

# SYSTEMS DESIGN / CAPSTONE PROJECT

## MIS 413

### User Guide 4.0

#### Binding Drop Down and Radio Button Lists to a SQL Data Source

Once you have built the proper Stored Procedures, you can use the SQL Data Source object to ‘automatically’ bind and build drop down lists, radio button lists and other objects.

- 1) Prior to attaching to the database, you need to have created your **SELECT** stored procedure.
- 2) **Test the stored procedure in SQL to insure you are getting the data you desire**
- 3) Back in Visual Studio; from your toolbox, find the “DATA” section
- 4) Slide a SQL Data Source onto your form (anywhere, it does not appear when running)
- 5) Click Configure Data Source (if this option does not appear, click the little arrow on the top right of the data source object you have slid on the page)
- 6) If you have previously created a data connection, select its name that from the drop down list, otherwise select NEW. Select MS SQL SERVER as your Data Source and the dialog box shown will appear (*you will only need to do this once per Database*)

The screenshot shows the 'Add Connection' dialog box in Visual Studio. The dialog is titled 'Add Connection' and contains the following fields and options:

- Data source:** Microsoft SQL Server (SqlClient) (Change...)
- Server name:** miscapstone (Refresh)
- Log on to the server:**
  - Use Windows Authentication
  - Use SQL Server Authentication
  - User name:** mis413
  - Password:** [masked]
  - Save my password
- Connect to a database:**
  - Select or enter a database name: mis413
  - Attach a database file: [empty] (Browse...)
  - Logical name:** [empty]
- Advanced...**
- Test Connection**, **OK**, **Cancel**

- 7) The properties are (see screen shot on previous page):
  - a. Server: miscapstone
  - b. Use SQL authentication (to log on to the server)
  - c. User Name: Either MIS 413 or the user name provided for your specific db
  - d. Password: the MIS 413 password was given in class, or the user password provided for your specific db
  - e. Check the 'save my password'
  - f. Select a database: either MIS 413 or the name of your specific database
  - g. Click Test Connection and you should receive an OK
- 8) It will then build a connection.
- 9) Click NEXT and accept the "Save" this connection option for future use
- 10) Once the connection is valid, it will then ask you for either the table, stored procedure name etc:
- 11) Select the "Specify a custom SQL stored procedure or store procedure", never select Specify columns from a table or view as it may lead to security issues. Click NEXT



### 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:

- 12) In the stored procedure drop down, select your previously built store procedure (i.e. validStatesSelectAll)

Configure Data Source - SqlDataSource1

**Define Custom Statements or Stored Procedures**

Click a tab to create a SQL statement for that operation.

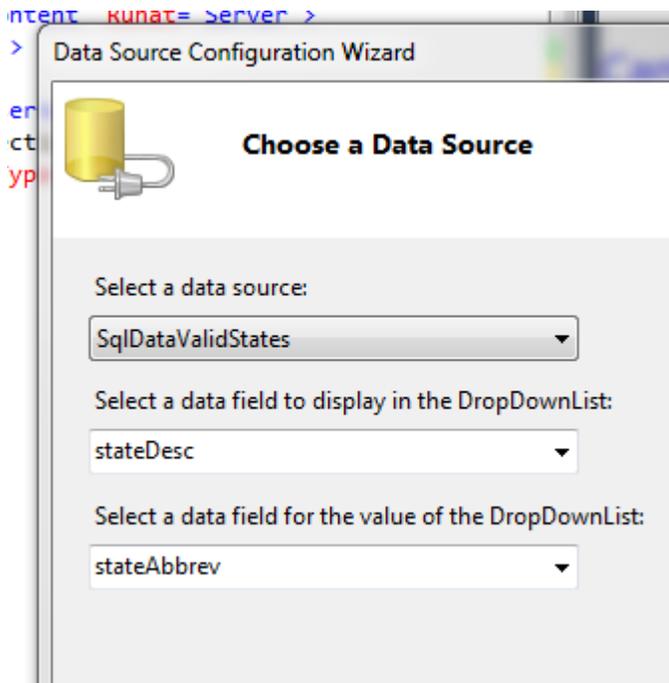
SQL statement:  
 Stored procedure:

< Previous

- 13) You can test the query at this point, and if valid select NEXT
- 14) Rename your SQLDataSource to represent the table/stored procedure as in:  
sqlDataValidStates or sqlDataValidMajors

**Populate the drop down or radio button list (if already build ignore steps 15 & 16.)**

- 15) Slide a drop down list or radio button list from the toolbox to your form in the appropriate location
- 16) Name your object (i.e. \_states or \_creditCardType)
- 17) Click the right arrow and choose DATA SOURCE
- 18) Select the SQL Data Source you built previously



- 19) Select the data field to display (you generally desire the Description field)
- 20) Select the data value field (you generally want the primary key field)
- 21) Click OK/Done and you are complete
- 22) Save your work
- 23) Test your application by clicking the green arrow to RUN, and your drop down list or radio button list should be populated!

**Special Properties to modify for the Drop Down List**

**A. Insure that the user has selected a value**

- 1) The previous steps will populate your drop down list with the items from the database so if you are using “States” the first entry would be Alabama. However we want to insure that the user has selected a state, so we would like the entry: -Please Select- to be the first entry in the drop down list.

- 2) To accomplish: click on your drop down list, go to properties, find the ITEMS property and add an ITEM with the text property of – Please Select- and value of 0.
- 3) Also change the Append Data Bound Property to TRUE. If you do not change this then the SQL statement will just override what is in the current list object.
- 4) Finally, if you have a Required Field Validator you need to set the Initial Value to 0 (the value for the –Please Select- item), thus the Required Field Validator will know if the user does not move from the initial Please Select item in the list.

## **B. A Drop Down or Radio Button List inside a GridView or similar object.**

- a. If you are modifying data in a grid view, you want the drop down list to automatically default to the current value for that item instead of the –Please Select- entry. (i.e if your drop down is states, you want it to default to NC or whatever value is in the database).
- b. To set the drop down to the current value in the database, you will use the Selected Value property. The DataTextField and DataValueFields will be populated from your SQL Data Source, however you will need to manually go to the HTML Source View and type in the:  
**SelectedValue = ‘<%# bind(“stateAbbrev”) %>’** code. NOTE: the field name in the selected value binding statement must be equal to the field name for the DataValue Field.
- c. Below is an example of a completed DropDownList inside a gridview

```
<asp:DropDownList ID="_states" runat="server"
DataSourceID="SqlDataStates" DataTextField="stateDesc"
DataValueField="stateAbbrev"
SelectedValue='<%# Bind("stateAbbrev") %>'>
<asp:ListItem Value="0">-Please Select-</asp:ListItem>
</asp:DropDownList>
```