

Creating Look-up Functionality

By Susan L. Miertschin

© 2003 by The McGraw-Hill Companies, Inc. All rights reserved.

Open a New Web Application Project

- ▶ Open Visual Studio
- ▶ File
 - ✖ New Project - ASP.NET Web Application
- ▶ Accept the name Default.aspx
- ▶ Fix it so that "Look-up Functionality" displays in the title bar of the browser at run time
- ▶ Fix the web.config file so that you can see error messages and turn off Windows Authentication
- ▶ Develop locally on usb
 - ✖ When complete: publish to lookup

2

Connect to the Database with Visual Studio's Server Explorer

- ▶ View
 - ✖ Server Explorer
- ▶ You may already have a data connection to the class FC database that uses your userid or WebUser as the logon id for the SQL server
- ▶ If you do not already have a data connection set up, go to the next slide to see what username and password combo is expected

3

Recall the Class SQL Server Connection Info for Fallcreek

cot-itec4339-02.cougarnet.uh.edu,17607

Logon id: WebUser

Password: Spring10#\$sec17607

Data Connection in Visual Studio

▶ Allows you to see

- ✖ Tables
- ✖ Views (to which you have been granted access)
- ✖ Stored procedures (to which you have been granted access)
- ✖ User-defined functions (to which you have been granted access)

5

No Separation of Function and Form Technique

This Example Uses the SQL Data Source Control

Look-up Functionality

- ▶ Select an identifier from a list
- ▶ Selection retrieves related data from other tables or from a view

7

Populate a Selection List

- ▶ Drag a SQL DataSource control onto the page
 - ✖ Configure it to retrieve both columns of information from vw_patient_names
- ▶ Drag a drop-down listbox onto the page
 - ✖ Configure it to use the SQL DataSource control to display patient names
- ▶ Save - Build - Start w/o Debugging to see data

8

AutoPostBack Property

- ▶ You want a selection in the drop-down list box to trigger a postback from the server
 - ✖ Why?
- ▶ Thus,
 - ✖ Select the drop-down list box in design mode
 - ✖ Look at its properties in the Properties window
 - ✖ Set Enable AutoPostBack property to True

9

Get Ready for Related Information - Create a Parameterized Query

- ▶ Drag another SQL Data Source control onto the form
- ▶ Configure it to use the same connection string
- ▶ But to deliver different data - based on the users selection of patient in the drop-down list
- ▶ Here's how ...

10

Build the Parameterized Query - 01

- ▶ Select the patient_note_t table as the source of data - check each field
- ▶ Click on Where button to create conditional where clause as basis for data to be retrieved

11

Build the Parameterized Query - 02

- ▶ Column to match is patient_in in patient_note_t table
- ▶ Match is to a control
 - ✗ The drop-down list control's selected value
 - ✗ Revealed through its SelectedValue property
- ▶ Add button adds a Where clause to the selection query
 - ✗ Where clause comparison phrase is displayed in dialog box

12

Text of Parameterized Query - Where Clause Phrase

[patient_in] = @patient_in

Value replacement is by
nameofdropdownlist.SelectedValue

13

Observe SQL

- ▶ Observe the SQL written
- ▶ Note that you can add an ORDER BY clause using the ORDER BY button
- ▶ Advanced lets you generate insert, delete, and update queries provided the field selection is sufficiently complete to allow data changes based on the data retrieved
- ▶ OK or Back to modify - Proceed to Finish
 - ✖ You may test the query on the way

14

Add a Display Control for the Related Data

- ▶ Add a data display control for the related table information
- ▶ Bind it to the SQL DataSource you just configured
- ▶ Enable paging
- ▶ Save - Build - Start w/o Debugging to test

(*Note* Only patients Chuck Rojas and Jesus Rojas had notes when I was testing this functionality. Chuck Rojas had 3 notes and Jesus Rojas had 1. Since you all have permission to change this data ... you will have to explore for yourself which patients have notes and how many.)

15

One More Set of Related Table Data

- ▶ Add a third SQL DataSource control and configure it to retrieve data from the patient-status_t table depending on the selection of patient from the drop-down list at run time
- ▶ Add a display control and bind it to the third SQL DataSource control

16

What if you wanted the status data and the note data to display in a single control

??????

17

Formatting the Grid View

- ▶ Refer to page 363 in the Bradley text - to the section titled Format the Grid
- ▶ There is also much info about this in the Boehm textbook

18

Build the Web Site/Solution

Run the Web Site/Solution

Will you see data?

Extension

- ▶ How do you fix the application so that the names in the dropdown list are in alphabetical order?

Creating Look-up Functionality

By Susan L. Miertschin

© 2003 by The McGraw-Hill Companies, Inc. All rights reserved.



© 2003 by The McGraw-Hill Companies, Inc. All rights reserved.
