

Figure 6-50

Applying the tel and email data types

```

<label for="phone">Phone number</label>
<input name="phone" id="phone" type="tel"
placeholder="(nnn) nnn-nnnn" />

<label for="email">E-mail *</label>
<input name="email" id="email" type="email" />

```

input box contains a telephone number

input box contains an e-mail address

- 4. Save your changes to the file.
 - 5. If you have access to a mobile device that uses a virtual keyboard, access the **survey.htm** file and verify that the keyboard layout changes when the focus is placed on the input boxes for the phone and email fields.
- Trouble?** At the time of this writing, not all mobile devices alter their virtual keyboard layouts for different HTML5 data types.

Specifying Dates and Times

Next, Alice wants you to set the data type of the **visitdate** field to **date**. For browsers that support the **date** type, this will bring up a calendar widget from which users can select a date. The **date** data type is only one of five date types used for entering date and time values; **date** is used for recording a calendar date as a text string in the format **yyyy-mm-dd** where **yyyy** is the four-digit year value, **mm** is the two-digit month value, and **dd** is the two-day day-of-the-month value.

The various date and time data types do not have a lot of support in the browser market at the time of this writing. Only Opera displays a calendar widget, while Google Chrome for Windows currently provides a box with spin arrows that can be used to enter a date value. Some handheld devices such as the BlackBerry provide a calendar scroll for selecting a date and/or time.

You'll change the data type of the **visitdate** field to **date** now.

To change the data type of the visitdate field:

- 1. Return to the **survey.htm** file in your text editor.
- 2. Locate the **input** element for the **visitdate** field and insert the attribute **type="date"** as shown in Figure 6-51.

6-51 Applying the date data type

```
<fieldset id="experience">
<legend>Share Your Experience at Red Ball Pizza</legend>
<label for="visitdate">Date of visit</label>
<input name="visitdate" id="visitdate" type="date" />
```

input box contains a date

- 3. Save your changes to the file.
- 4. If you have access to the Opera, Chrome for Windows, or BlackBerry browser, or another browser that supports calendar data types, open the **survey.htm** file in the browser and click the input box for the date of the customer's visit to Red Ball Pizza. As shown in Figure 6-52, Opera displays a calendar object from which a user can select the date of the visit.

6-52 Calendar widget in the Opera browser

March							2014	
Mon	Tue	Wed	Thu	Fri	Sat	Sun		
24	25	26	27	28	1	2		
3	4	5	6	7	8	9		
10	11	12	13	14	15	16		
17	18	19	20	21	22	23		
24	25	26	27	28	29	30		
31	1	2	3	4	5	6		

Today

Trouble? If you are using other browsers, the input box might be treated as a text box with no calendar object displayed.

Browsers that don't support the various date and time data types display text boxes just as they did before the introduction of HTML5. Once again, nothing is lost by switching to one of the new date and time data types, and something is gained for customers using one of the browsers that support those types.

Using the number Data Type

Next, you'll record the number of times a typical Red Ball Pizza customer dines out per month. Because this is a numeric value, Alice wants you to use the **number** data type. Input boxes with the **number** data type are displayed using a **spinner control** in which users click an up or down arrow to increase or decrease the field value, respectively.

The default effect of clicking the spin arrow is to change the field value by one unit. You can specify a different amount for the value to change and identify the minimum and maximum values of the field using the `step`, `min`, and `max` attributes, as follows:

```
<input name="name" type="number" value="value"
      step="value" min="value" max="value" />
```

where the `value` attribute provides the default field value, the `step` attribute indicates the amount by which the field value changes when a user clicks the spin arrow, the `min` attribute defines the minimum possible value, and the `max` attribute defines the maximum possible value of the field.

REFERENCE

Creating Spinner Controls and Range Sliders

- To create a spinner control for numeric data, enter the `input` element

```
<input name="name" type="number" value="value"
      step="value" min="value" max="value" />
```

where the `value` attribute provides the default field value, the `step` attribute indicates the amount by which the field value changes when a user clicks the spin arrow, the `min` attribute defines the minimum possible value, and the `max` attribute defines the maximum possible value of the field.

- To create a range slider control for numeric data, use the following `input` element

```
<input name="name" type="range" value="value"
      step="value" min="value" max="value" />
```

You'll add a new field to the survey form named `ordersPerMonth` that queries customers about how often they dine out; you'll set its default value to 1 and set the value to range from 0 up to 10 in steps of 1 unit.

To create a number data type:

- 1. Return to the `survey.htm` file in your text editor.
- 2. Directly below the selection list for the `infoSrc` field, insert the following code (see Figure 6-53):

```
<label for="ordersPerMonth">
    How many times do you dine out per month?
</label>
<input name="ordersPerMonth" id="ordersPerMonth"
       type="number" value="1"
       min="0" max="10" step="1" />
```

Applying a number data type

```

<option value="word">Word of Mouth</option>
<option value="other">Other</option>
</select>

<label for="ordersPerMonth">
    How many times do you dine out per month?
</label>
<input name="ordersPerMonth" id="ordersPerMonth"
       type="number" value="1" min="0" max="10" step="1" />

```

contains a number

Month field are to 10 in steps of 1

default value of the ordersPerMonth field is 1

- 3. Save your changes to the file.

You also want to create a style rule for this input box so that the box is 70 pixels wide.

- 4. Go to the **forms.css** file in your text editor and add the following style rule at the bottom of the file (see Figure 6-54):

```

/* Number input box styles */

#ordersPerMonth {
    width: 70px;
}

```

Style rule for the ordersPerMonth field

```

/* Number input box styles */

#ordersPerMonth {
    width: 70px;
}

```

sets the width of the input box to 70 pixels

- 5. Save your changes to the file and refresh **survey.htm** in your Web browser. Figure 6-55 shows the appearance of the input box in the Opera browser.

Number spin box in the Opera browser

Where did you hear about us? (select all that apply)

Internet
Magazine
Newspaper
Word of Mouth
Other

How many times do you dine out per month?

1

click the spin arrows to increase and decrease the numeric value between 0 and 10

Trouble? Depending on your browser, you might not see spin arrows next to the input box.

- 6. If they're displayed in your browser, click the **up** and **down spin arrows** to verify that you can increase and decrease the value in the input box by 1 unit, and that the field value is limited to the range 0 to 10.

Browsers that do not support the `number` data type also ignore the `step`, `min`, and `max` attributes, so they have no effect in those browsers and the input box is treated as just another text box.

Specifying a Numeric Range with the `range` Data Type

Recall that Alice also wants to give customers the ability to rate Red Ball Pizza service and food quality on a numeric scale from 0 to 10. To describe this set of numbers, you can use the `range` data type

```
<input name="name" type="range" value="value"  
      min="value" max="value" step="value" />
```

where `name` is the name of the data field, the `value` attribute provides the default field value, the `min` attribute provides the minimum possible value, the `max` attribute provides the maximum value, and the `step` value provides the size of the steps between the minimum and maximum. For example, the following HTML5 code creates a range box for the `saturation` field that covers a range of values from 0 to 100 in steps of 5 with a default value of 50:

```
<input name="saturation" type="range" value="50"  
      min="0" max="100" step="5" />
```

The `range` data type is rendered differently from the `number` data type. Rather than a spin box, the control object is a slider in which the data value is selected by sliding a marker horizontally across a bar. Currently, the Opera, Safari, and Chrome browsers support the `range` data type while the Firefox and Internet Explorer browsers do not, treating range boxes as text boxes designed for simple text input.

You'll create a range slider now for the `serviceRating` and `qualityRating` fields, which are designed to record the customer's rating of Red Ball Pizza's service and food quality.

To create the two range boxes:

- 1. Return to the `survey.htm` file in your text editor.
- 2. Directly above the label for the `textarea` element, insert the following code as shown in Figure 6-56:

```
<label>Rate the overall service<br />  
    (0 = poor; 10 = great)</label>  
<input name="service" id="service" type="range" value="5"  
      min="0" max="10" step="1" />  
  
<label>Rate the food quality<br />  
    (0 = poor; 10 = great)</label>  
<input name="quality" id="quality" type="range" value="5"  
      min="0" max="10" step="1" />
```

.56

Adding input elements with the range data type

```

<label>Rate the overall service<br />
(0 = poor; 10 = great)</label>
<input name="service" id="service" type="range" value="5"
min="0" max="10" step="1" />

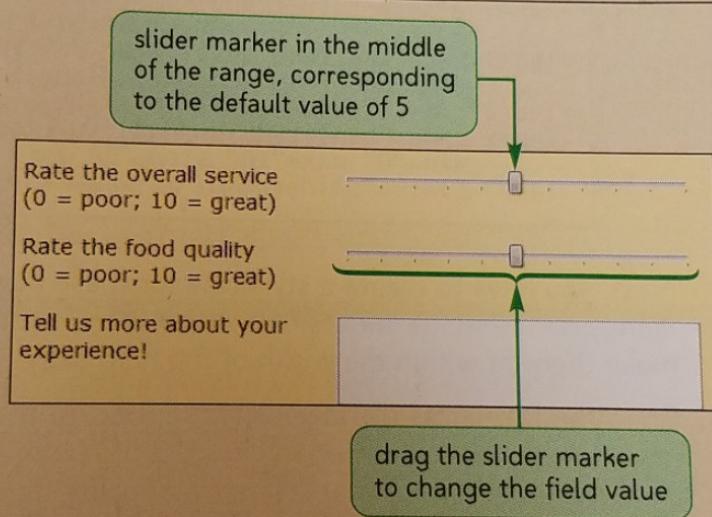
<label>Rate the food quality<br />
(0 = poor; 10 = great)</label>
<input name="quality" id="quality" type="range" value="5"
min="0" max="10" step="1" />

<label for="comments">Tell us more about your experience!</label>
<textarea name="comments" id="Comments"></textarea>

```

3. Save your changes to the file and then refresh the **survey.htm** file in your Web browser. Figure 6-57 shows the appearance of the two sliders in the Opera Web browser. Notice that the default value of 5 is represented by placing the slider marker in the exact center of the 0 to 10 range.

Range slider in the Opera browser



4. Drag the marker on the slider back and forth to confirm that the widget works as expected.

Trouble? If you are running Firefox or Internet Explorer, you might only see a text box with a default value of 5.

A problem with the range slider is that the minimum and maximum values represented on the slider bar are not displayed on the widget, so there is no indication of what value is actually being selected and stored in the data field. To add that information, you'll insert a label directly before and directly after the range slider to tell users that the minimum value on the slider represents a 0 and the maximum value represents a 10.

To add descriptive labels to the range slider:

- 1. Return to the **survey.htm** file in your text editor.
- 2. Directly before the **input** element for the service range slider, insert the following **label** element:
`<label class="sliderLabel">0</label>`
- 3. Directly after the **input** element for the service range slider, insert the following **label** element:
`<label class="sliderLabel">10</label>`
- 4. Repeat Steps 2 and 3 for the **input** element for the food quality range slider.
 Figure 6-58 highlights the revised code.

3 Adding descriptive labels to the range slider

```

<label>Rate the overall service<br />
(0 = poor; 10 = great)</label>
<label class="sliderLabel">0</label>
<input name="service" id="service" type="range" value="5" 
min="0" max="10" step="1" />
<label class="sliderLabel">10</label>

<label>Rate the food quality<br />
(0 = poor; 10 = great)</label>
<label class="sliderLabel">0</label>
<input name="quality" id="quality" type="range" value="5" 
min="0" max="10" step="1" />
<label class="sliderLabel">10</label>

```

Next, you have to create a style rule for these new labels so that they appear alongside the range sliders. You also have to resize the labels and the sliders to make them fit within the width of the field set.

- 5. Go to the **forms.css** file in your text editor. At the bottom of the file, insert the following style rules as shown in Figure 6-59. Note that you have to specifically *not* clear the label text so that it floats alongside the other objects within that line on the form.

```

/* Range slider styles */

label.sliderLabel {
  clear: none;
  font-size: 0.7em;
  margin: 10px 0px;
  text-align: center;
  width: 10px
}

input[type="range"] {
  width: 150px;
}

```

9 Style rules for the range slider

displays without waiting for the margins to be clear of floats

reduces the font size to 0.7 em

sets the top/bottom margin to 10 pixels; centers the text; and sets the width to 10 pixels

sets the width of the range slider to 150 pixels

```
/* Range slider styles */
label.sliderLabel {
    clear: none;
    font-size: 0.7em;
    margin: 10px 0px;
    text-align: center;
    width: 10px
}

input[type="range"] {
    width: 150px;
}
```

- 6. Save your changes to the file and then refresh **survey.htm** in your Web browser. Figure 6-60 shows the appearance of the range slider in the Opera browser.

Range slider in the Opera browser

Rate the overall service
(0 = poor; 10 = great)

Rate the food quality
(0 = poor; 10 = great)

INSIGHT

Exploring Form Controls with JavaScript Libraries

With the current lack of cross-browser support for many of the HTML5 form elements, you should only use them if they are not essential to reading and successfully completing your Web form. On the other hand, if your form requires the use of a spin box or a range slider, you should confine yourself to HTML 4 or XHTML elements and create your own control elements from a JavaScript library.

JavaScript libraries or JavaScript frameworks enhance HTML by adding elements and controls created using the JavaScript language. Among the most popular frameworks are jQuery, MooTools, Dojo Toolkit, and Yahoo! UI Library. Each provides the tools to create enhanced form control objects such as sliders and spin boxes.

To use these tools, you have to download the JavaScript source file for the library and usually another JavaScript program that creates the control object. Most of the libraries work with the input elements already present in your Web form and will include documentation about how you should modify the markup tags to take advantage of a specific JavaScript library. If you do use a JavaScript library, be aware that your users must have JavaScript enabled to be able to use those enhanced control objects.

Suggesting Options with Data Lists

The last data field that Alice wants added to the survey form is a text box in which customers can indicate their favorite Red Ball Pizza dish. There are a lot of possible answers and Alice doesn't want to limit the options to a selection list, but she does want to provide suggestions to customers as they type their entries. For example, as the user types a *B*, Alice would like the text box to display entries such as *Big Kahuna Pizza* or *BBQ Chicken Pizza* (both of which are Red Ball Pizza specials that start with *B*).

You can create this effect using the HTML5 `datalist` element

```
<datalist id="id">
    <option value="value" />
    <option value="value" />
    ...
</datalist>
```

where `id` is the id of the list and the values assigned to the different `option` elements provide the suggested entries in the list. To apply a `datalist` to a text input box, add the `list` attribute to the `input` element as follows

```
<input name="name" list="id" />
```

where `id` references the id of the data list. For example, to create an input box for the `favDish` field that offers a few suggested items, you could enter the following HTML5 code:

```
<input name="favDish" list="dishes" />
<datalist id="dishes">
    <option value="Antipasto Pizza" />
    <option value="Big Kahuna Pizza" />
    <option value="BBQ Chicken Pizza" />
</datalist>
```

The options in the `dishes` data list are just suggestions. The customer is not obligated to accept any options and can type a dish of his or her own choosing. Currently, the `list` attribute and the `datalist` element are supported only by the Firefox and Opera browsers. Other browsers that encounter this code ignore both the attribute and the element, and treat the input box as just another text box.

Creating and Applying a Data List

- To create a data list of possible values, enter the HTML code

```
<datalist id="id">
    <option value="value" />
    <option value="value" />
    ...
</datalist>
```

- where the `value` attribute provides the text of the possible values in the data list.

`<input name="name" list="id" />`

where `id` references the id of the data list structure.

Add an input box for the `favDish` field to the survey form now and augment it with a data list of suggested Red Ball Pizza dishes.

To create a data list:

- 1. Return to the **survey.htm** file in your text editor. Directly below the input box for the `ordersPerMonth` field, insert the following code (see Figure 6-61):

```
<label for="favDish">What's your favorite Red Ball dish?</label>
<input name="favDish" id="favDish" list="dishType" />
<datalist id="dishType">
    <option value="Antipasto Pizza" />
    <option value="Big Kahuna Pizza" />
    <option value="BBQ Chicken Pizza" />
    <option value="Mediterranean Herb Pizza" />
    <option value="Pasta Rolls" />
    <option value="Pesto Artichoke Pizza" />
    <option value="Sal's Stuffed Pizza" />
    <option value="Wing'd Pizza" />
</datalist>
```

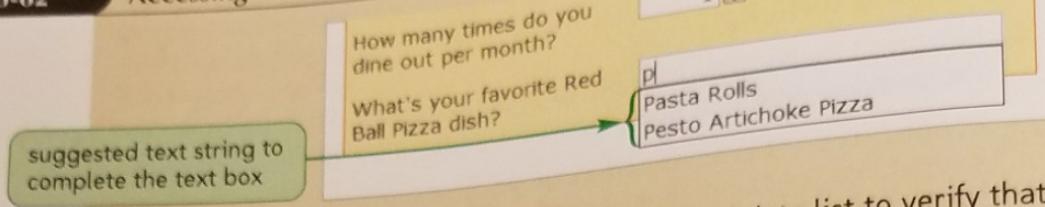
Adding a data list

```
<input name="ordersPerMonth" id="ordersPerMonth"
       type="number" value="1"
       min="0" max="10" step="1" />

<label for="favDish">What's your favorite Red Ball dish?</label>
<input name="favDish" id="favDish" list="dishType" />
<datalist id="dishType">
    <option value="Antipasto Pizza" />
    <option value="Big Kahuna Pizza" />
    <option value="BBQ Chicken Pizza" />
    <option value="Mediterranean Herb Pizza" />
    <option value="Pasta Rolls" />
    <option value="Pesto Artichoke Pizza" />
    <option value="Sal's Stuffed Pizza" />
    <option value="Wing'd Pizza" />
</datalist>
```

- 2. Save your changes to the file and then reopen **survey.htm** in the Firefox or Opera Web browser.
- 3. Click the input box for the `favDish` field and type the letter **p**. In Opera, the browser displays a list of two menu items that begin with the letter P (see Figure 6-62). In Firefox, the browser displays any option that contains the letter P, not just options that begin with the letter P.

6-62 Accessing a data list of options



4. Continue to type in letters from the selected data list to verify that the suggested food items change to reflect your text entry.

Now that you've completed all of the fields for the survey form, you'll next examine how to submit the form for processing. To do that, you'll create a form button.

Creating Form Buttons

So far, all of your control elements have involved entering field values. Another type of control element is one that performs an action. In forms, this is usually done with **form buttons**, each of which performs one of three actions:

- Run a command from a program linked to the Web page.
- Submit the form to the program running on the server.
- Reset the form to its starting values.

The first type of button you'll examine is the command button.

Creating a Command Button

A **command button** runs a command that affects the contents of the Web page or the Web browser itself. Command buttons are created by applying the **button** attribute to the **input** element

```
<input type="button" value="text" />
```

where **text** is the text that appears on the button. To associate an action with a command button, you link the button to a program that runs when the command button is clicked using the **onclick** attribute, as follows

```
<input type="button" value="text" onclick="program" />
```

where **program** is the name of the program. In most cases, the program is stored in a JavaScript file that is linked to the Web page. For example, the **input** element

```
<input type="button" value="Run Program" onclick="setup()" />
```

creates a command button that runs the JavaScript program **setup()** when the button is clicked. You won't be applying command buttons to the survey page.

Creating Submit and Reset Buttons

The two other kinds of form buttons are submit and reset buttons. A **submit button** submits a form to the server for processing when clicked. Clicking a **reset button** resets the form, changing all field values to their original default values and deleting any values that a user might have entered into the form. The syntax for creating these two buttons is

```
<input type="submit" value="text" />
<input type="reset" value="text" />
```

where once again the `value` attribute defines the text that appears on the button.

REFERENCE

Creating Form Buttons

- To create a form button to run a command, use the element

```
<input type="button" value="text" onclick="program" />
```

where `text` is the text that appears on the button and `program` is the program that is run in response to the user clicking the button.

- To create a form button to submit the form and its fields and values to a script, use the following element:

```
<input type="submit" value="text" />
```

- To create a form button to reset the form to its default values and appearance, use the following element:

```
<input type="reset" value="text" />
```

Alice wants the survey form to include both a submit button and a reset button. The submit button, which she wants labeled *Submit My Survey*, will send the form data to the server for processing when clicked. The reset button, which she wants labeled *Cancel*, will erase the user's input and reset the fields to their default values.

To add the submit and reset buttons to the survey form:

- Return to the **survey.htm** file in your text editor.
- Scroll to the bottom of the file. Directly above the closing `</form>` tag, insert the following code (see Figure 6-63):

```
<p>
  <input type="submit" value="Submit My Survey" />
  <input type="reset" value="Cancel" />
</p>
```

Creating submit and reset buttons

submit button type

reset button type

`<p>`
 `<input type="submit" value="Submit My Survey" />`
 `<input type="reset" value="Cancel" />`

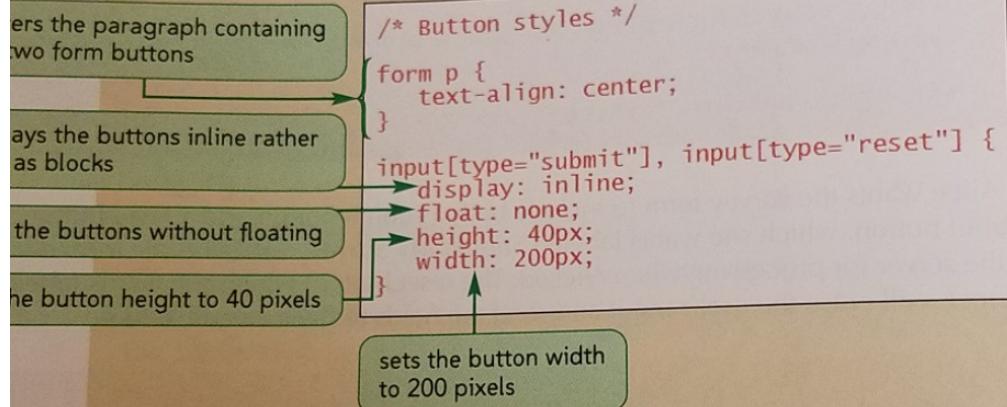
`</p>`

`</form>`

- 3. Save your changes to the file.
Next, you'll create style rules so that the submit and reset buttons appear centered below the rest of the form content.
- 4. Go to the **forms.css** file in your text editor. At the bottom of the file, insert the following code, as shown in Figure 6-64:

```
/* Button styles */  
  
form p {  
    text-align: center;  
}  
  
input[type="submit"], input[type="reset"] {  
    display: inline;  
    float: none;  
    height: 40px;  
    width: 200px;  
}
```

Figure 6-64 Style rules for the submit and reset buttons



- 5. Save your changes to the file and then refresh the **survey.htm** file in your Web browser. Figure 6-65 shows the completed Web page with all of the form elements and controls.

Final layout of the survey page as displayed by Opera



Customer Survey

Thank you for taking our customer survey. Your response helps Red Ball Pizza maintain the tradition that has made us the top-rated pizzeria in the metro area.

All participants are automatically entered into a monthly drawing to receive a Red Ball Express PizzaFest containing two large pizzas, a 2-liter soda, and a side order of chicken wings. Check your e-mail inbox for contest results.

Surveys are private and confidential. Red Ball Pizza will not share your contact information with third parties, ever.

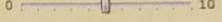
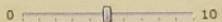


Required values are marked by an asterisk (*)

Customer Information

Name *	<input type="text" value="first and last name"/>
Street address	<input type="text"/>
City	<input type="text" value="Ormond Beach"/>
State (abbr.)	<input type="text" value="FL"/>
Postal code	<input type="text" value="nnnn (-nnnn)"/>
Phone number	<input type="text" value="(nnn) nnn-nnnn"/>
E-mail *	<input type="text"/>
Where did you hear about us? (select all that apply)	<input type="checkbox"/> Internet <input type="checkbox"/> Magazine <input type="checkbox"/> Newspaper <input type="checkbox"/> Word of Mouth <input type="checkbox"/> Other
How many times do you dine out per month?	<input type="text" value="1"/> 
What's your favorite Red Ball dish?	<input type="text"/>

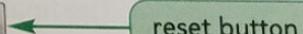
Share Your Experience at Red Ball Pizza

Date of visit	<input type="text"/>
Receipt number *	<input type="text" value="re-nnnnnn"/>
Order type	<input type="text" value="Dine In"/>
Was your service friendly?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Was your order correct?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Was your food hot?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Rate the overall service (0 = poor; 10 = great)	<input type="text" value="0"/> 
Rate the food quality (0 = poor; 10 = great)	<input type="text" value="0"/> 
Tell us more about your experience!	<input type="text"/>

E-mail me your newsletter for great coupons and specials!

 Submit My Survey

Cancel 

 reset button

Red Ball Pizza • 811 Beach Drive • Ormond Beach, FL 32175 • (386) 555 - 7499

Timea/Shutterstock.com
Ariwasabi/Shutterstock.com

- Enter some sample data into the form and then click the **Cancel** button to test the actions of your reset button. Verify that the form is reset to its initial state and default values. You'll test the actions of the submit button shortly.