

Use CSS and Bootstrap to format a Quotation application

The application for this exercise is a one-page application with the user interface that's shown below, where the formatting is done with CSS and Bootstrap instead of a table as used to be done with past design methodologies. When you're done, the application will look like the one below.

For this exercise, you'll start a new web application and add Bootstrap to it. Then, you'll add the Default.aspx file and its code-behind file provided. Finally, you'll work in Source view to modify the Default.aspx file so the web application uses Bootstrap and custom CSS for formatting.

Price quotation

Sales price

Discount percent

Discount amount **\$100.00**

Total price **\$400.00**

Start a new web application and work with an existing file

1. Start a new web application named Coding05-3 in your project folder.
2. Add the Bootstrap NuGet package to your application.
3. Add the Default.aspx file from your solution for the Price Quotation app from Coding 4 to this web application. Then, display the page in Source view and delete the entire Style element in the head section of the document.
4. Change the opening table tag from table to main. Then, remove the tags for the rows and columns of the table, but keep the aspx code for the text box controls, label controls, validators, and button.
5. Add HTML label elements that identify the server controls for the text boxes and the label controls that display the results. Then, delete the Font-Bold properties for the first text box and the two label controls.
6. Right-click the Default.aspx file and select the View in Browser command. Review how the application looks now, and then leave the browser running.

Set up the head section of the document to work with CSS and Bootstrap

7. Add the following viewport metatag to the head section:
`<meta name="viewport" content="width=device-width, initial-scale=1" />`
8. Add a new style sheet named site.css to the Content folder.
9. Drag the Bootstrap style sheet and the new site.css file from the Content folder to the head section to create link tags. Remember that site.css needs to come after the Bootstrap style sheet.

10. Drag the Bootstrap and jquery JavaScript files from the Scripts folder to the head section to create script tags. Remember that the Bootstrap script tag needs to come after jquery script tag.

Use the Bootstrap grid CSS classes to lay out the page

For the remainder of this exercise, you should view your changes after each step to be sure they worked correctly. To do that, you can save the changes in Visual Studio, switch to the running browser, and then refresh the browser.

11. Set the class attribute of the form element to “form-horizontal”, and change the class attribute of the main element to “container”. Then, move the h1 element so it’s inside the main element, and set its class attribute to “jumbotron”.
12. Add a div element after the h1 element, and set its class attribute to “form-group”. Then, use the Bootstrap column CSS classes to make the sales price label span 3 columns, the sales price text box span 3 columns, and the sales price validation controls span 6 columns. Use nested div tags as needed to make everything line up the way you want it to.
13. Add form-group div elements for the discount percent, discount amount, and total price fields, and put the fields in columns like the ones for the sales price fields.
14. Add a form-group div for the calculate button, and offset it 3 columns so it lines up with the text boxes and label controls above it.

Use Bootstrap CSS classes for additional formatting

15. Add the “control-label” class to the HTML label elements that identify the server controls, add the “form-control” class to the text box server controls, and add the “btn” and “btn-primary” classes to the button server control.
16. Add the “text-danger” class to the validation controls. To view the changes, enter invalid data after you refresh the browser.

Use custom CSS classes

17. Open the site.css file and change its default type selector from body to label. Then, add a rule that sets the font-weight to normal. This will override the default Bootstrap style of making label elements bold.
18. Still in site.css, code a class selector named .bold. Then, add a rule that sets the font-weight to bold. Finally, add the “bold” class to the first text box and the two ASP.NET label server controls.