Coding 8 - Use session state to store reservation data

In this exercise, you'll store the data that's entered into the Request page of the Reservation application so it can be displayed on a Confirmation page like the one that follows. To make that easier, you'll start from a project that contains the aspx code for the Confirmation page, the class that defines the session state object, and an enhanced Load event handler and a new DisplayReservation method for the Request page.

Review the new form, new class, enhanced Load event handler, and new method

1. Open the attached starting point web application and make sure you are working in a directory you normally use for working on projects in this class..

2. Open the Reservation class in the Models directory to see how it's defined. You may even want to print it so you can refer to it as you do this exercise.

3. Review the code for the Load event handler of the Request form. If the page request isn't a postback and a Reservation object is stored in session state (which means that it has already been created), this handler calls the DisplayReservation method that moves the data from session state to the controls on the Request page. Otherwise, the Load event handler sets the starting values for the arrival date text box and the radio buttons. Note that it sets the year in the copyright notice in the footer either way.

4. Review the code in the DispayReservation method to see how it gets the Reservation object from session state and then moves the data from the Reservation object to the controls on the form.

Code the Click event handler for the Submit button of the Request form

5. Start a Click event handler for the Submit button of the Request form. It should initialize a new Reservation object and then set its properties based on the values the user entered on the form. Then,

write a statement that saves the Reservation object in session state, and finish with a statement that redirects to the Confirmation page.

6. If you have any trouble writing the statements that convert the control data to properties in the Reservation object, here are some tips.

- To convert a text date to a DateTime object, use the Convert.ToDateTime method.

- To get the number of nights, you can subtract the departure date from the arrival date and then use the Days property of the resulting TimeSpan object.

- To convert a selection in the drop-down list for the number of people to an integer, use the Convert.ToInt32 method.

If you still have trouble, you can skip the conversions that you can't figure out.

Finish the Confirmation form

7. Review the Load event handler for the Confirmation form. This handler sets the year for the copyright in the footer and then displays reservation data in the labels on the form by calling a DisplayReservation method.

8. Code the DisplayReservation method. It should retrieve the Reservation object from session state, store it in a variable, and then get the values from the reservation object and move them to the label controls on the form. But if you skipped any of the properties in step 6 or 7, skip those in this method too.

9. Create a Click event handler for the Confirm Request button on the Confirmation form. It should display a message like the one on the form above.

10. Enhance the aspx code for the Modify Request button on the Confirmation form. It should post back to the Request form.

Test the application

11. Run the application, complete the Request form, and click the Submit button to display the Confirmation page. If the reservation data isn't displayed correctly, fix the problems and test again.

12. Click the Modify Request button to return to the Request form. The data for the reservation should still be displayed so you can modify it and click the Submit button again. When you do that, the Conformation page should show the changed data.

13. Continue testing and fixing until you're sure this application works.

Finally, publish your project to the web180even directory on the server. If your code won't run after publishing try clearing the files off of the server directory and FTP the whole project directly to the web180even directory. Finally, test your code on the server and submit the URL to the submission point.