

HTML Forms

Lecture 21

Robb T. Koether

Hampden-Sydney College

Mon, Mar 12, 2018

- 1 HTML Forms
- 2 The `input` Element
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

Outline

- 1 HTML Forms
- 2 The `input` Element
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

Forms

```
<form name="form_name" method="GET" action="file_name">
    :
</form>
```

- An HTML `form` element allows the client to enter data, which is then transmitted, typically, to a PHP program.
- The *form_name* is optional.
- The method is either `GET` or `POST`.
- The file *file_name* is either a PHP file or an HTML file.

HTML Forms

- Within the `form` tag, we need to include at least the following two attributes.
 - `method` – specifies how to transmit the data.
 - `action` – specifies what file to invoke when the form is submitted.
- We may also include other attributes.
 - `name` – used for reference.
 - `id` – used for reference
 - `onsubmit` – invokes Javascript code when the form is submitted.

- Both `GET` and `POST` will transmit the form values to the action file, but by slightly different means.
 - `GET` - The values are appended to the name of the invoked action file, visible in the URL.
 - `POST` - The values are transmitted “quietly.”

HTML Forms

- Within the `form` element, we may use a variety of methods to convey data to the server
 - Buttons
 - Text boxes
 - Radio buttons
 - Check boxes
 - Select lists
 - Text areas

Outline

- 1 HTML Forms
- 2 **The `input` Element**
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

The input Element

Text Boxes

```
<input type="input-type" name="element_name"/>
```

- An `input` element allows various forms of data entry, depending on its type.
- It needs a `type` attribute.
- Most `input` elements also need a `name` attribute.
 - The name becomes the associative index in the PHP program.
- Sometimes an `id` attribute is needed.
 - The `id` attribute is a unique identifier, used for reference purposes.
- Other attributes are used as needed, e.g., `value`.
- The `input` tag is self-closing.
- The type of input may be `submit`, `reset`, `text`, `password`, `radio`, or `checkbox`.

Outline

- 1 HTML Forms
- 2 The `input` Element
 - **Submit and Reset Buttons**
 - Text Boxes
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

Submit Button

Submit Button

```
<form method="GET" action="myaction.php">  
    :  
    <input type="submit" value="Click here"/>  
</form>
```

- This creates a standard button labeled “Click here.” (The default label is “Submit Query.”)
- The form data are transmitted to the file `myaction.php` when the submit button is pressed.
- If the web page has more than one form element, then each one should have its own submit button.

Reset Button

Submit Button

```
<form method="GET" action="myaction.php">  
  :  
  <input type="reset"/>  
</form>
```

- This creates a standard button labeled “Reset.” (The default label is “Reset.”)
- When the reset button is pressed, all values are given their initial values, as defined by the HTML code.

Outline

- 1 HTML Forms
- 2 **The `input` Element**
 - Submit and Reset Buttons
 - **Text Boxes**
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

The text Type

Text Boxes

Enter username:

```
<input type="text" name="username"/>
```

- The above example displays the text `Enter username:` with a **text box** after it.
- The user may enter his user name in the text box.
- When he clicks the submit button, his user name is transmitted to the action file.

The text Type

- Suppose that the user enters `bigjohn` in the text box.
- When the user submits the form, the form will create the URL

`myaction.php?username=bigjohn`

and send it to the server.

The text Type

- In the file `myaction.php`, the user name will be received through the associative array `$_GET`, where

```
$_GET['username'] = "bigjohn"
```

- The POST method is similar, except that
 - The values are not attached to the filename, so they do not appear in the URL.
 - The associative array is `$_POST`, not `$_GET`.

```
$_POST['username'] = "bigjohn"
```


Outline

- 1 HTML Forms
- 2 **The `input` Element**
 - Submit and Reset Buttons
 - Text Boxes
 - **Passwords**
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

The password Type

Passwords

```
<form method="POST" action="myaction.php">  
  Enter password:  
  <input type="password" name="password"/>  
</form>
```

- If the type is `password` rather than `text`, then the behavior is the same, except that the plain text is hidden as it is typed.
- For example, if the user types `hampden`, all that will appear is
●●●●●●●.
- Obviously, when sending a password, the `POST` method should be used. (Obviously, the password should be encrypted before being transmitted.)

Outline

- 1 HTML Forms
- 2 **The `input` Element**
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - **Radio Buttons**
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

The `radio` Type

- The **radio button** needs `type`, `name`, and `value` attributes.
- All radio buttons in a group use the same name, but different values.
- Only one button can be “on” at a time.

The radio Type

Radio Buttons

What size do you want?

```
<input type="radio" name="size" value="small"/> Small  
<input type="radio" name="size" value="medium"/> Medium  
<input type="radio" name="size" value="large"/> Large
```

- The user may select “Small,” “Medium,” or “Large.”

The radio Type

Radio Buttons

```
$_GET['size'] = "medium"
```

- When the user submits the form, the name of the transmitted attribute is `size` and its value is whichever button was last selected.

The radio Type

Radio Buttons

What size do you want?

```
<input type="radio" name="size" value="small" checked="checked"/> Small  
<input type="radio" name="size" value="medium"/> Medium  
<input type="radio" name="size" value="large"/> Large
```

- We may pre-select a category.
- This is a good idea for radio buttons, so that exactly one will always be selected.

Outline

- 1 HTML Forms
- 2 **The `input` Element**
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - Radio Buttons
 - **Check Boxes**
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

The checkbox Type

- **Check boxes** are similar to radio buttons, except that the user may check any combination of boxes, including none.
- The check box needs `type`, `name`, and `value` attributes.
- Because the choices are independent, check boxes are not logically grouped.
- Therefore, each box may have its own name.

The checkbox Type

Check Boxes

Which sports do you play?

```
<input type="checkbox" name="bskt"/> Basketball
```

```
<input type="checkbox" name="foot"/> Football
```

```
<input type="checkbox" name="base"/> Baseball
```

- Each selected item is given the default value `on`.
- All other items are undefined.

The checkbox Type

Check Boxes

Which sports do you play?

```
<input type="checkbox" name="sport" value="bskt" /> Basketball  
<input type="checkbox" name="sport" value="foot" /> Football  
<input type="checkbox" name="sport" value="base" /> Baseball
```

- We may assign one name to the group and use values other than “on.”
- But what will happen when the user checks more than one item?

The checkbox Type

Check Boxes

Which sports do you play?

```
<input type="checkbox" name="sport[]" value="bskt"/> Basketball  
<input type="checkbox" name="sport[]" value="foot"/> Football  
<input type="checkbox" name="sport[]" value="base"/> Baseball
```

- We may use an array name for the group of check boxes.
- In this case, `$_POST['sport']` or `$_GET['sport']` is an array.
- Only those values that are checked are placed in the array.
- In this case it is necessary to specify the value for each checkbox.

Outline

- 1 HTML Forms
- 2 The `input` Element
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

Labels

Which sports do you play?

```
<input type="checkbox" name="sport[]" id="bskt" value="bskt"/>  
<label for="bskt">Basketball</label>  
:
```

- Each radio button or checkbox has an associated label, indicating its purpose.
- We can create a `<label>` element that associates the label with the button through the button's `id`.
- Then the user may click on either the button itself or the label.

CSS :hover Selector

CSS :hover Selector

```
<style>
label:hover
{
    color: white;
    font-weight: bold;
    background-color: yellow;
}
</style>
```

- To indicate this to the user, we may use the `:hover` **selector** in the CSS section.
- The label will take on the CSS attributes when the mouse “hovers” over the label.

Outline

- 1 HTML Forms
- 2 The `input` Element
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

The password Tag

Text Areas

```
<textarea name="msg">Enter message here</textarea>
```

- For larger quantities of text, we can use a `textarea` element.
- A `textarea` element creates a rectangle in which we may enter multi-line text.
- The `textarea` element can set the size of the box by using the attributes `rows` and `cols`.
- The text between the tags will appear in the rectangle.
- If the text exceeds the size of the box, a scroll bar will appear.

Outline

- 1 HTML Forms
- 2 The `input` Element
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

The select Tag

Select List

What is your favorite color?

```
<select name="color">
<option value="red">Red</option>
<option value="yellow">Yellow</option>
<option value="green">Green</option>
<option value="blue">Blue</option>
</select>
```

- A `select` element allows the user to select from a pre-defined list of choices.
- Each choice is defined by an `option` element within the `select` tags.
- This creates a pop-up menu from which the user can make a selection.

The select Tag

Select List

What is your favorite color?

```
<select name="color">
<option value="red">Red</option>
<option value="yellow">Yellow</option>
<option value="green" selected="selected">Green</option>
<option value="blue">Blue</option>
</select>
```

- One of the options may be designated “selected.”
- In this example, the “green” option will appear selected when the page is loaded.

Multiple Selections

Select List

What is your favorite color?

```
<select name="color[]" multiple="multiple" size="3">
<option value="red">Red</option>
<option value="yellow">Yellow</option>
<option value="green" selected="selected">Green</option>
<option value="blue">Blue</option>
</select>
```

- With the `multiple` attribute, we can make multiple selections.
- If we use the `multiple` attribute, then we should make the name of the `select` element an array.
- With the `size` attribute, we can specify how many options are in view at one time.

Outline

- 1 HTML Forms
- 2 The `input` Element
 - Submit and Reset Buttons
 - Text Boxes
 - Passwords
 - Radio Buttons
 - Check Boxes
- 3 Enhancing Radio Buttons and Checkboxes
- 4 Text Areas
- 5 Select Lists
- 6 Assignment

Assignment

Assignment

- Visit the W3Schools website
 - `http://www.w3schools.com/html/`
 - Visit the sections labeled “HTML Forms.”