Ceate a Reservation form

In this exercise, you'll create a Reservation form like the one below. To make this easier, you'll start from a web application that includes some of the aspx code for the form.

Open the web application and review the files

1. Open the XEx06Reservation web application.

2. Review the code in the Request.aspx web page. Then, set Request.aspx as the start page for the application.

3. Run the Request page to see how much work has been done for you. Notice that it includes the header, footer, headings, labels, text boxes for the arrival date and first name, and the buttons. So your job is to add the rest of the ASP.NET controls.

Add the controls for the form

4. Note that the control for the arrival date is just a text box. Then, add a text box for the departure date that's like the one for the arrival date.

5. Add the drop-down list for the number of people, with choices of 1, 2, 3, and 4.

6. Add a radio button list or three radio button controls for bed type. Note that the div that will contain these controls has a "bedtype" class already assigned.

7. Add the text boxes for the last name, email address, and telephone number entries.

Add the C# code for the form

8. Code an event handler for the Click event of the Submit button that displays this message in the label that's below the buttons:

    Thank you for your request.
    We will get back to you within 24 hours.

9. Code an event handler for the Click event of the Clear button that returns all the control values to their defaults. If you used individual radio buttons for the bed type, you may need to turn the checked attribute off for the Two Queens and One Queen options as well as turning the checked attribute for the King option on.

Add the required field validators

10. Add the required field validators to the right of the arrival date, departure date, first name, last name, email address, and telephone number fields. The error message should just be "Required Field". Set the CssClass property for this and all the other validators in this exercise to "text-danger".

Add the compare validators

11. Add a compare validator for the arrival date text box. This validator should check that a date is entered into this text box.

12. Add a compare validator for the departure date text box. This validator should check that the departure date is after the arrival date.

13. Test these validators, and fix any problems. When you test the arrival date for a valid date entry, keep in mind that many different formats are acceptable.

Add the regular expression validators

14. Add regular expression validators for the email address and telephone number fields.

15. Test these validators.

Make any final adjustments

16. Test the application one last time, and make any final adjustments. For instance, to get the error messages to display properly, the Display property for each one should be set to Dynamic, and the CssClass property should be set to "text-danger".