

HTML Forms

BY JAROSLAV MOHAPL

Abstract

How to write an HTML form, create control buttons, a text input and a text area. How to input data from a list of items, a drop down list, and a list box. Simply useful things about HTML forms.

Key Phrases HTML form, radio button, check box, text box, drop down list, list box.

Key Words HTML, form, button, data, submit, reset, controls.

Introduction

Forms are designed for interaction of the user with the server. Till recently, everything what the user typed in the form remained on his client machine till the moment the submit button was pressed. With the fast spread of the AJAX technology, this is no longer true, and every user's keystroke and mouse button click can be sent directly to the server without the form being reloaded. This page deals with the HTML layout of the form. The dynamic that must be implemented to deliver the data safely to the server will be discussed elsewhere.

Simple form

A form is created by the form element and there is nothing visible about it in the open HTML document. Perhaps, except that the form element has a block display, i.e. it has the same effect as the paragraph element. The input elements inside the form are the visible part a user can, and often must, interact with in order to submit the form. The input element is self-closing and has a number of attributes specifying its shape and function. The simplest form is implemented as follows:

```
<form action="#" method="post" name="myform">
  This is a dummy form.<br />
  <input type="submit" value="Submit Form" />
</form>
```

The above form displays a clickable button, which, on a mouse click, changes its shape to indicate that it is held down. The action attribute specifies the URL where the form is submitted. The method takes two values, get and post. While get submits data to the server via a query, post does not. The name attribute is utilized by JavaScript and scripting languages in general.

Control buttons

The following input types create clickable buttons: button, submit, reset, and image. The text on the button is drawn by the value attribute. The button attribute is usually accompanied by the onclick attribute specifying the action triggered by pushing the button. The image attribute must be followed by the src attribute specifying the image source replacing the default button shape. On click, the image button triggers the submit action and adds to the data coordinates of the pointer. That is useful for creation of image maps. The submit attribute creates a submit button forcing submission of the form to the URL specified in the form's action attribute. The reset button brings all form input elements to the default state. Typical examples of buttons look like this:

```
<input type="submit" value="Submit" />
<input type="reset" value="Reset" />
<input type="button" value="Preview" onclick="preview()" />
<input type="image" src="mickey.gif" onclick="setlabel()" />
```

Text input and text area

Input values determining text fields are: text, and password. Besides two input field types, we have also the textarea, which is a tag determining the text area element. The following code draws text input elements.

```
<input type="text" size="20" maxlength="20"/>
<input type="password" size="10" maxlength="10"/>
<textarea cols="20" rows="10"> Default text. </textarea>
```

The size attribute determines width of the text/password field in characters. The cols and rows determine the number of columns and rows, respectively, in the text area. The text is automatically wrapped and if this text area is exceeded, a scroll bar is automatically added on the side to allow viewing of the hidden text.

Input from a list of items

Such an option provide: radio buttons, check boxes, and list boxes. Radio buttons are used in situations when we want the user indicate exactly one choice from several possibilities in terms of yes or no. The next example presents a group of three radiobuttons.

```
<fieldset>
  <legend>Your favorite veggies are:</legend>
  <input type="radio" name="radioset" />
    apples <br />
  <input type="radio" name="radioset" checked="checked"/>
    bananas <br />
  <input type="radio" name="radioset" />
    potatoes <br />
</fieldset>
```

The fieldset element draws a line around the items to choose from. The legend element text is the caption for the area. By default, the radio buttons are drawn empty. The checked attribute assures the element is drawn as selected. The common name is necessary to assure that only one item can be selected. Without it, multiple selection is possible, which is not desirable, because that is the behavior the user expects from check boxes, not radio buttons! As you guess, a checkbox list can be invoked like this.

```
<fieldset>
  <legend>Check what you want to buy from us:</legend>
  <input type="checkbox" name="box1" />
    apples <br />
  <input type="checkbox" name="box2" checked="checked"/>
    bananas <br />
  <input type="checkbox" name="box3" />
    potatoes <br />
</fieldset>
```

Drop down list and list box

The select tag defines an element resembling a list box, or menu, one can click on and highlight the selected item. It admits one selection only by default, but can be turned into a multi-selection box using the multiple attribute. Whether it is a list box or a drop down list depends on the size attribute. Boxes of size 1 with more than one element are automatically displayed as drop down lists. The list items are enclosed in option elements as follows:

```
<select>
  <option>Beef and potatoes</option>
  <option>Pork and dumplings</option>
  <option>Chicken with rice</option>
</select>
```

Hidden elements

The hidden element serves as a storage for data the form designer must use but does not intend to display in the form. The element is self-closing and has form

```
<hidden value="You will not see this in your form!"/>
```

References

C. D. Knukles and D. S. Yuen: *Web Applications, Concepts & Real World Designs*. *Wiley* 2005.

Web Design Group HTML Help and Web Authoring Reference. *Web Design Group* 2006.

HTML Forms and Input. *W3 Schools* 2006.

Markup Validation Service. *World Wide Web Consortium* 2006.