

HTML5 and CSS3, Chapter 7

Geoffrey Matthews

Department of Computer Science
Western Washington University

April 5, 2017

Source code from book

- ▶ `http://www.aharrisbooks.net/haio/`

Forms

- ▶ Forms are used to provide input from the user.
- ▶ This section of the book shows how to build forms.
- ▶ In order to *do* something with the data, we need a programming language.
- ▶ In Part IV we use Javascript on the client to do something.
- ▶ in Part V we use PHP on the server to do something.
- ▶ Consider this part a reference for later.

Types of forms

- ▶ Text boxes
- ▶ Password boxes
- ▶ Text areas
- ▶ Select lists
- ▶ Check boxes
- ▶ Radio buttons
- ▶ Buttons
- ▶ Labels
- ▶ Fieldsets and legends

Basic elements of a form

```
<h1>A basic form</h1>
<form action = "">
  <h2>Form elements go here</h2>
  <h3>Other HTML is fine, too.</h3>
  <p>
    <input type = "text"
      value = "googoo" />
  </p>
</form>
```

```
<p>
  Note this code is slightly improved from book
  version. I've placed the input
  element inside the form
  tag. Thanks to Jim for the catch!
</p>
```

Organizing a form with fieldsets

```
<h1>Sample Form with a Fieldset</h1>
<form action = "">
  <fieldset>
    <legend>Personal Data</legend>
    <p>
      <label>Name</label>
      <input type = "text" />
    </p>
    <p>
      <label>Address</label>
      <input type = "text" />
    </p>
    <p>
      <label>Phone</label>
      <input type = "text" />
    </p>
  </fieldset>
</form>
```

Fieldsets

- ▶ Not necessary, but group parts of a form.
- ▶ Legend identifies the entire form.
- ▶ Label identifies each input form.
- ▶ Can use paragraphs to group labels and inputs.
- ▶ Label tag can have a `for` attribute to connect with a specific input.

A standard text form

```
<form action = ">  
  <p>  
    <label>Name</label>  
    <input type = "text"  
      id = "txtName"  
      value = "Jonas"/>  
  </p>  
</form>
```


Text form attributes

- ▶ `type`: determines the general type of input.
 - ▶ `text`
 - ▶ `password`
 - ▶ *etc.*
- ▶ `id`: when we deal with the data, the program has to know the name (identifier) of the data.
- ▶ `value`: default value.
- ▶ `size`: number of characters displayed.
- ▶ `maxlength`: largest number of characters allowed.

Password

```
<fieldset>
  <legend>Enter a password</legend>
  <p>
    <label>Type password here</label>
    <input type = "password"
           id = "pwd"
           value = "secret" />
  </p>
</fieldset>
```

- ▶ Replaces visible text characters with asterisks.
- ▶ Provides *no* security except for people looking over your shoulder.
- ▶ SSL (Secure Socket Layer) provides real security.

Multi-line text input

```
<p>
  <label>
    Please enter the sum total of
    Western thought. Be brief.
  </label>
</p>
<p>
  <textarea id = "txtAnswer"
            rows = "10"
            cols = "40"></textarea>
</p>
```

- ▶ It needs an id
- ▶ Specify the size with rows and cols.
- ▶ The content goes between the tags.
- ▶ Whitespace between the tags goes in the output.

Drop-down selections

```
<label>What is your favorite color?</label>
<select id = "selColor">
  <option value = "#ff0000">Red</option>
  <option value = "#00ff00">Green</option>
  <option value = "#0000ff">Blue</option>
  <option value = "#00ffff">Cyan</option>
  <option value = "#ff00ff">Magenta</option>
  <option value = "#ffff00">Yellow</option>
  <option value = "#000000">Black</option>
  <option value = "#ffffff">White</option>
</select>
```

- ▶ Saves screen space.
- ▶ Limits input.
- ▶ The value can be different from what the user sees.

Check boxes

```
<p>
  <input type = "checkbox"
        id = "chkPeace"
        value = "peace" />
  World peace
</p>
<p>
  <input type = "checkbox"
        id = "chkHarmony"
        value = "harmony" />
  Harmony and brotherhood
</p>
<p>
  <input type = "checkbox"
        id = "chkCash"
        value = "cash" />
  Cash
</p>
```

This all seems inconsistent

- ▶ Forms are inconsistent:
 - ▶ Sometimes the value of a form element is visible to the users, sometimes not.
 - ▶ Sometimes the text the user sees is inside the tag, sometimes not.
- ▶ The standards for HTML evolved over time.
- ▶ It's difficult to change a pattern that has thousands or millions of uses.
- ▶ It's best to consult an example any time you have to use one of these.

Radio buttons

```
<p>    <input type = "radio"
        name = "radPrice"
        id = "rad100"
        value = "100" />Too much    </p>
<p>    <input type = "radio"
        name = "radPrice"
        id = "rad200"
        value = "200" />Way too much    </p>
<p>    <input type = "radio"
        name = "radPrice"
        id = "rad5000"
        value = "5000"
        checked = "checked" />
    You've got to be kidding.    </p>
```

Radio buttons

- ▶ Only one can be checked at a time.
- ▶ They have to be in a group.
- ▶ They have to be the same name!
 - ▶ This creates the group.
- ▶ You can have more than one group, just use different names.
- ▶ One of them has to be checked.

Other buttons

```
<legend>
  input-style buttons
</legend>
<input type = "button"
      value = "input type = button" />
<input type = "submit" />
<input type = "reset" />
</fieldset>
<fieldset>
  <legend>button tag buttons</legend>
  <button type = "button">
    button tag
  </button>
  <button>
    <img src = "clickMe.gif"
        alt = "click me" />
  </button>
```

Button tags vs. button attributes

- ▶ The type determines the style.
- ▶ The caption goes in the block.
- ▶ You can incorporate other elements.

New form input types

- ▶ date
- ▶ time
- ▶ datetime
- ▶ datetime-local
- ▶ week
- ▶ month
- ▶ color
- ▶ number
- ▶ range
- ▶ search
- ▶ email
- ▶ tel
- ▶ url