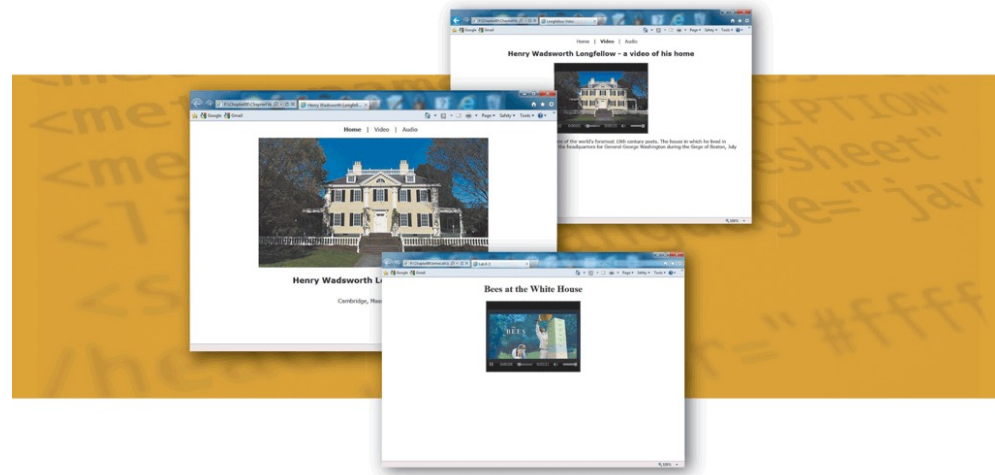# Web Design with

HTML5 & CSS3

8th Edition

Chapter 8

Creating Tables and Forms

Chapter Objectives

- Define table elements

- Describe the steps used to plan, design, and code a table

- Create a table with rows and data

- Insert a table caption

- Style a table for tablet and desktop viewports

- Describe form controls and their uses

Chapter Objectives (continued)

- Use the form and input elements

- Create text input controls, labels, and check boxes

- Create a selection menu with multiple options

- Use the textarea element

- Create a Submit button

- Create a Reset button

Discovering Tables

- Tables
  - Compare data or outline a detailed topic in a compact format
  - Consist of rows, columns, and cells
- **Row** – It is a horizontal line of information
- **Column** – It is a vertical line of information

Discovering Tables (continued 1)

- **Cell**
  - It is the intersection of a row and a column and usually contains data
  - The two types of cell are: a heading cell or a data cell
    - A **heading cell** displays text as bold and center-aligned
    - A **data cell** displays normal, left-aligned text and contains information appropriate for the column and row

Discovering Tables (continued 2)

- Figure 8–4 shows examples of the three elements: rows, columns, and cells
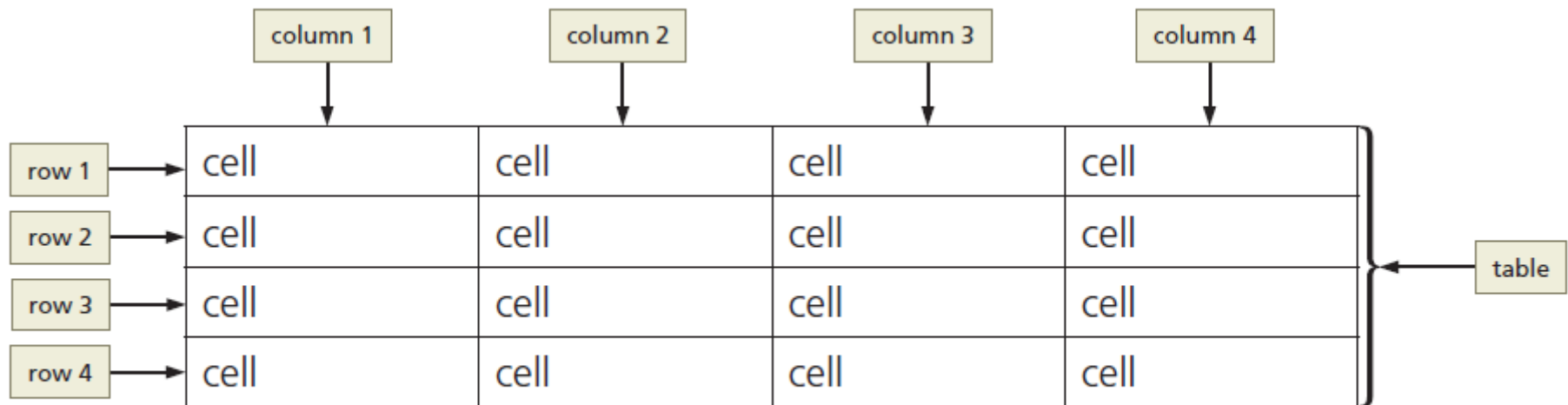
Figure 8–4

Creating a Table with HTML Elements

- Table 8–1 lists the HTML elements used to create a table

Table 8–1 HTML Table Elements

Element	Indicates the start and end of:	Contains:
\<table> … \</table>	Table within a webpage	All related table elements
\<tr> … \</tr>	Table row within a table	Table data cells
\<th> … \</th>	Table header cell	Table header content
\<td> … \</td>	Table data	Table cell content
\<caption> … \</caption>	Table caption	Table caption or title
\<thead> … \</thead >	Table header area	Grouped header content
\<tbody> … \</tbody >	Table body area	Grouped body content
\<tfooter> … \</tfooter >	Table footer area	Grouped footer content

Creating a Table with HTML Elements (continued)

- Creating a table on a webpage
  - The <table> and </table> tags indicate the starting and ending of a table
  - The <tr> and </tr> tags indicate the starting and ending of each table row
  - The <td> and </td> tags indicate the starting and ending tags for data elements within the table row elements

Table Borders, Headers, and Captions

- **Table border** – It is the line that defines the perimeter of the table

- **Table header**

  - It is a heading cell and identifies the row or column content

  - It is defined with a starting **<th>** tag and an ending **</**th**>** tag

Table Borders, Headers, and Captions (continued 1)

- **Table caption**
  - It is a descriptive text that serves as a title or identifies the table's purpose
  - It is defined with a starting <caption> tag and an ending </caption> tag
  - It is inserted after the starting <table> tag
  - A table can have only one caption

- Tables can include headers and captions individually or in combination

Table Borders, Headers, and Captions (continued 2)

- Figure 8–6 shows a table created with the following code, which includes a table caption and table headers

```
<table>
   <caption>College Course
   Recommendations by Semester</caption>
   <tr>
       <th>Semester 1</th>
       <th>Semester 2</th>
       <th>Semester 3</th>
       <th>Semester 4</th>
   </tr>
```

Table Borders, Headers, and Captions (continued 3)

```
<tr>
      <td>English I</td>
      <td>English II</td>
      <td>Spanish I</td>
      <td>Spanish II</td>
</tr>
<tr>
      <td>College Algebra</td>
      <td>College Geometry</td>
      <td>Calculus</td>
      <td>Trigonometry</td>
```

Table Borders, Headers, and Captions (continued 4)

```
    </tr>
    <tr>
        <td>Physical Science</td>
        <td>Biology</td>
        <td>Humanities</td>
        <td>World History</td>
    </tr>
</table>
```

Table Borders, Headers, and Captions (continued 5)

Figure 8–6

Use of Tables

- Tables
  - Display data in rows and columns
  - Should not be used to design a layout for a webpage
  - Help organize information so that it is easier for the user to read
  - Useful if the webpage needs to display a structured, organized list of information

Planning the Table

- To create effective tables
  - Plan the information that will appear in columns and rows
  - Create a design that presents the information clearly
- When planning a table for responsive web design, give careful consideration to the mobile viewport because of its screen size
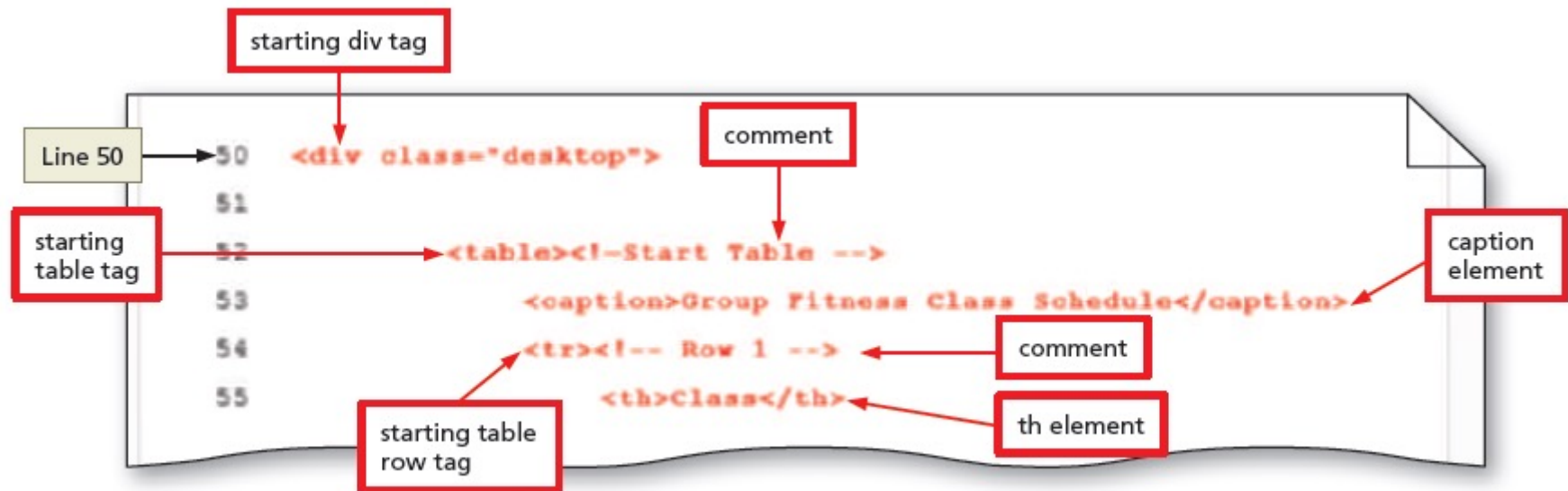
To Add a Table Element to the Classes Page

Figure 8–11

Styling Table Elements

Table 8–2 Common CSS Table Properties		
Property	**Example**	**Explanation**

background-color	th {
	    background-color: #F5F5F0;
	}
		Displays the table header with a light gray background

border	table, th, td {
	    border: 0.1em solid #000000;
	}
		Displays the table, table header, and data cells with a thin solid black border

border-collapse	table {
	    border-collapse: collapse;
	}
		Collapses borders in the table so that adjacent cells share borders

color	caption {
	    color: #003300;
	}
		Displays the table caption in dark green text

height	td {
	    height: 2em;
	}
		Sets the height of a table data cell to 2em

margin	table {
	    margin-top: 2em;
	}
		Applies a 2em top margin to the table

padding	caption, th, td {
	    padding: 1em;
	}
		Applies 1em of padding to the table caption, header, and data cells

text-align	td {
	    text-align: center;
	}
		Aligns the table data in the center of the cell

vertical-align	td {
	    vertical-align: center;
	}
		Aligns the table data vertically in the middle of the cell

width	table {
	    width: 80%;
	}
		Sets the width of the table to 80% of the page width

Styling Table Elements (continued)

- Separated border
  - When a border is applied to table elements, by default, each cell has its own border, making the table appear to use double lines between each table data cell

- Collapsed border
  - Use the `border-collapse` property with a value of `collapse` to display a table with single, consolidated borders
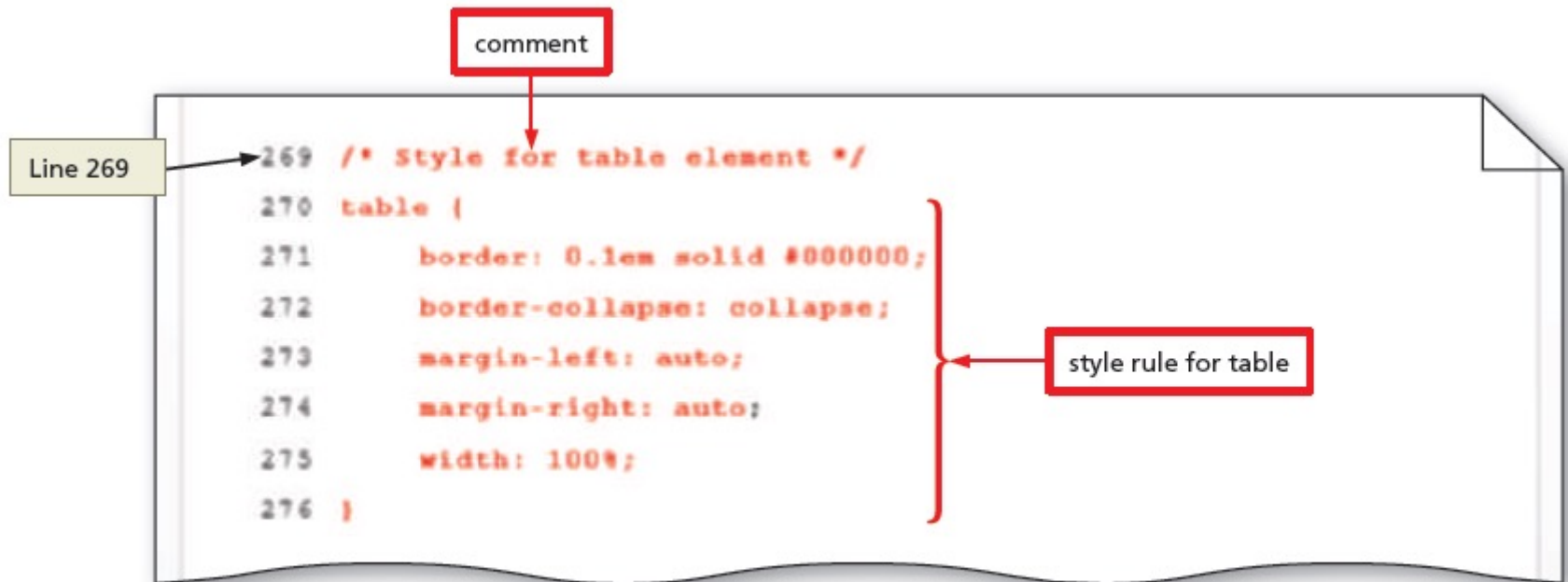
To Style a Table for a Tablet Viewport

Figure 8–22

comment

Line 269

style rule for table

```
269  /* Style for table element */
270  table {
271       border: 0.1em solid #000000;
272       border-collapse: collapse;
273       margin-left: auto;
274       margin-right: auto;
275       width: 100%;
276  }
```

To Style a Table for a Tablet Viewport (continued 1)

Figure 8–23

© Rob Marmion/Shutterstock.com

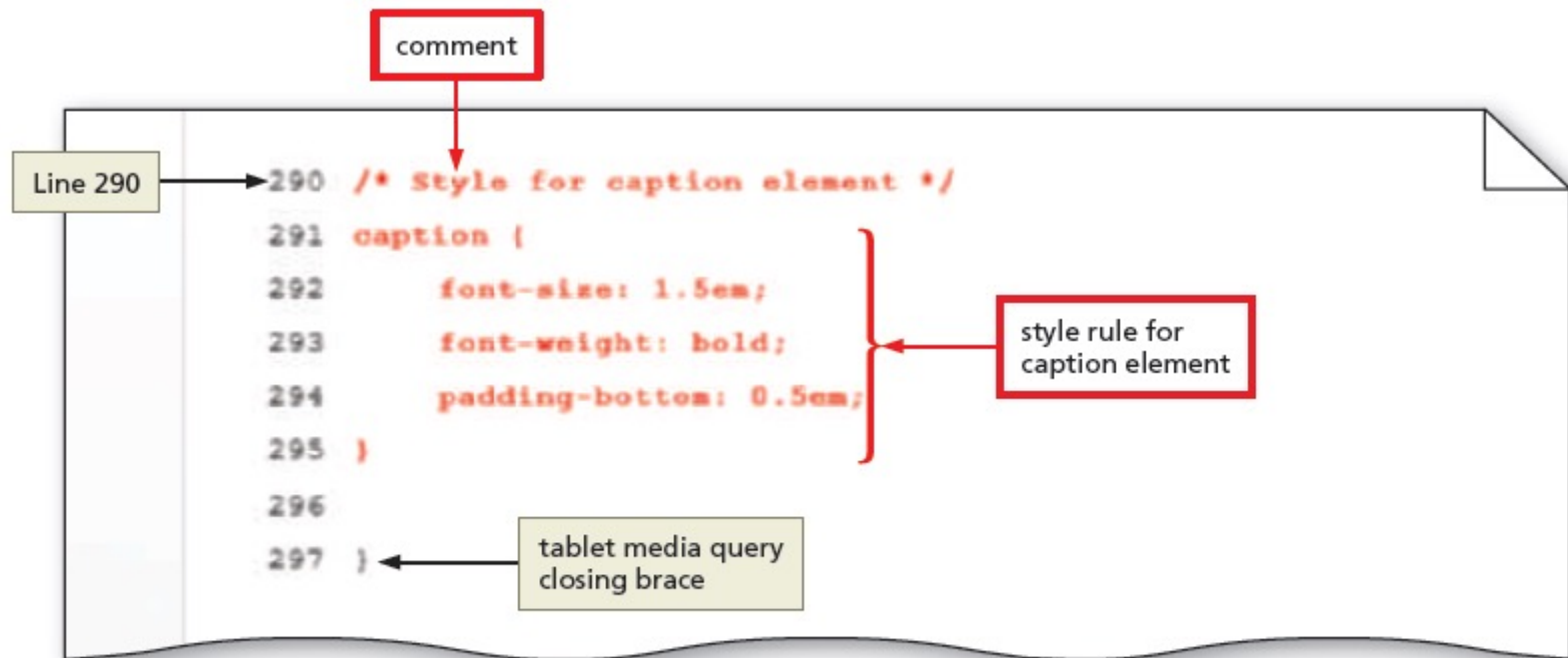# To Style a Table for a Tablet Viewport (continued 2)

Figure 8–28

To Style a Table for a Tablet Viewport (continued 3)

formatting applied to caption element

Group Fitness Class Schedule

Class	Days	Times	Instructor	Room
Cardio	Mon, Wed, Fri	6:00am, 6:00pm	Schultz	B
Boot Camp	Tue, Thu	5:00am, 5:00pm	Taylor	B
Spinning	Tue, Thu	6:00am, 6:00pm	Roberts	A
Kickboxing	Mon, Wed, Fri	8:00am, 7:15pm	Lawrence	A
Yoga	Tue, Thu	6:00am, 6:00pm	Schultz	B
Zumba	Mon, Wed, Fri	7:00am, 6:00pm	Roberts	A

Figure 8–29

To Style a Table for a Desktop Viewport
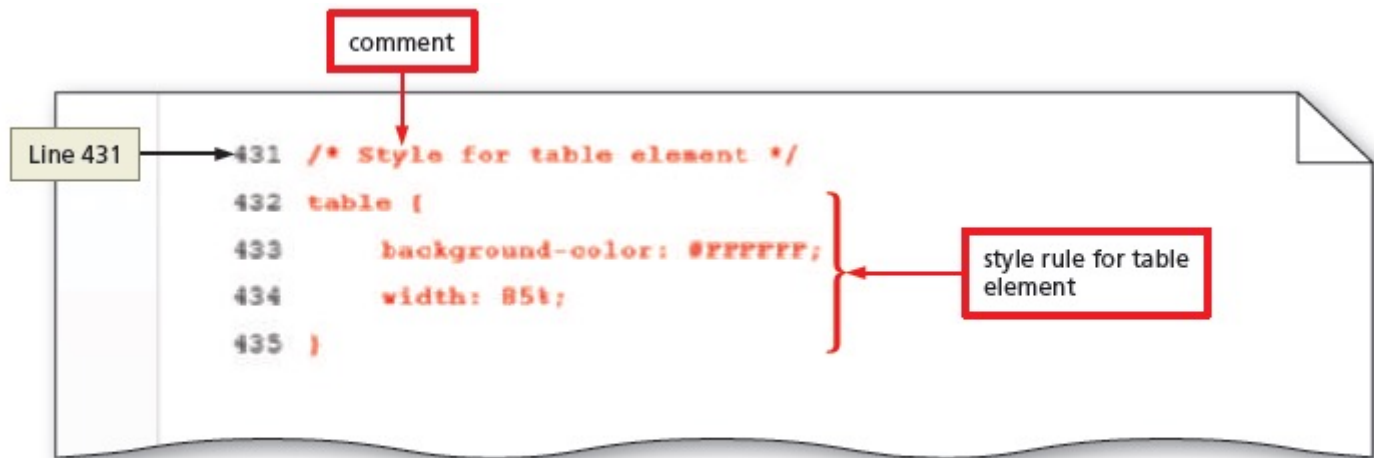
Figure 8–30

To Style a Table for a Desktop Viewport (continued 1)

comment

Line 437

437 /* Style for caption element */
438 caption {
439   color: #FFFFFF;
440 }
441
442 }

style rule for caption element

desktop media query closing brace
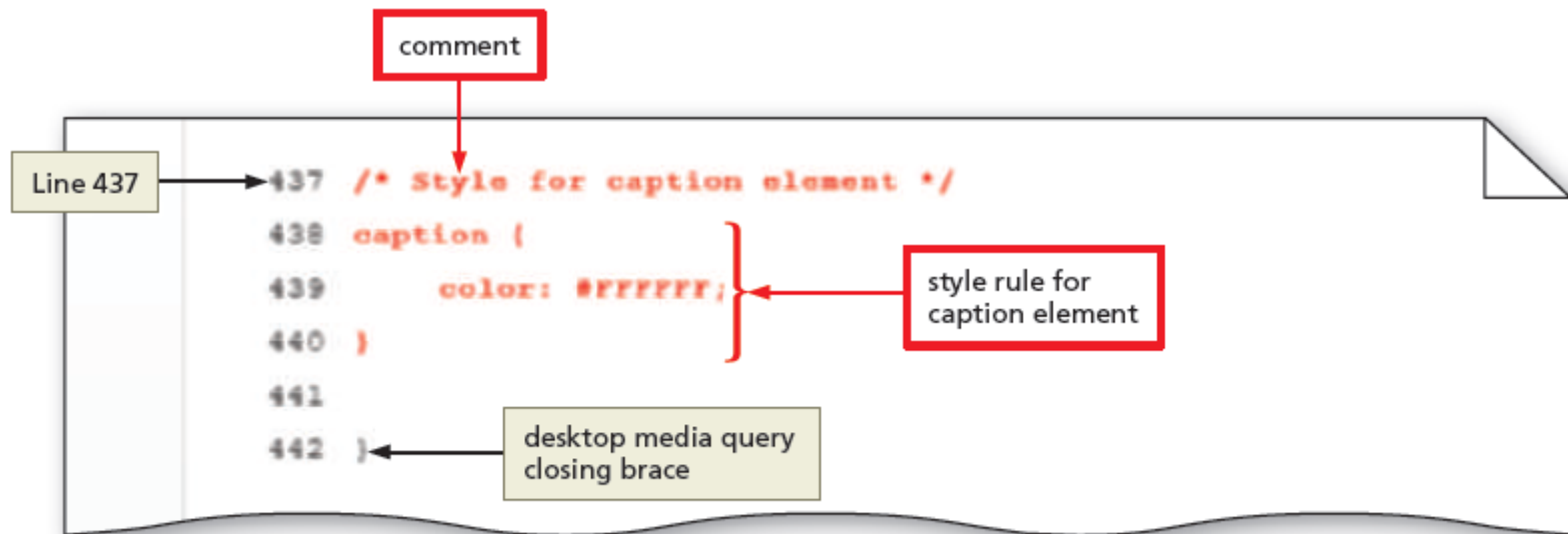
Figure 8–31

To Style a Table for a Desktop Viewport (continued 2)

Figure 8–32

Creating Webpage Forms

- Forms provide a structured way to collect information from webpage visitors

- Visitors complete webpage forms to register for an account or to make a purchase.

- Businesses use forms to gather visitor or customer information and store it in a database for future use

Form Controls

- All form elements are contained with the starting <form> tag and the ending </form> tag

- **Input control** – It is an interactive mechanism for users to enter text or make selections on a form
  - A label is text describing the type of information to enter with an input control.

- Most controls in an HTML form are defined by using the `type` attribute of the `input` element

Form Controls (continued 1)

- Define other controls using separate elements, such as the `textarea` and `select` elements

- The two input controls are:

  - **Data input control** – It is used to make a selection or perform a command

  - **Text input control** – It accepts text, such as names, dates, and passwords, and is called an input field

Form Controls (continued 2)

Table 8–3 Input Types

Input Type	Description	Code Example
button	Creates a button; typically used to run a script when clicked	`<input type="button" onclick="alert('Good Morning!')" value="My Button">`
checkbox	Creates a single item or a list of items	`<input type="checkbox" name="fruit" value="banana">`
date	Creates an input field used to contain a date; the field may appear as a date picker, depending on the browser	`<input type="date" name="birthday">`
datetime	Creates an input field for a date and time with a time zone	`<input type="datetime" name="bdaydatetime">`
datetime-local	Creates an input field for a date and time without a time zone	`<input type="datetime-local" name="bddatetime">`
email	Creates an input field for an email address	`<input type="email" name="email" id="email">`
file	Creates a file-select field and a Browse button	`<input type="file" name="doc">`
hidden	Creates a control that is hidden from the user but contains information to process the form	`<input type="hidden" name="ship">`
image	Creates a graphical button instead of the default button	`<input type="image" name="reset" src="reset.png" alt="Reset">`
month	Creates an input field for a month and year; the field may appear as a date picker, depending on the browser	`<input type="month" name="bdaymth">`
number	Creates an input field for a numeric value	`<input type="number" name="cost">`
password	Creates a single-line field for a relatively small amount of text and masks the entered text as asterisks or bullets	`<input type="password" name="pw" id="pw">`

radio	Creates a list item	`<input type="radio" name="state" value="AL">`
		`<input type="radio" name="state" value="AK">`
		`<input type="radio" name="state" value="AZ">`

range	Creates an input field for a value within a range; the field may appear as a slider control, depending on the browser	`<input type="range" name="survey" min="0" max="10">`
reset	Resets the form	`<input type="reset" value="Reset Form">`
search	Creates an input field used as a search field	`<input type="search" name="search">`
submit	Submits a form for processing	`<input type="submit" value="Submit Form">`
tel	Creates an input field for a telephone number	`<input type="tel" name="phone" id="phone">`
text	Creates a single-line field for text	`<input type="text" name="fName" id="fName">`
time	Creates an input field for a time without a time zone; the field may appear as a time picker, depending on the browser	`<input type="time" name="time">`
url	Creates an input field for a URL	`<input type="url" name="page">`
week	Creates an input field for a week and year; the field may appear as a date picker, depending on the browser	`<input type="week" name="week">`

Form Controls (continued 3)

- Each input control has attributes that are used more frequently than the others:

  - **name** – It identifies the specific information that is being sent when the form is submitted for processing. All controls have a name

  - **id** – It provides a unique ID for the element. Use the `id` attribute with input controls

  - **value** – It specifies the value of an `input` element and varies depending on input type

Form Controls (continued 4)

- Common input controls used with a form include text, password, email, checkbox, select, submit, reset, etc.

- Text input controls include the following types:
  - **text box** (text control), for small amounts of text
  - **password text box** (password control), for entering a password
  - **email text box** (email control), for entering an email address

Form Controls (continued 5)

- **telephone text box** (tel control), for entering a telephone number
- **date text box** (date control), for entering a date
- **text area box** (textarea control), for larger amounts of text

- These text input controls have two frequently used attributes:
  - **size** – It determines the width of the control in characters
  - **maxlength** – It specifies the maximum number of characters accepted

Form Controls (continued 6)

– Example:

o The first line of the following code creates a 25-character text box for the user's last name and the second line creates an eight-character text box for the user's password:

```
<p>Last Name: <input
name="lastname" type="text"
size="25"></p>
<p>Password: <input name="password
type="password" size="8"></p>
```

Form Controls (continued 7)

- **Password control** – It is a text control as it provides a text box for the password a visitor enters

- **Email control** – It is a text box where visitors enter an email address

- **Tel control** – It is a text box where visitors enter a telephone number

- **Date control** – It is a text box that accepts a date

Form Controls (continued 8)

- **Textarea control** – It creates a text box that allows multiple lines of input
  - To create a textarea control, use the `textarea` element instead of the `input` element
  - It has two primary attributes, which set the size of the textarea control:
    - o **rows**, which specifies the number of rows, or lines, in the textarea control
    - o **cols**, which sets the width of the textarea control as the number of columns, with each column containing one character

Form Controls (continued 9)

o The following is an example of HTML code defining a textarea control:

```
<label>What products would you like
to see us offer?</label>
<textarea name="feedback" rows="3"
cols="100"></textarea>
```

Form Controls (continued 10)

- **Checkbox control**
  - It allows a webpage visitor to select items from a list of one or more choices
  - The following code is an example for two checkbox controls that might appear on a form for a grocery store website

    ```
    <input name="fruit" type="checkbox"
    value="apple" checked="checked">Apple
    <input name="fruit" type="checkbox"
    value="peach">Peach
    ```

Form Controls (continued 11)

- **Radio control**
  - It limits the webpage visitor to only one choice from a list of choices
  - Each choice has a **radio button**, or option button, which appears as an open circle
  - By default, all radio buttons are deselected
  - To set a particular button as the default, use the `checked` attribute and value within the <input> tag

Form Controls (continued 12)

- The following is sample code to create two radio controls that might appear in a rental car website form:

```
<input name="car" type="radio"
checked="checked"  value="car">Car

<input name="truck" type="radio"
value="truck">Truck
```

- A visitor can choose to rent a car or a truck, with the Car radio button already selected when the form opens

Form Controls (continued 13)

- **Select control**
  - It creates a selection menu from which the visitor makes one or more choices and is suitable when a limited number of choices are available
  - It appears on a form as a text box with a list arrow
  - The following is sample code for a select control:

```
<select name="station">
      <option>Pandora</option>
      <option>Internet Radio</option>
      <option>Live365</option>
      <option>Jango</option>
</select>
```

Form Controls (continued 14)

- **Submit button**
  - It sends the form information to the appropriate location for processing
  - When it is clicked on the form, the name of each control and the value of its data are sent to the server to be processed
  - The submit control is created with the following code:

    ```
    <input type="submit" value="Submit">
    ```
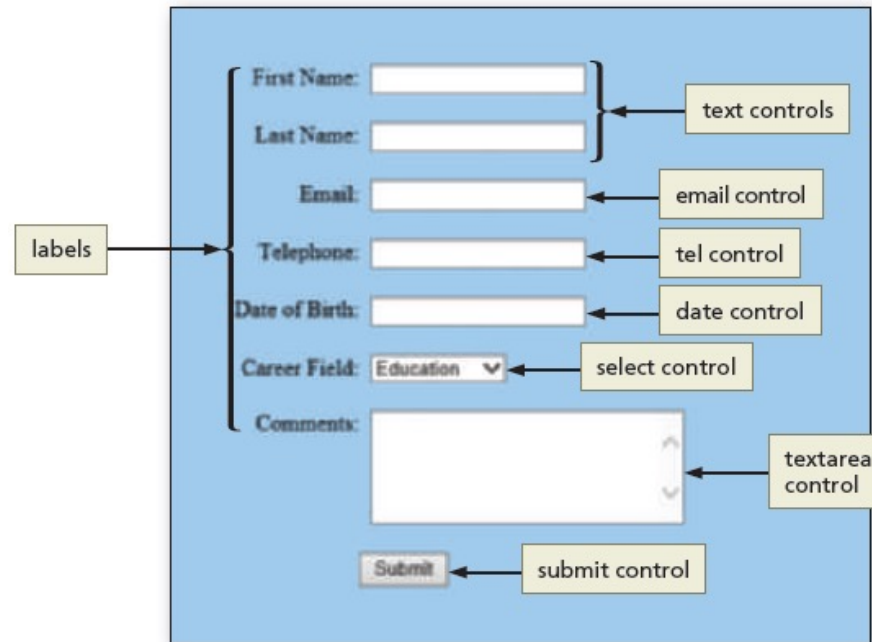
Form Controls (continued 15)

- **Reset button**
  - Clears any input entered in the form, resetting the input controls to their defaults
  - A webpage form must include a submit control and a reset control
  - The `value` attribute specifies the text that appears on the button
  - The reset control is created with the following code:

    ```
    <input type="reset" value="Reset">
    ```

Form Controls (continued 16)

- Figure 8–34 shows an example of a form with several input controls, including text, email, tel, date, select, and textarea

Figure 8–34

Form Controls (continued 17)

- **Form Labels**
  - They identify the type of information to enter into or select from an input control
  - They are added to a form using the `label` element
  - To connect them to their controls, include the `for` attribute with the same value as the input control's `id` value

Form Controls (continued 18)

- The following code creates a label and a text box for a visitor's first name:

```
<label for="fName">First Name:</label>
<input type="text" name="fName"
id="fName">
```

Form Controls (continued 19)

Table 8–4 Common Form Attributes

Attribute	Description	Code Example
accept-charset	Specifies the character set used for the form submission	`<form accept-charset="UTF-8">`
action	Specifies where to submit the form data (a URL)	`<form method="POST" action="form.php">`
autocomplete	Specifies whether a form or input field should use autocomplete; when enabled, the browser can complete input fields based on information entered in previous input fields	`<form autocomplete="on">`
autofocus	Specifies that an input field should have focus when the page is displayed, which places the insertion point within a specific input field	`<input type="text" name="fName" id="fName" autofocus>`
disabled	Specifies that the input field is disabled and not available for user input	`<input type="radio" name="terms" value="Accept" disabled>`
enctype	Specifies the encoding of the form for submitting data.	`<form enctype="app/urlencoded">`
form	Specifies which form an input field belongs to when multiple forms are used within a website	`<input type="text" name="fName" id="fName" form="form1">`
formaction	Specifies the URL of a file that will process the input control when the form is submitted, overriding the form action attribute; use the formaction attribute with type="submit" and type="image" input types	`<input type="submit" value="Submit" formaction="process.asp">`
formenctype	Specifies how to encode form data during form submission; use the formenctype attribute with type="submit" and type="image" input types	`<input type="submit" formenctype="multipart/form-data">`
formmethod	Specifies the HTTP method used to transfer the form data, overriding the method attribute	`<input type="submit" value="Submit" formmethod="post">`
formnovalidate	Specifies to not validate an input element	`<input type="submit" value="Submit" formnovalidate>`
formtarget	Specifies a keyword that determines how to display a response when the form is submitted, such as a new, blank window	`<input type="submit" value="Submit" formtaget="_blank">`

Form Controls (continued 20)

Table 8–4 (*continued*)

Attribute	Description	Code Example
height and width	Specifies the height and width for an image input type; always specify a height and width for the image input type	`<input type="image" src="btn1" alt="button 1" height="25" width="30">`
list	Used with the datalist element to specify predefined options for an input element	`<input list="music">` `<datalist id="music">` `<option value="Country">` `<option value="Classical">` `<option value="Hip Hop">` `<option value="Rock">` `</datalist>`
max and min	Specifies the maximum and minimum values for an input element	`<input type="number" name="survey" min="1" max="10">`
maxlength	Specifies the maximum number of characters allowed within the input field	`<input type="text" name="fName" id="fName" maxlength="15">`
method	Specifies the HTTP method used to submit the form data	`<form method="POST">`
multiple	Specifies that user may input more than one value within an input element; used with email and file input types	`<input type="email" name="email" id="email" multiple>`
novalidate	A form attribute that specifies not to validate form data when the submit button is clicked	`<form novalidate>`
pattern	Specifies a regular expression for checking an input element value; used with text, search, url, tel, email, and password input types	`<input type="password" name="pw" id="pw" pattern="[A-Za-z]{8}">`
placeholder	Specifies a hint of the type of information expected within an input field	`<input type="email" name="email" id="email" placeholder="youremail@domain.com">`
readonly	Specifies that the input field is a read-only field and cannot be modified	`<input type="text" name="ssn" id="ssn" value="555123654" readonly>`
required	Specifies that an input field is required	`<input type="text" name="fName" id="fName" required>`
size	Specifies the size (length) of an input field	`<input type="text" name="fName" id="fName" size="20">`
step	Specifies the legal number intervals for an `<input>` element	`<input type="number" name="math" step="5">`
target	Specifies the target address in the action attribute	`<form target="_blank">`
value	Specifies the value for an input field.	`<input type="text" name="fName" id="fName" value="Kate">`

Form Processing

- `action`
  - It is an attribute of the <form> tag that specifies the browser's action when submitting the form
- **Common Gateway Interface (CGI) script**
  - It communicates with the web server and sends the information on the webpage form to the server for processing

Form Processing (continued 1)

- `method`

  - It is an attribute of the <form> tag that specifies how to send the data entered in the form to the server to be processed

- **Get method**

  - It appends the name-value pairs to the URL indicated in the action attribute

  - Example of a form tag with the get method and specified action is as follows:

  ```
  <form method="GET" action="formInfo.php">
  ```

Form Processing (continued 2)

- **Post method**
  - It sends a separate data file with the name-value pairs to the URL indicated in the action attribute
  - It is used very commonly because it can be used to send sensitive form data and does not have a size limitation
  - Example of a form tag with the post method and specified action is as follows:

```
<form method="POST" action="formInfo.php">
```

Styling Forms

- CSS styles are applied to forms to improve the appeal and usefulness of the form and its controls

- As with tables, consider forms in the context of responsive design

- Controls can be included to collect optional information, such as product feedback, in tablet and desktop viewports
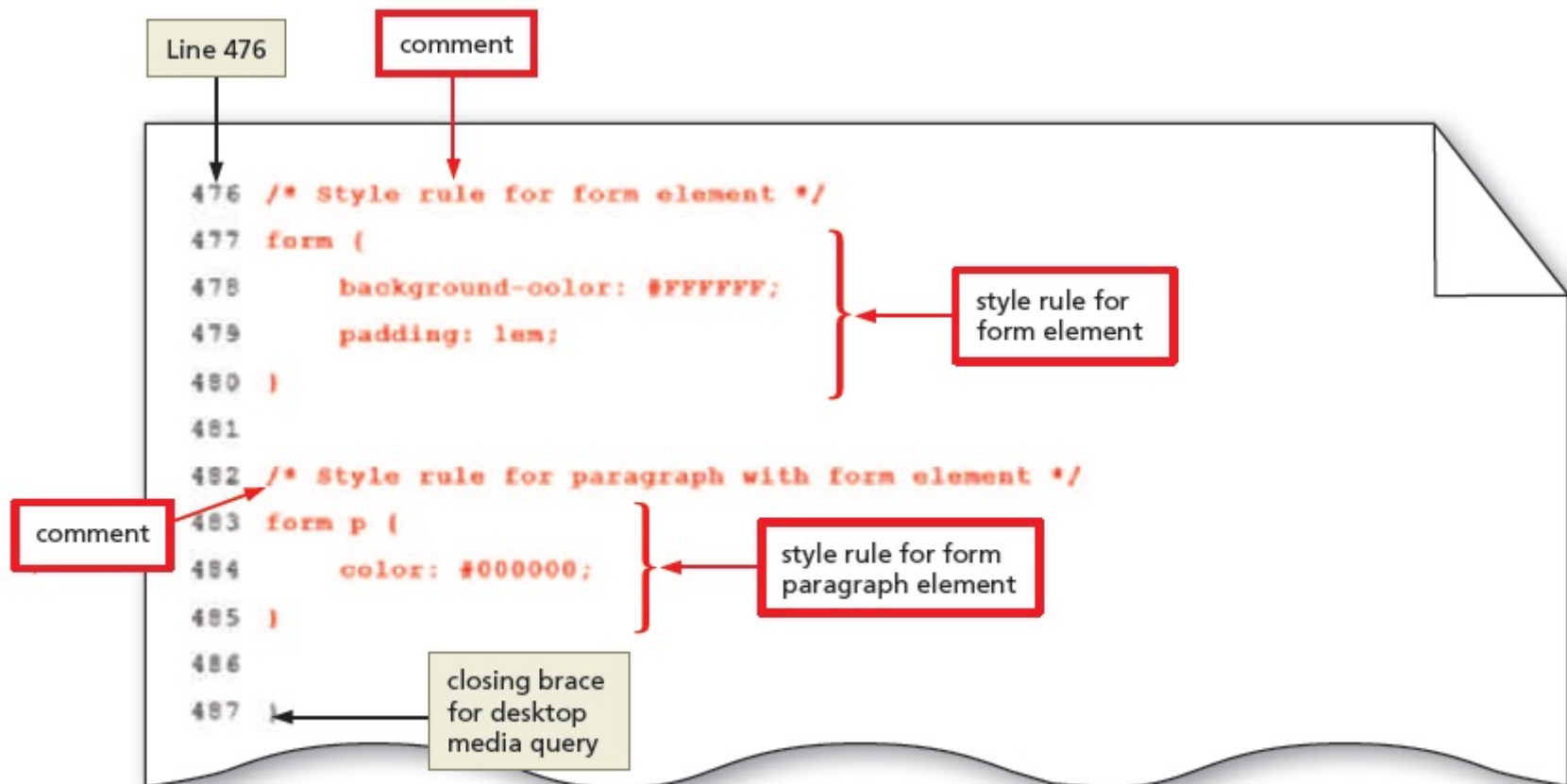
To Style a Form for a Desktop Viewport

Figure 8–58

To Style a Form for a Desktop Viewport (continued)

Figure 8–59

HTML5 & CSS3
8ᵗʰ Edition

Chapter 8 Complete