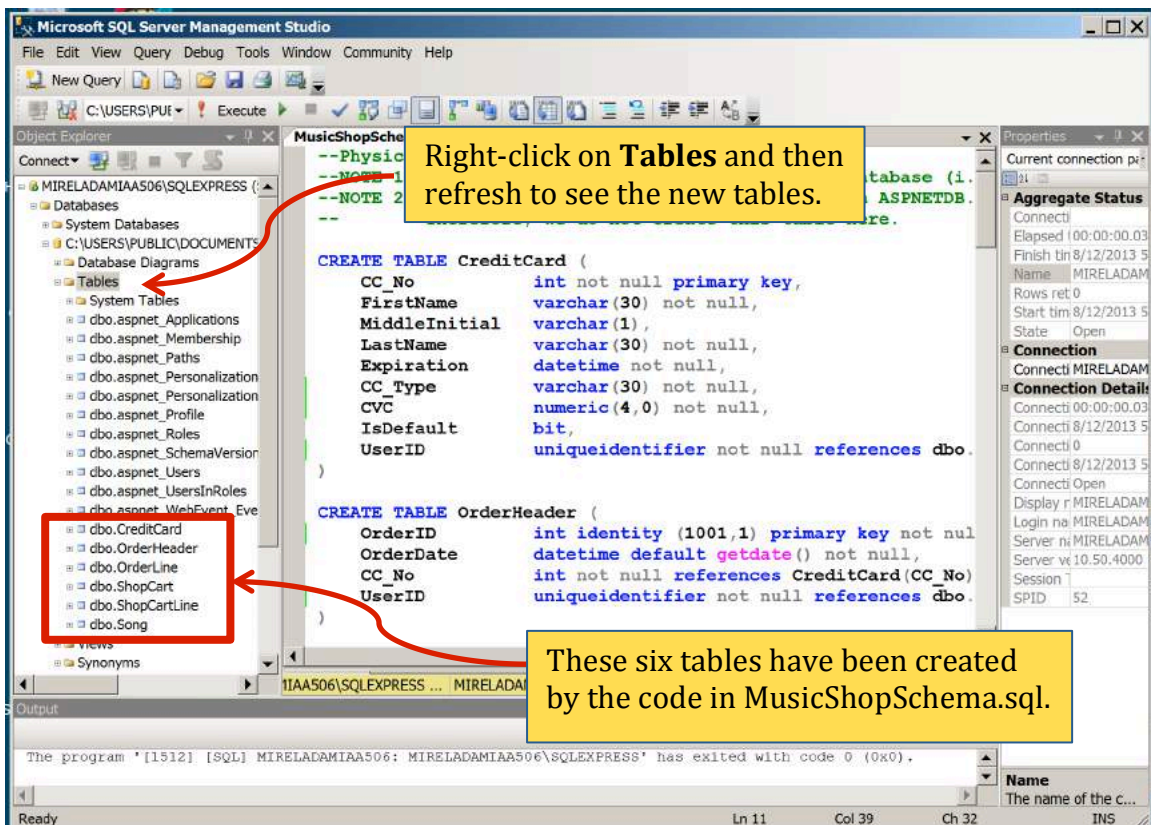




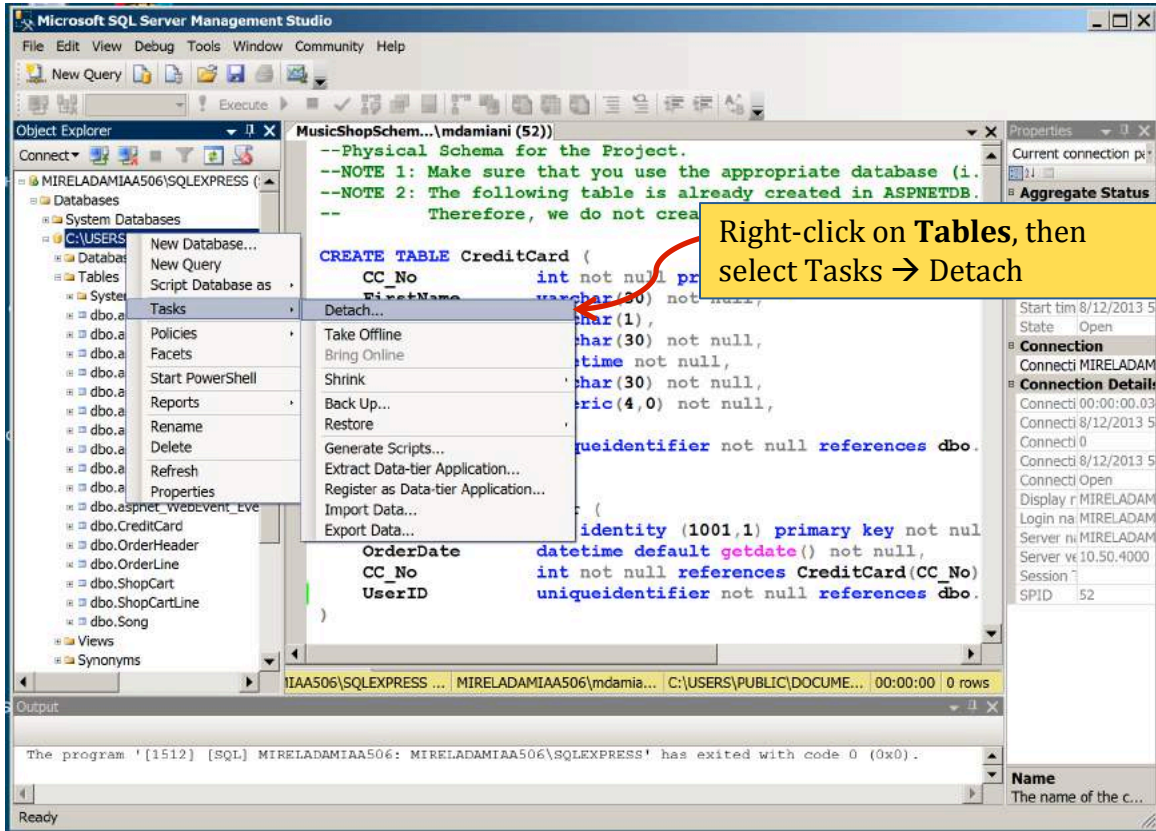




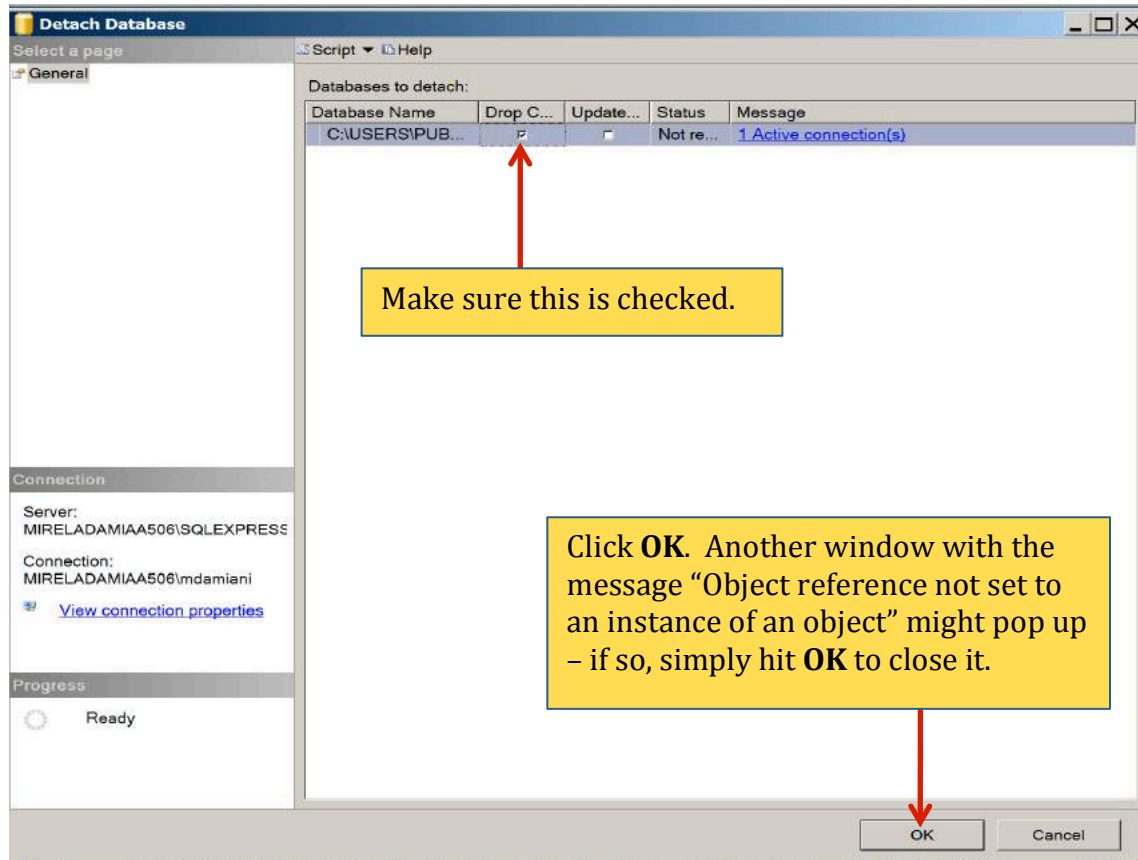
The SQL code from MusicShopSchema.sql adds 6 tables to the database: Song, ShopCart, ShopCartLine, OrderHeader, OrderLine and CreditCard. It also adds some sample data to the Song table. After the SQL code gets executed, you should be able to see the new tables in the Object Explorer when you refresh.



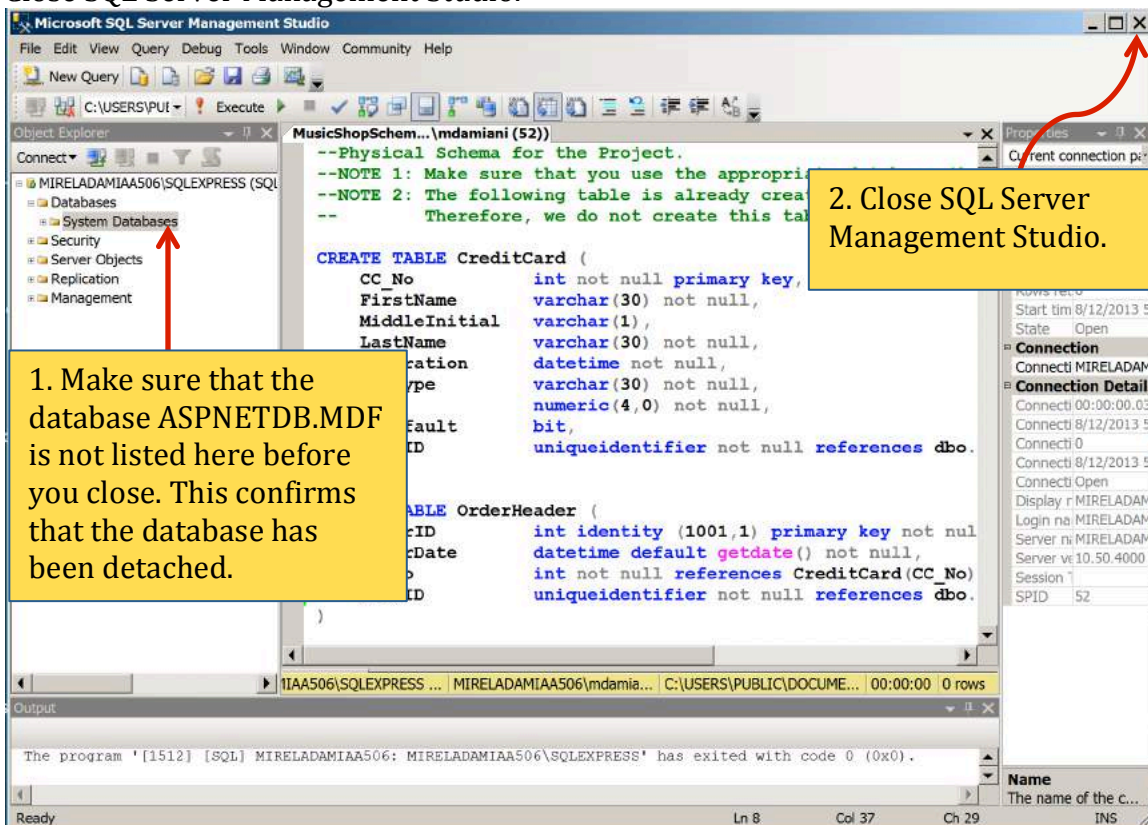
Now that we have created the tables we needed to work with, the job of the SQL server is done. We need to detach the database, so that it can be used by other applications (our Web Development server in particular).



You will be prompted with a window for selecting the database to detach.



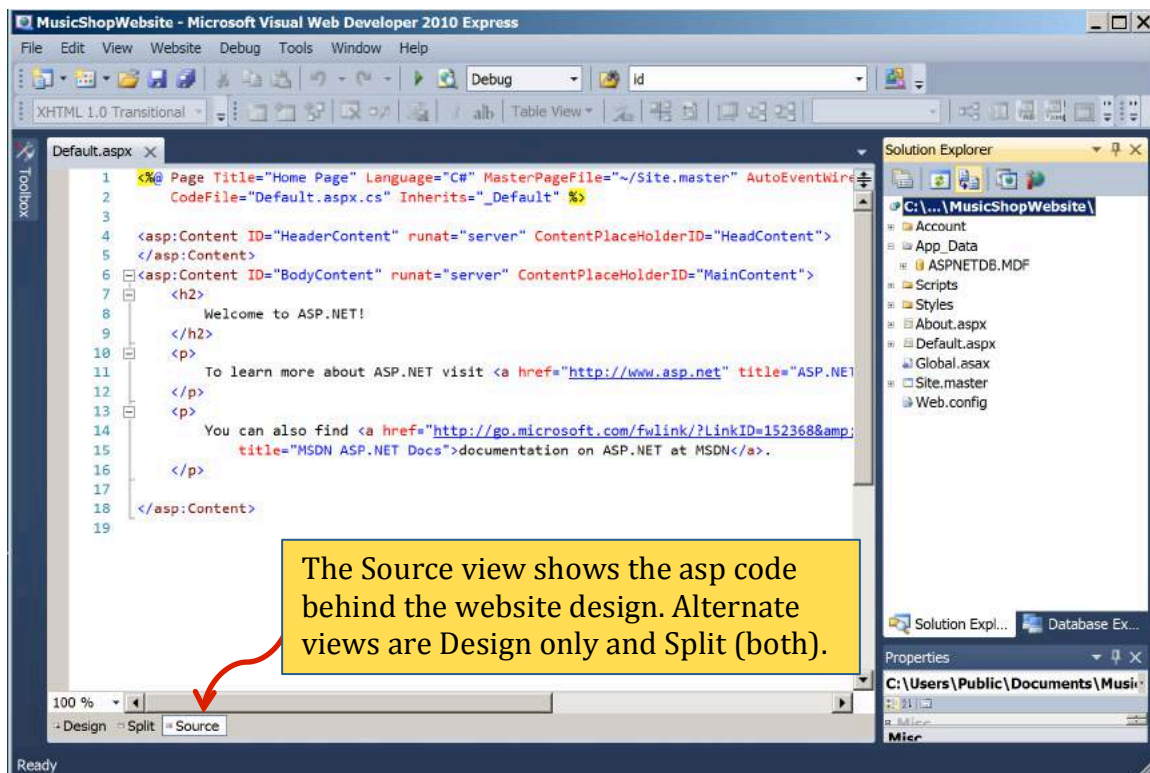
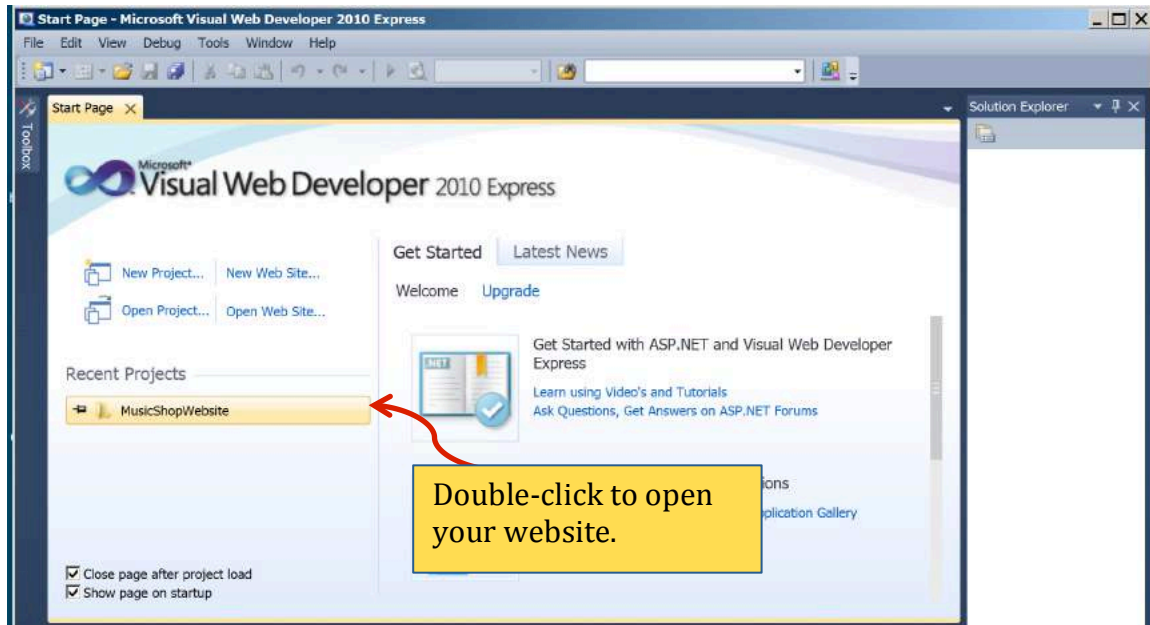
Close SQL Server Management Studio.

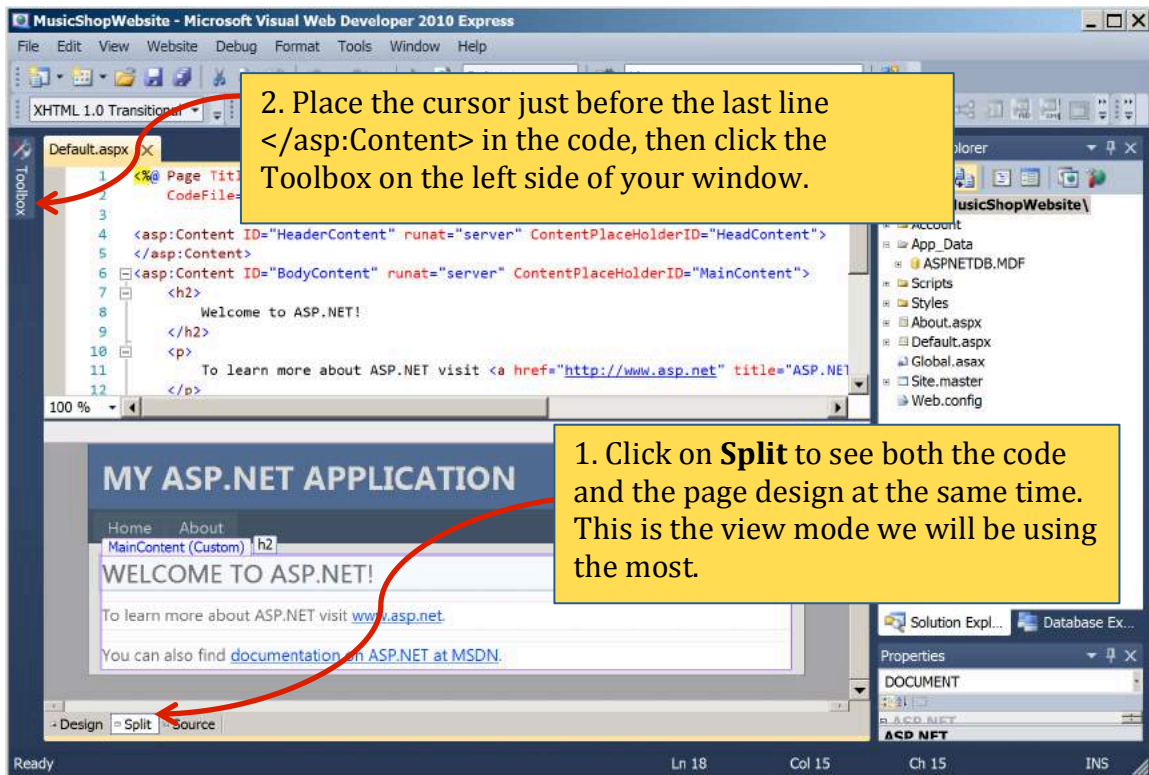


Step 4. Retrieving Data from the Database

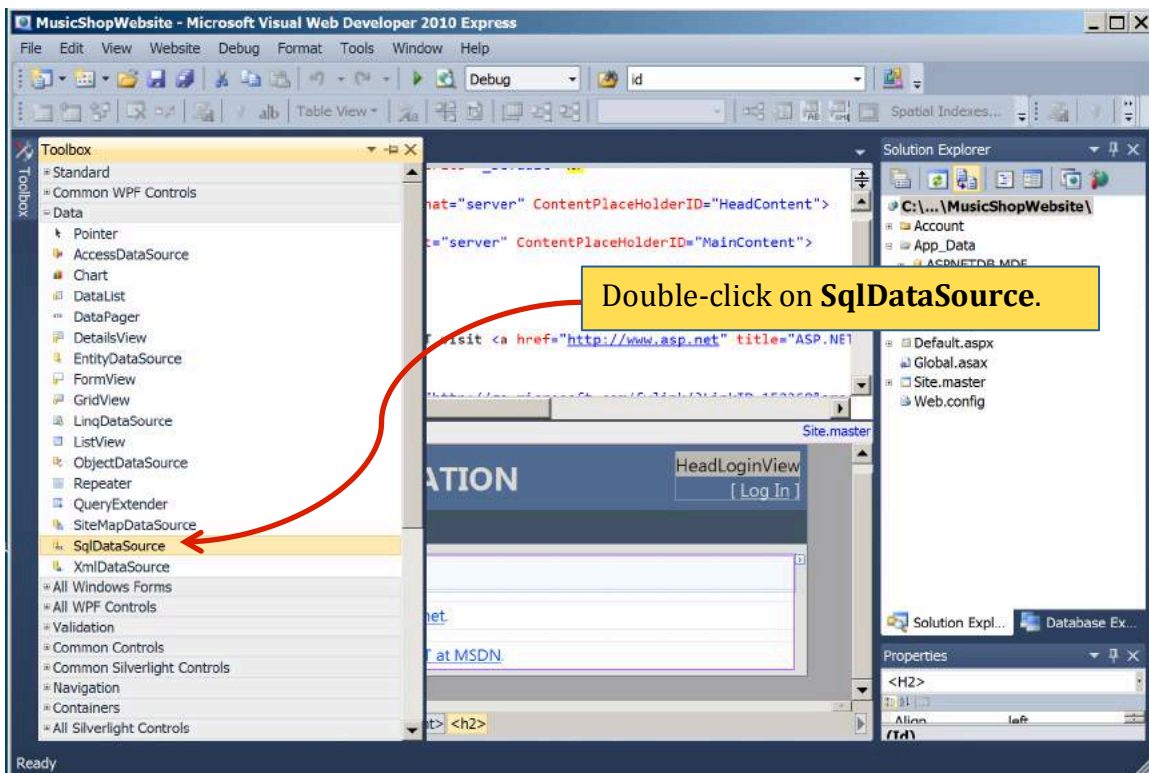
Now that we have created our database, our next step is accessing data from the database. Among many other Web controls, ASP.NET contains an SqlDataSource control designed to retrieve data from a database, and a GridView control designed to display the retrieved data on a page.

Open MS Visual Web Developer. Your website may be listed under Recent Projects, so you can click on it directly. If not listed, then select File → Open and navigate to your website in your Users/Public folder.

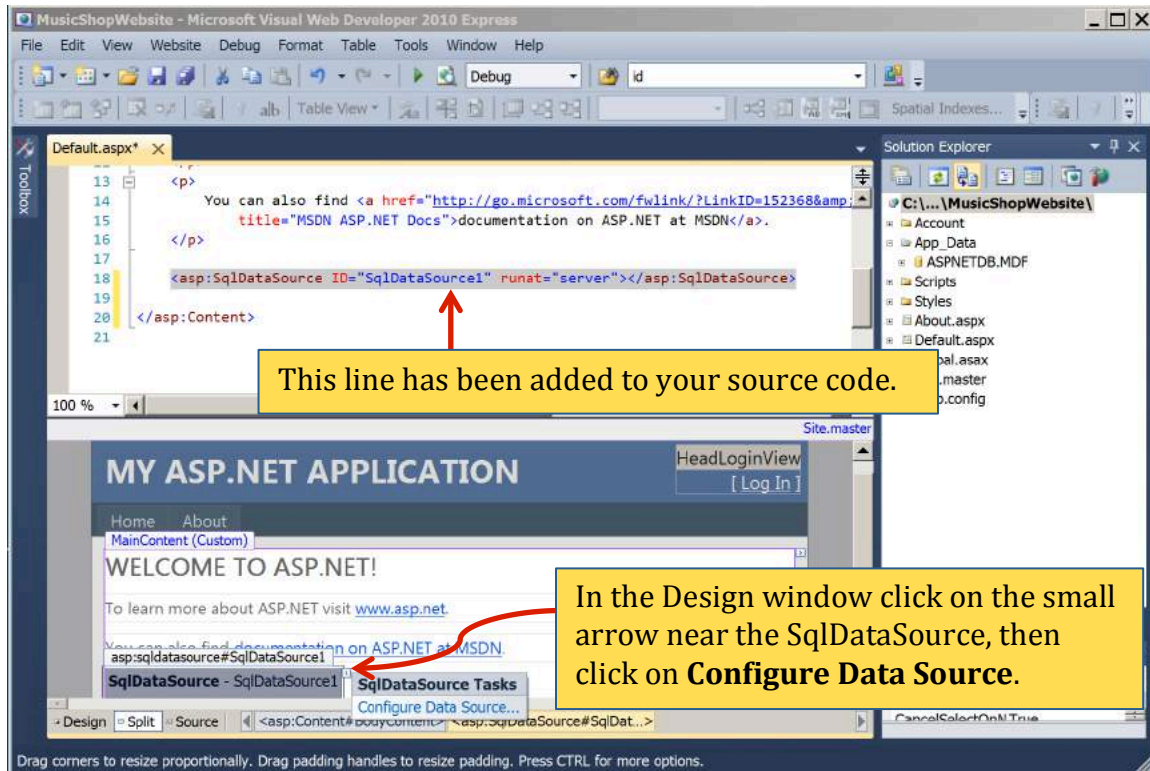




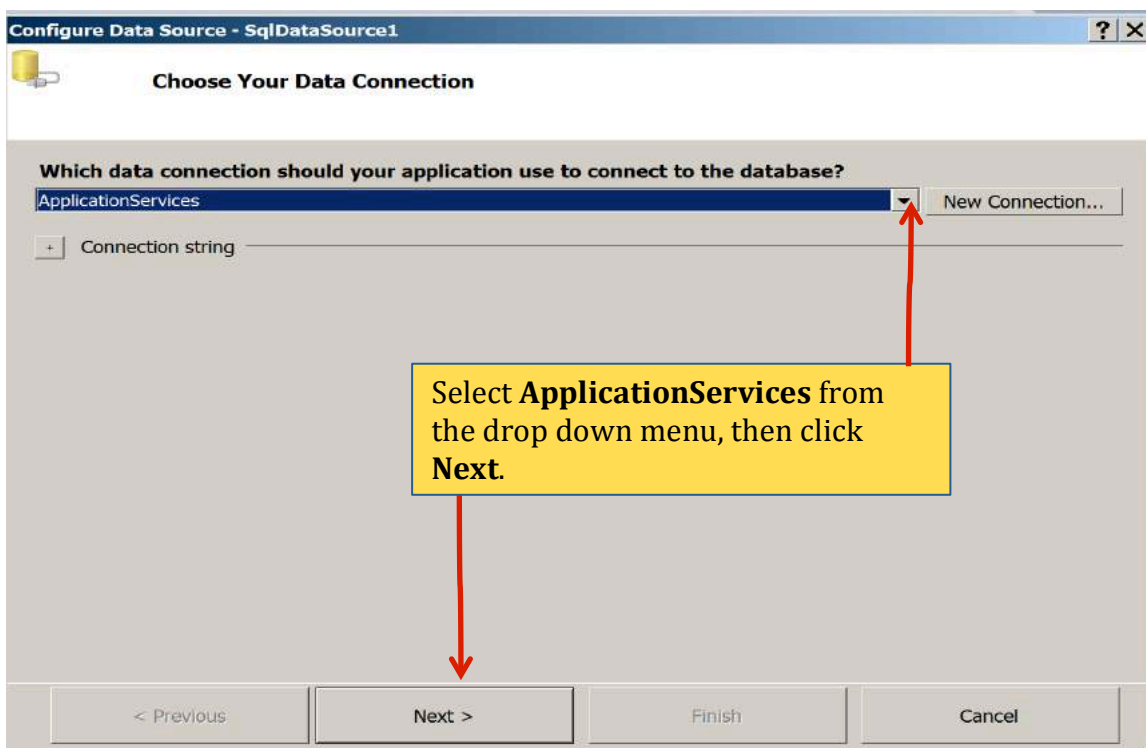
If you don't find the Toolbox, go to View → Other Windows → Toolbox, or Windows → Reset Window Layout. From the drop-down menu of the Toolbox, double click `SqlDataSource` (you'll find it under the Data section of the Toolbox).



Once you have added the **SqlDataSource** you should be able to see the code generated in the Source view.



To be able to retrieve data, the SqlDataSource control needs to know two bits of information: (1) How to connect to the database, and (2) What data to retrieve from the database. The control wizard will next prompt you to provide this information.



Configure Data Source - SqlDataSource1

Configure the Select Statement

How would you like to retrieve data from your database?

Specify a custom SQL statement or stored procedure
 Specify columns from a table or view

Name:

Columns:

- *
- SongID
- Title
- Album
- Artist
- Genre
- ReleaseDate
- ListPrice
- ClipName

Return only unique rows

WHERE...

ORDER BY...

Advanced...

SELECT statement:
SELECT * FROM [Song]

< Previous Next > Finish Cancel

1. Select **Song** from the drop down menu.

2. Make sure * (which means everything) is checked, then click **Next**.

Configure Data Source - SqlDataSource1

Test Query

To preview the data returned by this data source, click Test Query. To complete this wizard, click Finish.

SongID	Title	Album	Artist	Genre	ReleaseDate	ListPrice	ClipName
101	Disorganized Fun	Studio Album	Ronald Jenkees	Electronic	4/5/2009	0.9900	Fun.mp3
102	Harry Potter	Very Potter	StarKid	Showtunes	7/31/2010	0.9900	HarryPotter.mp3
103	Cannon in D	Ronald Jenkees	Ronald Jenkees	Classic	11/11/2007	0.9900	CannonD.mp3
104	Bad Day	Chipwrecked	Alvin & Chipmunks	Pop	11/15/2011	0.9900	Chipmunks
105	Beat Box	Album1	Joe Publique	Rock	11/11/2007	0.9900	BeatBox.mp3
106	Coyote Kisses	Acid Wolfpack	Joe and Bryce	Electronic	3/26/2012	0.9900	Coyote.mp3
107	Part of Me	Teenage Dream	Katie Perry	Pop-Rock	2/5/2012	0.9900	PartOfMe.mp3

1. You may test your query. Make sure that the above table is displayed upon clicking **Test Query**.

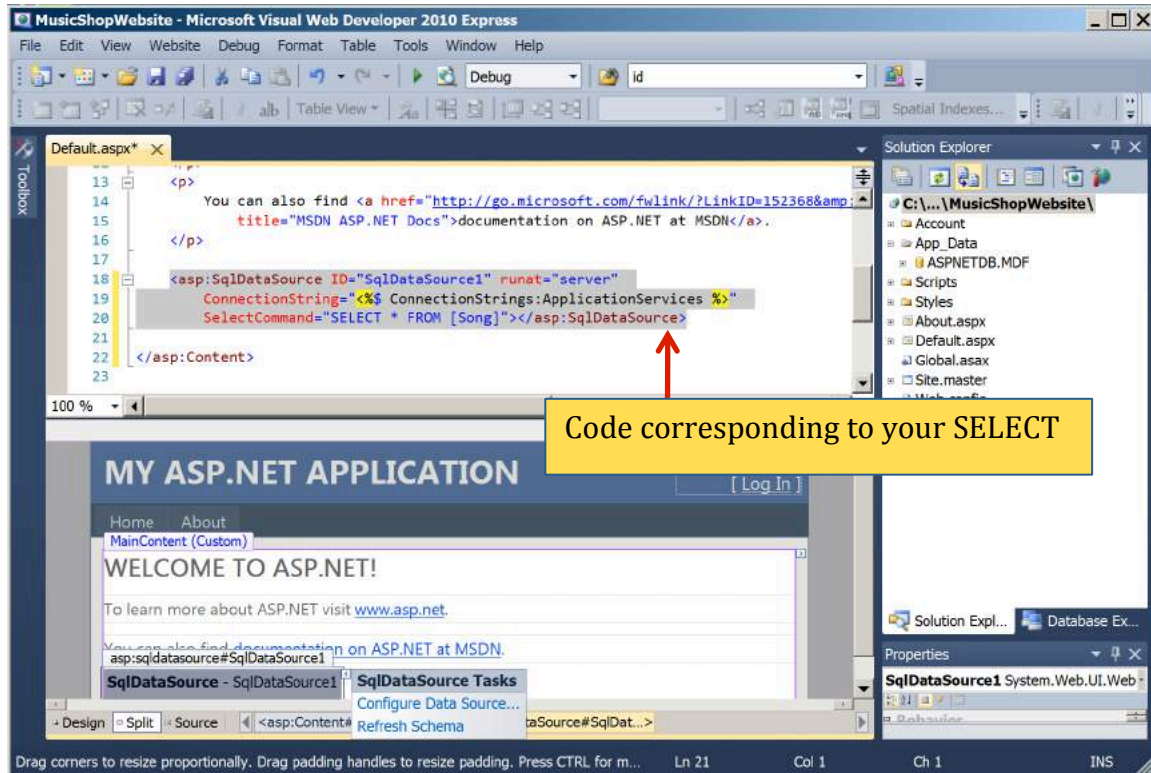
Test Query

SELECT statement:
SELECT * FROM [Song]

2. Click **Finish**.

< Previous Next > Finish Cancel

You should be able to see the table retrieved and the Select Statement from the database when you hit Test Query.

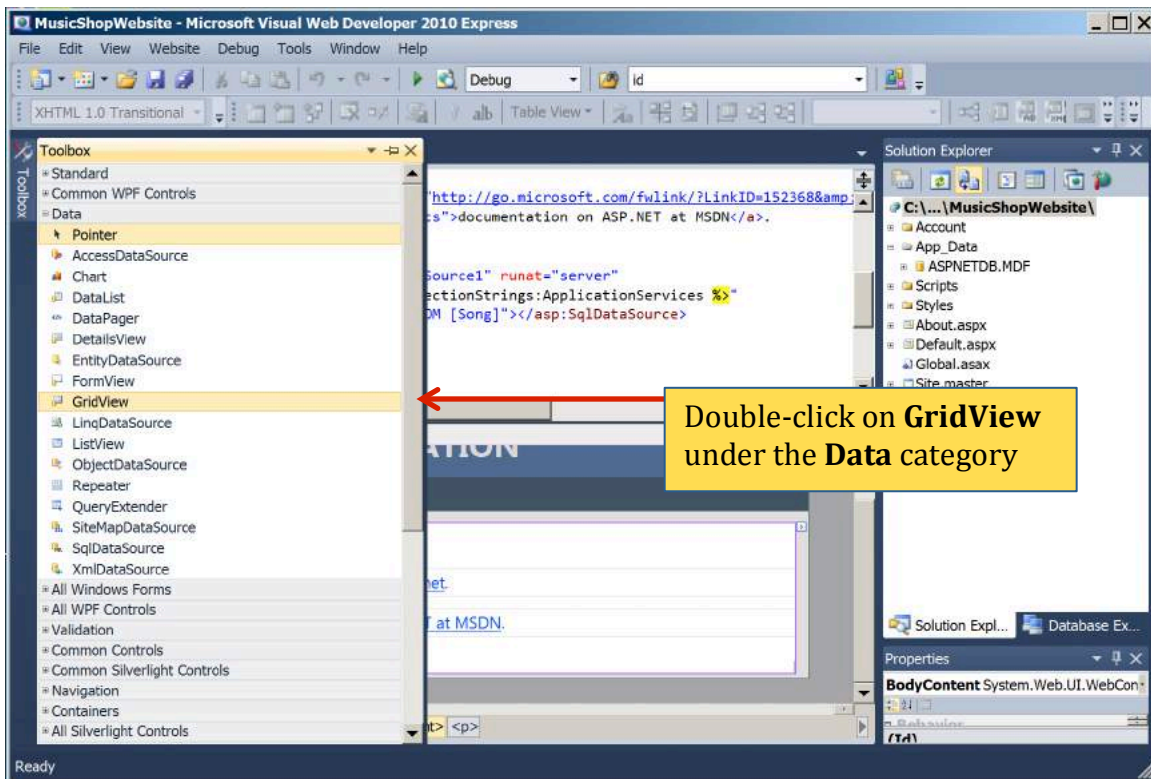
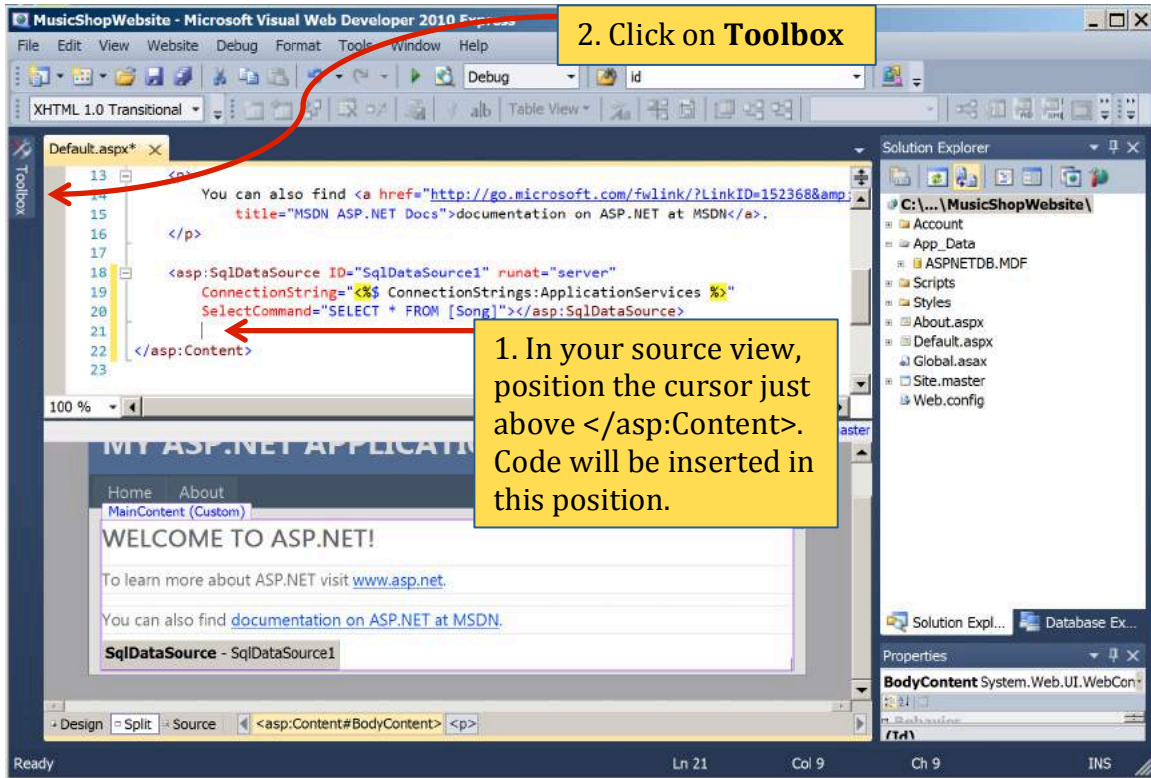


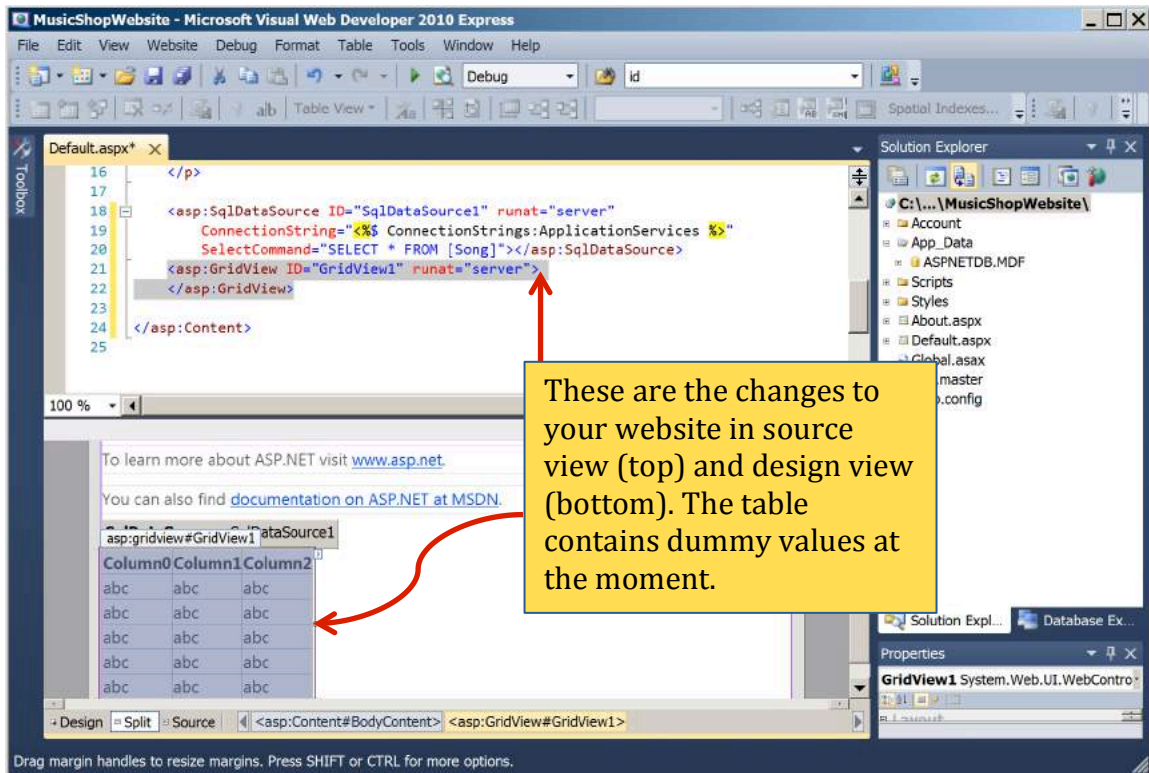
Step 5. Displaying Data with GridView

After the data source control has been added to the page and configured, displaying its data through a data Web control is a cinch: Simply drag the GridView control onto the ASP.NET page and, from its smart tag, specify what data source control to use. That's it! Take note of the name **SqlDataSource1** of your data source control (see the source code in Default.aspx).

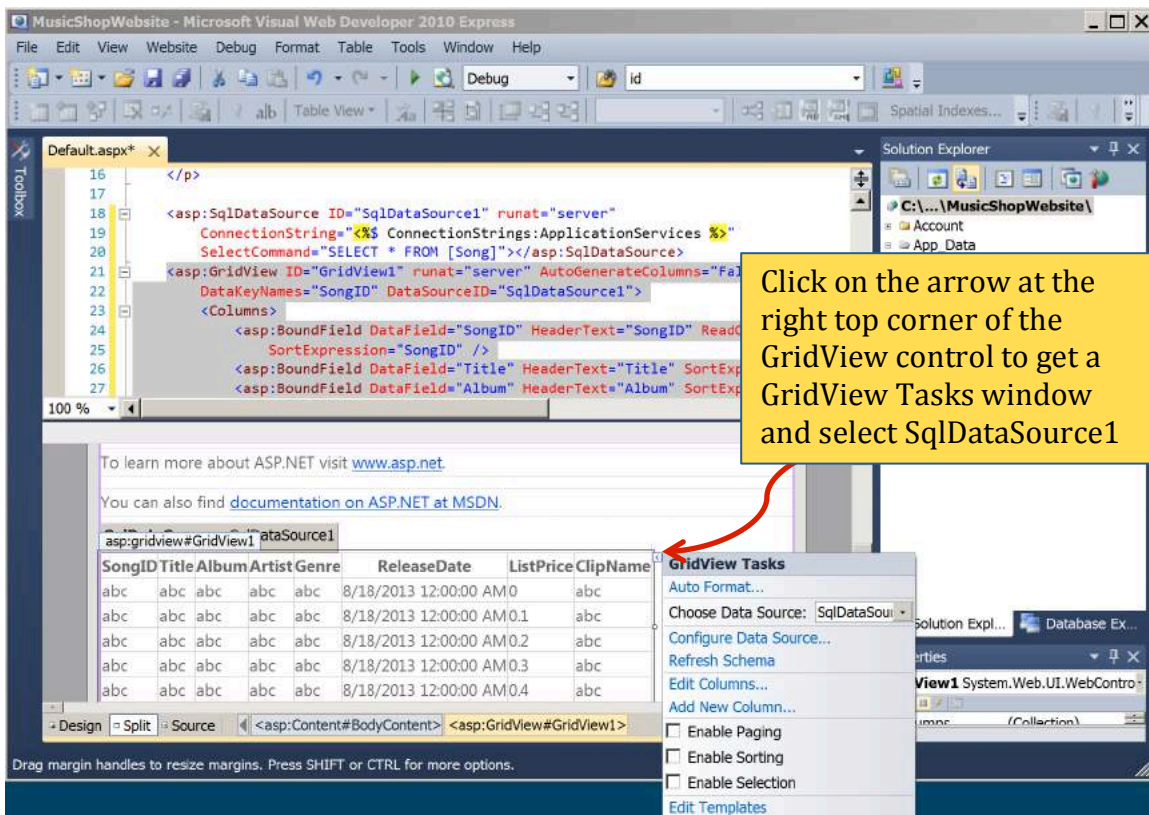
The GridView control will display data in a grid, with one row for each record in the data source control.

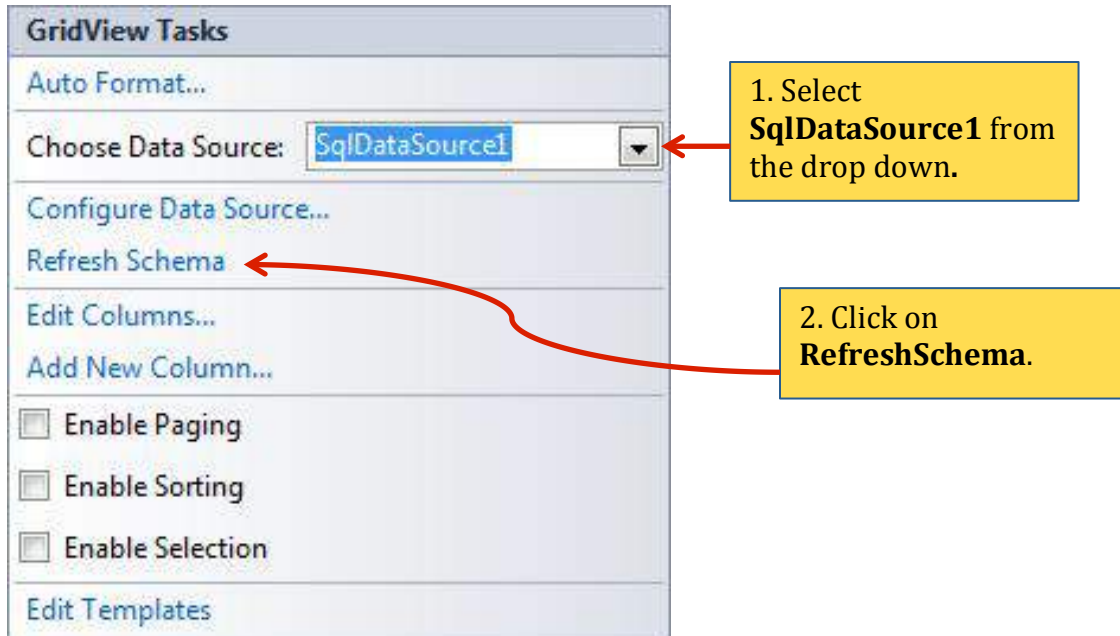
Position the cursor in the Source view similar to the screen shot below.



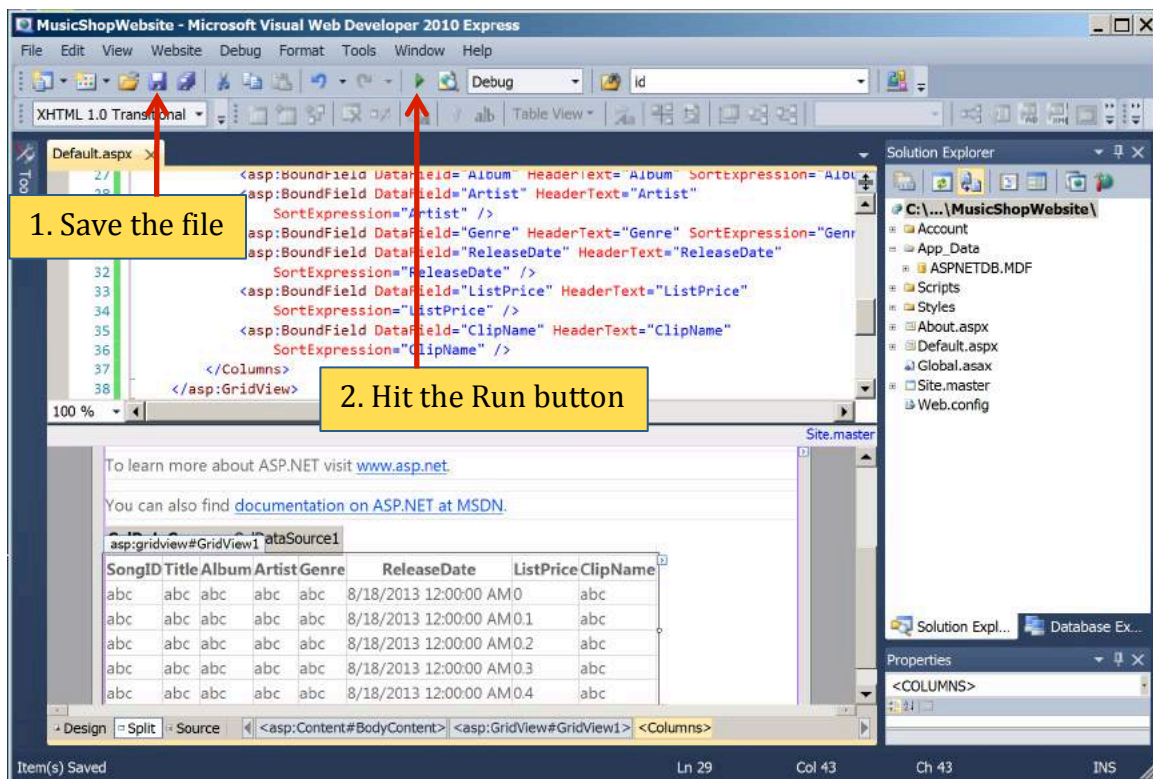


Next we need to associate a data source with the GridView. Click the arrow at the right top corner of the Grid View control to get a GridView Tasks window and select **SqlDataSource1** from the drop down.





Once you have selected the Data Source, hit on Refresh Schema to view the column names of the Song table in grid view.



Save the file and you will be able to view the Columns from the database. Now hit the Run button, so that you can view the page in a browser.

Home Page x
localhost:49228/MusicShopWebsite/Default.aspx

My ASP.NET APPLICATION [Log In]

Home About

WELCOME TO ASP.NET!

To learn more about ASP.NET visit www.asp.net.

You can also find [documentation on ASP.NET at MSDN](#).

SongID	Title	Album	Artist	Genre	ReleaseDate	ListPrice	ClipName
101	Disorganized Fun	Studio Album	Ronald Jenkees	Electronic	4/5/2009 12:00:00 AM	0.9900	Fun.mp3
102	Harry Potter	Very Potter	Starkid	Showtunes	7/31/2010 12:00:00 AM	0.9900	HarryPotter.mp3
103	Cannon in D	Ronald Jenkees	Ronald Jenkees	Classic	11/11/2007 12:00:00 AM	0.9900	CannonD.mp3
104	Bad Day	Chipwrecked	Alvin & Chipmunks	Pop	11/15/2011 12:00:00 AM	0.9900	Chipmunks
105	Beat Box	Album1	Joe Publicque	Rock	11/11/2007 12:00:00 AM	0.9900	BeatBox.mp3
106	Coyote Kisses	Acid Wolfpack	Joe and Bryce	Electronic	3/25/2012 12:00:00 AM	0.9900	Coyote.mp3
107	Part of Me	Teenage Dream	Katlie Perry	Pop-Rock	2/5/2012 12:00:00 AM	0.9900	PartOfMe.mp3

You can now view the actual values from the database.

Make sure you stop debugging every time after you run the program, because only by doing so you can make changes to your program.

MusicShopWebsite (Running) - Microsoft Visual Web Developer 2010 Express

File Edit View Website Debug Format Tools Window Help

Continue F5
Stop Debugging Shift+F5
 Build Web Site
 Step Into F11
 Step Over F10
 Step Out Shift+F11
 Toggle Breakpoint F9
 Windows
 Clear All DataTips
 Export DataTips ...
 Import DataTips ...
 Options and Settings...

Default.aspx X

XHTML 1.0 Transitional

Database Explorer

- Data Connections
- ASPNETDB.MDF
- Windows Azure
- Windows Azure Web Sites

```

HeaderText="Album" SortExpression="Album" /
HeaderText="Artist"
HeaderText="Genre" SortExpression="Genre" /
HeaderText="ReleaseDate"
HeaderText="ListPrice"
HeaderText="ClipName"

```

asp:gridview#GridView1 DataSource1

SongID	Title	Album	Artist	Genre	ReleaseDate	ListPrice	ClipName
abc	abc	abc	abc	abc	8/18/2013 12:00:00 AM	0	abc
abc	abc	abc	abc	abc	8/18/2013 12:00:00 AM	0.1	abc
abc	abc	abc	abc	abc	8/18/2013 12:00:00 AM	0.2	abc
abc	abc	abc	abc	abc	8/18/2013 12:00:00 AM	0.3	abc
abc	abc	abc	abc	abc	8/18/2013 12:00:00 AM	0.4	abc

Design Split Source <asp:Content#BodyContent> <asp:GridView#GridView1> <Columns>

http://go.microsoft.com/fwlink/?LinkID=152368&clcid=0x409 Ln 29 Col 43 Ch 43 INS

CONGRATULATIONS! You now have an ASP.NET page that displays the contents of the query executed by `SqlDataSource1`. Take a moment to review the process for displaying data in an ASP.NET page:

1. The first step is to add the data source control that retrieves the appropriate subset of data.
2. Next you add a data Web control (in our case `GridView`) and, through its smart tag, specify to use the data source added in Step 1.

Study the ASP.NET code in `Default.aspx`. You'll find it quite intuitive:

```
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%$ ConnectionStrings:ApplicationServices %>"
    SelectCommand="SELECT * FROM [Song]">
</asp:SqlDataSource>
```

```
<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
    DataSourceID="SqlDataSource1">
<Columns>
    <asp:BoundField DataField="SongID" HeaderText="SongID" ReadOnly="True"
        SortExpression="SongID" />
    <asp:BoundField DataField="Title" HeaderText="Title" SortExpression="Title" />
    <asp:BoundField DataField="Album" HeaderText="Album" SortExpression="Album" />
    <asp:BoundField DataField="Artist" HeaderText="Artist" SortExpression="Artist"/>
    <asp:BoundField DataField="Genre" HeaderText="Genre" SortExpression="Genre" />
    <asp:BoundField DataField="ReleaseDate" HeaderText="ReleaseDate"
        SortExpression="ReleaseDate" />
    <asp:BoundField DataField="ListPrice" HeaderText="ListPrice"
        SortExpression="ListPrice" />
    <asp:BoundField DataField="ClipName" HeaderText="ClipName"
        SortExpression="ClipName" />
</Columns>
</asp:GridView>
```

Soon you might be able to type in the HTML and Web control syntax by hand, rather than using the drag-and-drop capabilities of Web Developer.