

## Chapter 3: Introduction to HTML Part 2

CS 80: Internet Programming

Instructor: Mark Edmonds

### HTML5 Forms

HTML5 introduces new input types for forms

- These input types are self-validating
  - No need for javascript validation!
- However, the input data should be verified server-side

### HTML5 Forms

New HTML5 form input types

- Each is part of a `<input type="type">` tag
- `color`: allows user to input a RGB, many browsers have color picker.
  - Input must be a hexadecimal value (color code)
- `date`: calendar date input, many browsers have a calendar-style picker

### HTML5 Forms

New HTML5 form input types

- `datetime`: date and time input
  - yyyy-mm-ddThh:mm+ff in UTC (English time)
- `datetime-local`: date and time input
  - yyyy-mm-ddThh:mm+ff

### HTML5 Forms

New HTML5 form input types

- `email`: email input
  - address@domain.com

### HTML5 Forms

New HTML5 form input types

- `month`: month input
  - yyyy-mm
- `number`: any number input
  - min and max attributes set the min and max possible valid values

## HTML5 Forms

New HTML5 form input types

- `range`: range slider between particular values - no validation required
  - min and max attributes set the min and max possible values
  - all values are valid

## HTML5 Forms

New HTML5 form input types

- `search`: search field, behaves like a text input
- `tel`: telephone number input, **HTML5 does not validate this input!**

## HTML5 Forms

New HTML5 form input types

- `time`: time input
  - hh:mm
- `url`: valid URL input
  - http://domain.com

## HTML5 Forms

New HTML5 form input types

- `week`: input a specific week (nn is a week number, between 01 and 53)
  - yyyy-Wnn
- If a browser does not support the specified input type, the type will be treated as "text".
  - Means you must *always* validate server-side!

**Example: new\_forms.html**

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <title>New Forms in HTML5</title>
7 </head>
8
9 <body>
10  <h3>Demo of the new HTML5 form input types</h3>
11  <form method="post" action="http://deitel.com">
12    <p>
13      <label>color:
14        <input type="color" />
15      </label>
16    </p>
17    <p>
18      <label>date:
19        <input type="date" />
20      </label>
21    </p>
22    <p>
23      <label>datetime:
24        <input type="datetime" />
25      </label>
26    </p>
27    <p>
28      <label>datetime-local:
29        <input type="datetime-local" />
30      </label>
31    </p>
32    <p>
33      <label for="email">email:
34        <input type="email" id="email" />
35      </label>
36    </p>
37    <p>
38      <label>month:
39        <input type="month" />
40      </label>
41    </p>
42    <p>
```

```
43     <label>number:
44         <input type="number" min="-50" max="100" />
45     </label>
46 </p>
47 <p>
48     <label>range:
49         <input type="range" min="5" max="10" value="7"/>
50     </label>
51 </p>
52 <p>
53     <label>search:
54         <input type="search" />
55     </label>
56 </p>
57 <p>
58     <label>tel:
59         <input type="tel" />
60     </label>
61 </p>
62 <p>
63     <label>time:
64         <input type="time" />
65     </label>
66 </p>
67 <p>
68     <label>url:
69         <input type="url" />
70     </label>
71 </p>
72 <p>
73     <label>week:
74         <input type="week" />
75     </label>
76 </p>
77 <input type="submit" value="Submit"/>
78 <input type="reset" value="Reset"/>
79 </form>
80 </body>
81
82 </html>
```

## Input Focusing

Input `tabindex` attribute

- `tabindex` attribute enables controlling the order of input focus on TAB
- Ordering starts at 1

## HTML5 Forms

New HTML5 form attributes

- `autofocus`: applied to input tag, automatically gives focus to this element (e.g. user can type immediately)
  - Can only be used on one input element per document

## HTML5 Forms

New HTML5 form attributes

- `formnovalidate`: applied to input tag, allows a form to bypass validation
  - Must be applied to an input type
  - Means you must validate server side!!
- `novalidate`: applied to the form tag, allows a form to bypass validation
  - Means you must validate server side!!

## HTML5 Forms

New HTML5 form attributes

- `placeholder`: applied to input tag, temporary text in an input field
  - Typically either the format of the input, or an example
  - Only applies to the following input types:
    - \* `text`, `search`, `url`, `tel`, `email`

## HTML5 Forms

New HTML5 form attributes

- `required`: applied to input tag, requires the user to input the form element
  - Form cannot be submitted until they do
  - this is overridden by `formnovalidate` or `novalidate`

## HTML5 Forms

New HTML5 form attributes

- `pattern`: applied to input tag, requires the input match the specified regular expression

## Regular Expressions

A regular what?

- What's a regular expression (regex)?
  - A pattern matching tool
  - Text is described through a pattern, rather than a full description outlining every character

## Regular Expressions

Telephone pattern explained

- Pattern:

```
1 \(\d{3}\) +\d{3}-\d{4}
```

- Preliminary note: \ in regex is similar to & in HTML
  - / escapes the special character
    - \* Escaping means stripping it of its special meaning

## Regular Expressions

Telephone pattern explained

- Pattern:

```
1 \(\d{3}\) +\d{3}-\d{4}
```

- ( has a special meaning in regex, namely to start saving the matched pattern to a *group* (saving a section of the pattern for later use)
- ) marks the end of the saved group
- You can then reference the group later with \# where # is the group number used in the expression you wish to reference
  - E.g. to reference the first saved group, use \1
- Because of this special meaning, we have to escape it

## Regular Expressions

Telephone pattern explained

- Pattern:

```
1 \(\d{3}\) +\d{3}-\d{4}
```

- `\d` says we want to look for a digit ([0-9])
  - In this case, the escape gives `d` a special meaning - namely a digit
- `{3}` says we want to look for the previous pattern 3 times (3 digits)

## Regular Expressions

Telephone pattern explained

- Pattern:

```
1 \(\d{3}\) +\d{3}-\d{4}
```

- `" "` is a literal space to look for
  - `+` means match the previous pattern 1 or more times
- Then, we look for 3 more digits, followed by a `-` then 4 more digits

## HTML5 Forms

New HTML5 form attributes

- `autocomplete`: added to form or input tag, allows user to use previously submitted information when returning to the same form
  - You can turn autocomplete on or off for specific elements, overriding a value already set
  - Allows you to protect sensitive information from being autocomplete candidates
    - \* E.g. password, credit card info, etc.

## datalist element

- Provides input options for a text form input
- As the user starts typing, the list will narrow
  - How the list narrows depends on the browser
- The datalist must be connected to an input through the `id` attribute

- This is the glue between the list and the corresponding form input
- Also means one list can be used on multiple form inputs

### Example: `adv_new_forms.html`

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <title>New Forms in HTML5</title>
7 </head>
8
9 <body>
10  <h3>Demo of the new HTML5 form input types</h3>
11  <form method="post" action="http://deitel.com">
12    <p>
13      <label>color:
14      <input type="color"/>
15    </label>
16  </p>
17  <p>
18    <label>date:
19    <input type="date" />
20  </label>
21  </p>
22  <p>
23    <label>email:
24    <input type="email" placeholder="edmonds_mark@smc.edu"
25      autofocus/>
26  </label>
27  </p>
28  <p>
29    <label>number:
30    <input type="number" min="-50" max="100" />
31  </label>
32  </p>
33  <p>
34    <label>range:
35    <input type="range" min="5" max="10" value="7"/>
36  </label>
37  </p>
```



```
37     <p>
38         <label>tel:
39             <input type="tel" pattern = "\(\d{3}\) +\d{3}-\d{4}"
              placeholder="(###) ###-####" />
40         </label>
41         <input type="tel" placeholder="telephone">
42     </p>
43     <p>
44         <label>url:
45             <input type="url" required />
46         </label>
47     </p>
48     <input type="submit" value="Submit" />
49     <input type="reset" value="Reset" />
50 </form>
51
52 <h3>Form without validation through form submit input type (
    formnovalidate)</h3>
53
54 <form action="http://deitel.com">
55     <p>
56         <label>number:
57             <input type="number" autofocus />
58         </label>
59     </p>
60     <p>
61         <label>url:
62             <input type="url" required />
63         </label>
64     </p>
65     <input type="submit" value="Submit" formnovalidate />
66     <input type="reset" value="Reset" />
67 </form>
68
69 <h3>Form without validation through form tag (novalidate)</h3>
70
71 <form action="http://deitel.com" novalidate autocomplete="on">
72     <p>
73         <label>number:
74             <input type="number" id="num" autofocus />
75         </label>
76     </p>
77     <p>
```

```
78     <label>url:
79     <input type="url" id="url" required />
80     </label>
81 </p>
82 <p>
83     <!-- for connects to input with id="pt" for autofocus when parent
84          /child relationship does not exist-->
85     <label for="pt">
86         plain text:
87     </label>
88     <input type="text" id="pt" required autocomplete="off"/>
89 </p>
90 <p>
91     <label for="month">Month:
92     <input type="text" id="month" list="months"/>
93     <datalist id="months">
94         <option value = "January">
95         <option value = "February">
96         <option value = "March">
97         <option value = "April">
98         <option value = "May">
99         <option value = "June">
100        <option value = "July">
101        <option value = "August">
102        <option value = "September">
103        <option value = "October">
104        <option value = "November">
105        <option value = "December">
106    </datalist>
107    </label>
108 </p>
109 <input type="submit" value="Submit" />
110 <input type="reset" value="Reset" />
111 </form>
112 </body>
113 </html>
```

### Example: fig3\_17.html

```
1 <!DOCTYPE html>
2 <!-- Fig. 3.17: autocomplete.html -->
```

```
3  <!-- New HTML5 form autocomplete attribute and datalist element. -->
4  <html>
5
6  <head>
7      <meta charset="utf-8">
8      <title>New HTML5 autocomplete Attribute and datalist Element</title>
9  </head>
10
11 <body>
12     <h1>Autocomplete and Datalist Demo</h1>
13     <p>This form demonstrates the new HTML5 autocomplete attribute and
14         the datalist element.
15     </p>
16     <!-- turn autocomplete on -->
17     <form method="post" autocomplete="on">
18         <p>
19             <label>First Name:
20                 <input type="text" id="firstName" placeholder="First
21                     name" /> (First name)
22             </label>
23         </p>
24         <p>
25             <label for="lastName">Last Name:
26                 </label>
27                 <input type="text" id="lastName" placeholder="Last name" />
28                 (Last name)
29         </p>
30         <p>
31             <label>Email:
32                 <input type="email" id="email" placeholder="name@domain
33                     .com" /> (name@domain.com)
34             </label>
35         </p>
36         <p>
37             <label for="txtList">Birth Month:
38                 <input type="text" id="txtList" placeholder="Select a
39                     month" list="months" />
40                 <datalist id="months">
41                     <option value="January">
42                     <option value="February">
43                     <option value="March">
44                     <option value="April">
```

```
40         <option value="May">
41         <option value="June">
42         <option value="July">
43         <option value="August">
44         <option value="September">
45         <option value="October">
46         <option value="November">
47         <option value="December">
48     </datalist>
49     </label>
50 </p>
51 <p>
52     <input type="submit" value="Submit" />
53     <input type="reset" value="Clear" />
54 </p>
55 </form>
56 </body>
57
58 </html>
```

### Page structure elements

- **header**: used to specify the header of the webpage, but can actually be included multiple times
  - Used to give introductory content or navigational links
- **footer**: used to specify the footer of the webpage, but can actually be included multiple times
  - Used to give authorship information, copyright, contact, sitemap, etc

### Page structure elements

- **time**: used to identify a date, a time, or both
- **nav**: used to group navigational links
  - Do not place all links in the nav element, only the navigational ones (could be in the header or the footer typically)
- **figure**: describes a figure, used to group a **figcaption** and the **img** tag
- **figcaption**: provides a caption for the corresponding figure

### Page structure elements

- **article**: specifies independent, self-contained content
  - Can be nested

- Intended for forum posts, blog posts, news story, commentary, etc
- `details`: provides information about a specific topic
- `summary`: displays summary of content inside of a `details` tag
  - The details of the summary are contained in a `detailstag`

## Page structure elements

- `section`: provides a section-based structure to a page; similar to
- `article`, but more general
  - Can be nested
  - Used frequently with `id` attributes for internal linking
  - Typically also comes with a header of some kind
- `aside`: describes content related to something, but not critical
  - Could be background history or related information

## Page structure elements

- `meter`: shows an amount
  - Similar to a `range` (but is not input), or a loading bar (but does not animate)
- `mark`: highlight enclosed text
- `wbr`: controls when to break a word when text wrapping to the next line
  - Would this require a closing tag?

## Example: new\_elements.html

```
1 <!DOCTYPE html>
2 <!-- Fig. 3.18: sectionelements.html -->
3 <!-- New HTML5 section elements. -->
4 <html>
5
6 <head>
7   <meta charset="utf-8">
8   <title>New HTML5 Section Elements</title>
9 </head>
10
11 <body>
12   <header>
13     <!-- header element creates a header for the page -->
14     
15     <h1>Welcome to the Deitel Buzz Online</h1>
```

```
16     <!-- time element inserts a date and/or time -->
17     <time>2012-01-17</time>
18 </header>
19
20 <section id = "1">
21 <!-- Begin section 1 -->
22     <nav>
23         <!-- nav element groups navigation links -->
24         <h2> Recent Publications</h2>
25         <ul>
26             <li>
27                 <a href = "http://www.deitel.com/books/iw3http5">Internet &
                    World Wide Web How to Program, 5/e</a>
28             </li>
29             <li>
30                 <a href = "http://www.deitel.com/books/androidfp/">Android
                    for Programmers: An App-Driven Approach</a>
31             </li>
32             <li>
33                 <a href = "http://www.deitel.com/books/iphonefp">iPhone for
                    Programmers: An App-Driven Approach</a>
34             </li>
35             <li>
36                 <a href = "http://www.deitel.com/books/jhttp9/">Java How to
                    Program, 9/e</a>
37             </li>
38             <li>
39                 <a href = "http://www.deitel.com/books/cpphttp8/">C++ How to
                    Program, 8/e</a>
40             </li>
41             <li><a href = "http://www.deitel.com/books/vcsharp2010http">
                    Visual C# 2010 How to Program, 4/e</a>
42             </li>
43             <li>
44                 <a href = "http://www.deitel.com/books/vb2010http">Visual
                    Basic 2010 How to Program</a>
45             </li>
46         </ul>
47     </nav>
48 </section>
49
50 <section id = "2">
51 <!-- Begin section 2 -->
```

```
52     <h2>How to Program Series Books</h2>
53     <h3><em>Java How to Program, 9/e</em></h3>
54
55     <figure>
56     <!-- figure element describes the image -->
57         <img src = "jhttp.jpg" alt = "Java How to Program, 9/e" />
58         <!-- figurecaption element inserts a figure caption -->
59         <figcaption>
60             <em>Java How to Program, 9/e</em>cover.
61         </figcaption>
62     </figure>
63     <!--article element represents content from another source -->
64
65     <article>
66         <header>
67             <h5>From
68                 <em>
69                     <a href = "http://www.deitel.com/books/jhttp9/">Java How to
                        program, 9/e: </a>
70                 </em>
71             </h5>
72         </header>
73         <p>Features include:
74             <ul>
75                 <!-- mark element highlights text -->
76                 <li>Rich coverage of fundamentals, including <mark>two
                        chapters on control statements.</mark></li>
77                 <li>Focus on <mark>real-world examples.</mark></li>
78                 <li><mark>Making a Difference exercises set.</mark></li>
79                 <li>Early introduction to classes, objects, methods and
                        strings.</li>
80                 <li>Integrated exception handling.</li>
81                 <li>Files, streams and object serialization.</li>
82                 <li>Optional modular sections on language and library
                        features of the new Java SE 7.</li>
83                 <li>Other topics include: Recursion, searching, sorting,
                        generic collections, generics, data structures, applets,
                        multimedia, multithreading, databases/JDBC&trade;, web-app
                        development, web services and an optional ATM Object-
                        Oriented Design case study.</li>
84             </ul>
85             <!-- summary element represents a summary for the -->
86             <!-- content of the details element -->
```

```
87     <details>
88     <summary>Recent Edition Testimonials</summary>
89     <ul>
90     <li>"Updated to reflect the state of the art in Java
          technologies; its deep and crystal clear explanations
          make it indispensable. The social-consciousness [Making
          a Difference] exercises are something really new and
          refreshing." <strong>&mdash;Jos&eacute; Antonio Gonz&
          aacute;lez Seco, Parliament of Andalusia</strong></li>
91     <li>"Gives new programmers the benefit of the wisdom
          derived from many years of software development
          experience."<strong> &mdash;Edward F. Gehringer, North
          Carolina State University</strong></li>
92     <li>"Introduces good design practices and methodologies
          right from the beginning. An excellent starting point
          for developing high-quality robust Java applications." <
          strong>&mdash;Simon Ritter, Oracle Corporation</strong
          ></li>
93     <li>"An easy-to-read conversational style. Clear code
          examples propel readers to become proficient in Java." <
          strong>&mdash;Patty Kraft, San Diego State University</
          strong></li>
94     <li>"A great textbook with a myriad of examples from
          various application domains&mdash; excellent for a
          typical CS1 or CS2 course." <strong>&mdash;William E.
          Duncan, Louisiana State University</strong></li>
95     </ul>
96     </details>
97 </p>
98 </article>
99 <!-- aside element represents content in a sidebar that's -->
100 <!-- related to the content around the element -->
101 <aside>
102     The aside element is not formatted by the browsers.
103 </aside>
104 <h2>Deitel Developer Series Books</h2>
105 <h3><em>Android for Programmers: An App-Driven Approach</em></h3>
106 Click <a href = "http://www.deitel.com/books/androidfp/">here</a>
    for more information or to order this book.
107 <h2>LiveLessons Videos</h2>
108 <h3><em>C# 2010 Fundamentals LiveLessons</em></h3>
109 Click <a href = "http://www.deitel.com/Books/LiveLessons/">here</a>
    for more information about our LiveLessons videos.
```



```
110     </section>
111
112     <section id = "3"> <!-- Begin section 3 -->
113         <h2>Results from our Facebook Survey</h2>
114         <p>If you were a nonprogrammer about to learn Java for the first
            time, would you prefer a course that taught Java in the context
            of Android app development? Here are the results from our survey
            :</p>
115         <!-- meter element represents a scale within a range -->
116         0 <meter min = "0" max = "54" value = "14"></meter> 54
117         <p>Of the 54 responders, 14 (green) would prefer to learn Java in
            the context of An<wbr>droid app development.</p>
118     </section>
119     <!-- footer element represents a footer to a section or page, -->
120     <!-- usually containing information such as author name, -->
121     <!-- copyright, etc. -->
122     <footer>
123     <!-- wbr element indicates the appropriate place to break a -->
124     <!-- word when the text wraps -->
125         <h6>&copy; 1992-2012 by Deitel & Associ<wbr>ates, Inc. All
            Rights Reserved.</h6>
126         <!-- address element represents contact information for a -->
127         <!-- document or the nearest body element or article -->
128         <address>
129             Contact us at <a href = "mailto:deitel@deitel.com">deitel@deitel.
                com</a>
130         </address>
131     </footer>
132
133 </body>
134 </html>
```

## Tools in a Toolbox

Remember, these are general guidelines. HTML is like a toolbox; you are free to use them as you see fit.

A hammer is better at nailing than a saw. In general, it's best to stick to the convention.

## Advanced HTML5

Video

- `controls` attribute (no arguments/assignment) allows you to add video control buttons, such as play, pause, volume
- `height` and `width` can be added to control the size of the video
- `muted` will start the video initially muted
- `loop` will determine whether or not the video should loop after finishing

## Advanced HTML5

### Video

- `autoplay` will start the video as soon as it loads
- The `video` tag is dissimilar from the `img` tag
  - You can use the video tag as a void tag (like an `img` tag) or you can specify multiple `source` tags
  - `source` tags specify a video file and a type
    - \* A `video` tag can have multiple `source` tags; the browser will play the first type supported.
    - \* If no types are supported, the text in between the `video` tag will be displayed.

## Advanced HTML5

### Audio

- Extremely similar to playing video
  - Only difference with video is no `height` and `width` attribute
  - `controls`, `autoplay`, `loop`, `muted` are all the same
  - Must specify an audio file (like MP3) instead of video

### Example: `audio.html`

```
1 <h3>Audio with single source</h3>
2 <audio src="track.mp3" width="320" height="240" controls>
3
4 <h3>Audio with multiple sources</h3>
5 <audio width="320" height="240" controls>
6     <source src="track.mp3" type="audio/mp3">
7     <source src="track.ogg" type="audio/ogg">
8     Your browser does not support the audio tag.
9 </audio>
```

## Advanced HTML5

### Canvas

- `canvas`: allows you to draw 2D and 3D drawings on the screen
  - This requires javascript, so just hold on, we will go over this later!