

Chapter 3

Introduction to HTML5: Part 2

INTERNET & WORLD WIDE WEB
HOW TO PROGRAM, 5/E

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

Objectives

- Build a form using the new HTML input types
- Specify an input element in a form as the one that should receive the focus by default
- Use self-validating input elements.
- Specify temporary placeholder text in various input elements
- Use autocomplete input elements that help users re-enter text that they've previously entered in a form.
- Use a datalist to specify a list of values that can be entered in an input element and to autocomplete entries as the user types.
- Use HTML5's new page structure elements to delineate parts of a page. including headers, sections, figures, articles, footers and more.

DR. PARAVASTU

3.1 Introduction**3.2** New HTML5 Form input Types

- 3.2.1 input Type color
- 3.2.2 input Type date
- 3.2.3 input Type datetime
- 3.2.4 input Type datetime-local
- 3.2.5 input Type email
- 3.2.6 input Type month
- 3.2.7 input Type number
- 3.2.8 input Type range
- 3.2.9 input Type search
- 3.2.10 input Type tel
- 3.2.11 input Type time
- 3.2.12 input Type url
- 3.2.13 input Type week

3.3 input and dataList Elements and autocomplete Attribute

- 3.3.1 input Element autocomplete Attribute
- 3.3.2 dataList Element

3.3 Page-Structure Elements

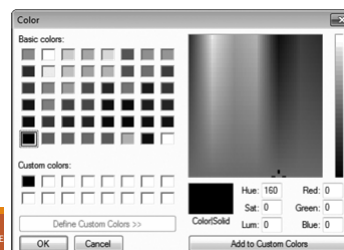
- 3.4.1 header Element
- 3.4.2 nav Element
- 3.4.3 figure Element and figcaption Element
- 3.4.4 article Element
- 3.4.5 summary Element and details Element
- 3.4.6 section Element
- 3.4.7 aside Element
- 3.4.8 meter Element
- 3.4.9 footer Element
- 3.4.10 Text-Level Semantics: mark Element and wbr Element

- ▶ These are not yet universally supported by all browsers.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type color

- The **color input type** enables the user to enter a color.
- At the time of this writing, most browsers render the **color input type** as a text field in which the user can enter a hexadecimal code or a color name.
- In the future, when you click a **color input**, browsers will likely display a *color picker* similar to the Microsoft Windows color dialog

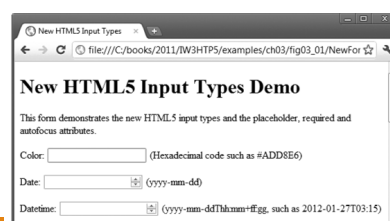


COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type color

autofocus Attribute

- The **autofocus attribute**—an optional attribute that can be used in only one **input** element on a form—automatically gives the focus to the **input** element, allowing the user to begin typing in that element immediately.
- Figure 3.3 shows **autofocus** on the **color** element—the first **input** element in our form—as rendered in Chrome. You do not need to include **autofocus** in your forms.



COPYRIGHT © PEARSON, INC., 2013.

3 | Autofocus in the color input element using Chrome.

input Type color

Validation

- The new **HTML 5** **input** types are *self validating* on the client side, eliminating the need to add complicated JavaScript code to your web pages to validate user input, reducing the amount of invalid data submitted and consequently reducing Internet traffic between the server and the client to correct invalid input.
- *The server should still validate all user input.*
- When a user enters data into a form then submits the form the browser immediately checks the self-validating elements to ensure that the data is correct (Fig. 3.4).

COPYRIGHT © PEARSON, INC., 2013. ALL RIGHTS RESERVED.

input Type color

- Figure 3.5 lists each of the new HTML5 input types and provides examples of the proper formats required for each type of data to be valid.

input type	Format
color	Hexadecimal code
date	yyyy-mm-dd
datetime	yyyy-mm-dd
datetime-local	yyyy-mm-ddThh:mm
month	yyyy-mm
number	Any numerical value
email	name@domain.com
url	http://www.domain-name.com
time	hh:mm
week	yyyy-Wnn

- To bypass validation, add the `formnovalidate` attribute to **input type submit**

```
<input type = "submit" value  
= "Submit" formnovalidate />
```

Fig. 3.5 | Self-validating input types.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type date

- The `date input type` enables the user to enter a date in the form `yyyy-mm-dd`.
- Firefox and Internet Explorer display a text field in which a user can enter a date such as `2012-01-27`.
- Chrome and Safari display a `spinner control`—a text field with an up-down arrow () on the right side—allowing the user to select a date by clicking the up or down arrow.
- The start date is the *current date*.
- Opera displays a calendar from which you can choose a date.
- In the future, when the user clicks a `date input`, browsers are likely to display a date control similar to the Microsoft Windows one shown in Fig. 3.6.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type datetime

- The **datetime input type** enables the user to enter a date (year, month, day), time (hour, minute, second, fraction of a second) and the time zone set to UTC (Coordinated Universal Time or Universal Time, Coordinated).
- Currently, most of the browsers render **datetime** as a text field; Chrome renders an up-down control and Opera renders a date and time control.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type datetime-local

- The **datetime-local input type** enables the user to enter the date and time in a *single* control.
- The data is entered as year, month, day, hour, minute, second and fraction of a second.
- Internet Explorer, Firefox and Safari all display a text field.
- Opera displays a date and time control.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type email

- The **email input type** enables the user to enter an e-mail address or a list of e-mail addresses separated by commas (if the **multiple** attribute is specified).
- Currently, all of the browsers display a text field.
- If the user enters an *invalid* e-mail address (i.e., the text entered is *not* in the proper format) and clicks the Submit button, a callout asking the user to enter an e-mail address is rendered pointing to the **input** element (Fig. 3.7).
- HTML5 does not check whether an e-mail address entered by the user actually exists—rather it just validates that the e-mail address is in the *proper format*.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type email

placeholder Attribute

- The **placeholder attribute** allows you to place temporary text in a text field.
- Generally, **placeholder** text is *light gray* and provides an example of the text and/or text format the user should enter (Fig. 3.8).
- When the *focus* is placed in the text field (i.e., the cursor is in the text field), the **placeholder** text disappears—it's not “submitted” when the user clicks the Submit button (unless the user types the same text).

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type email

- HTML5 supports placeholder text for only six input types—text, search, url, tel, email and password.

requiredAttribute

- The **required attribute** forces the user to enter a value before submitting the form.
- You can add **required** to any of the input types.
- In this example, the user *must* enter an e-mail address and a telephone number to submit the form (Fig. 3.9).

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type month

- The **month input type** enables the user to enter a year and month in the format yyyy-mm, such as 2012-01.
- If the user enters the data in an improper format (e.g., January 2012) and submits the form, a callout stating that an invalid value was entered appears.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type number

- The **number input type** enables the user to enter a numerical value—mobile browsers typically display a numeric keypad for this **input** type.
- Internet Explorer, Firefox and Safari display a text field in which the user can enter a number. Chrome and Opera render a spinner control for adjusting the number.
- The **min** attribute sets the minimum valid number.
- The **max** attribute sets the maximum valid number.
- The **step** attribute determines the increment in which the numbers increase.
- The **value** attribute sets the initial value displayed in the form (Fig. 3.10).
- The spinner control includes only the valid numbers.
- If the user attempts to enter an invalid value by typing in the text field, a callout pointing to the **number input** element will instruct the user to enter a valid value.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type range

- The **range input type** appears as a *slider* control in Chrome, Safari and Opera (Fig. 3.12).
- You can set the minimum and maximum and specify a value.
- The **range input type** is *inherently self-validating* when it is rendered by the browser as a slider control, because *the user is unable to move the slider outside the bounds of the minimum or maximum value*.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type search

- The `search input type` provides a search field for entering a query.
- This `input` element is functionally equivalent to an `input` of type `text`.
- When the user begins to type in the search field, Chrome and Safari display an X that can be clicked to clear the field (Fig. 3.13).
-

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type tel

- The `tel input type` enables the user to enter a telephone number—mobile browsers typically display a keypad specific to entering phone numbers for this `input` type.
- At the time of this writing, the `tel input type` is rendered as a text field in all of the browsers.
- HTML5 does *not* self validate the `tel input type`.
- To ensure that the user enters a phone number in a proper format, we've added a `pattern` attribute that uses a *regular expression* to determine whether the number is in the format:
 - (555) 555-5555
- When the user enters a phone number in the wrong format, a callout appears requesting the proper format, pointing to the `tel input` element (Fig. 3.14).
-

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type time

- The **time input type** enables the user to enter an hour, minute, seconds and fraction of second (Fig. 3.15).
- The HTML5 specification indicates that a time must have two digits representing the hour, followed by a colon (:) and two digits representing the minute.
- Optionally, you can also include a colon followed by two digits representing the seconds and a period followed by one or more digits representing a fraction of a second (shown as `ff` in our sample text to the right of the time input element in Fig. 3.15).
-

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type url

- The **url input type** enables the user to enter a URL.
- The element is rendered as a text field, and the proper format is `http://www.deitel.com`.
- If the user enters an improperly formatted URL (e.g., `www.deitel.com` or `www.deitelcom`), the URL will *not* validate (Fig. 3.16).
- HTML5 does not check whether the URL entered is valid; rather it validates that the URL entered is in the proper format.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Type week

- The **week input type** enables the user to select a year and week number in the format `yyyy-wnn`, where `nn` is 01–53—for example, `2012-w01` represents the first week of 2012. Internet Explorer, Firefox and Safari render a text field.
- Chrome renders an up-down control.
- Opera renders *week control* with a down arrow that, when clicked, brings up a calendar for the current month with the corresponding week numbers listed down the left side.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input and dataList Elements and autocomplete Attribute

- Figure 3.17 shows how to use the new **autocomplete** attribute and **dataList** element.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

input Element autocomplete Attribute

- The **autocomplete attribute** can be used on **input** types to automatically fill in the user's information based on previous input—such as name, address or e-mail.
- You can enable **autocomplete** for an entire form or just for specific elements.
- For example, an online order form might set **autocomplete = "on"** for the name and address **inputs** and set **autocomplete = "off"** for the credit card and password **inputs** for security purposes.
- **Autocomplete attribute works only if you specify a name or id attribute for the input element**

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

datalist Element

- The **datalist element** provides input options for a **text input** element.
- At the time of this writing, **datalist** support varies by browser.
- In this example, we use a **datalist** element to obtain the user's birth month.
- Using Opera, when the user clicks in the text field, a drop-down list of the months of the year appears. If the user types "M" in the text field, the list on months is narrowed to March and May.
- When using Firefox, the drop-down list of months appears only after the user begins typing in the text field. If the user types "M", all months containing the letter "M" or "m" appear in the drop-down list—March, May, September, November and December.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

Page-Structure Elements

- HTML5 introduces several new page-structure elements (Fig. 3.18) that meaningfully identify areas of the page as headers, footers, articles, navigation areas, asides, figures and more.

-

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

header Element

- The **header element** creates a header for this page that contains both text and graphics.
- The **header** element can be used multiple times on a page and can include HTML headings (<h1> through <h6>), navigation, images and logos and more.

time Element

- The **time element**, which does not need to be enclosed in a **header**, enables you to identify a date (as we do here), a time or both.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

nav Element

- The **nav element** groups navigation links.
- In this example, we used the heading Recent Publications and created a **ul** element with seven **li** elements that link to the corresponding web pages for each book.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

figure Element and figcaption Element

- The **figure element** describes a figure (such as an image, chart or table) in the document so that it could be moved to the side of the page or to another page.
- The **figcaption element** provides a caption for the image in the **figure** element.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

article Element

- The **article** element describes standalone content that could potentially be used or distributed elsewhere, such as a news article, forum post or blog entry.
- You can nest **article** elements. For example, you might have reader comments about a magazine nested as an **article** within the magazine **article**.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

summary Element and details Element

- The **summary element** displays a right-pointing arrow next to a summary or caption when the document is rendered in a browser (Fig. 3.19).
- When clicked, the arrow points downward and reveals the content in the **details element**.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

section Element

- The **section element** describes a section of a document, usually with a heading for each section—these elements can be nested.
- In this example, we broke the document into three **sections**—the first is Recent Publications.
- The **section** element may also be nested in an article.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

aside Element

- The **aside element** describes content that's related to the surrounding content (such as an **article**) but is somewhat separate from the flow of the text.
- For example, an **aside** in a news story might include some background history.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

meter Element

- The **meter element** renders a visual representation of a measure within a range (Fig. 3.20).
- In this example, we show the results of a recent web survey we did.
- The `min` attribute is "0" and a `max` attribute is "54" —indicating the total number of responses to our survey.
- The `value` attribute is "14", representing the total number of people who responded "yes" to our survey question.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

footer Element

- The **footer element** describes a *footer*—content that usually appears at the bottom of the content or **section** element.
- In this example, we use the **footer** to describe the copyright notice and contact information.

COPYRIGHT © PEARSON, INC. 2013. ALL RIGHTS RESERVED.

Text-Level Semantics: mark Element and wbr Element

- The **mark element** highlights the text that's enclosed in the element.
- The **wbr element** indicates the appropriate place to break a word when the text wraps to multiple lines.
- You might use **wbr** to prevent a word from breaking in an awkward place.
- The **<wbr>** tag represents a line break opportunity - the browser should only render a line-break if necessary.
- The **<wbr>** tag has no end tag.