Form

Lecture 6

What are forms?

- <form> is just another kind of HTML tag
- HTML forms are used to create (rather primitive) GUIs on Web pages
  - Usually the purpose is to ask the user for information
  - The information is then sent back to the server
- A form is an area that can contain form elements **is used to create a data input form.**
  - The syntax is: <form ***parameters***> ***...form elements...*** </form>
  - Form elements include: buttons, checkboxes, text fields, radio buttons, drop-down menus, etc
    - Other kinds of HTML tags can be mixed in with the form elements
  - **The FORM element has three attributes:**
    - **ACTION, METHOD, and ENCTYPE.**
  - A form usually contains a Submit button to send the information in he form elements to the server
  - The form's ***parameters*** tell JavaScript how to send the information to the server (there are two different ways it could be sent)
  - Forms can be used for other things, such as a GUI for simple programs

Forms

METHOD:

- Specifies the way in which the data from the user are encoded.
- The default METHOD is GET, although the POST method is preferred.
- GET: The CGI program receives the encoded form input in the QUERY_STRING variable, which follows the "?" in the URL that calls the script.
- POST: The CGI script or program receives the encoded form input in its standard input stream. The CONTENT_LENGTH must be used.

Forms

ACTION:

- Specifies the destination URL to which the form should be submitted, once it has been completed by the user.
- If no URL is specified, the URL of the current document containing the form is used.
- MAILTO Action: The data from the form is mailed to the specified E-mail address. Use the POST method.

Forms

ENCTYPE:

- Tell the browser how the data from a form should be encoded when it is returned to the server.
- The default is “application/x-www-form-urlencoded” that converts spaces to “+” and uses “&” to delineated different data fields.

The <form> tag

- The <form ***arguments***> … </form> tag encloses form elements (and probably other HTML as well)
- The arguments to form tell what to do with the user input
  - action="***url***" (required)
    - Specifies where to send the data when the Submit button is clicked
  - method="get" (default)
    - Form data is sent as a URL with ?form_data info appended to the end
    - Can be used *only* if data is all ASCII and not more than 100 characters
  - method="post"
    - Form data is sent in the body of the URL request
    - Cannot be bookmarked by most browsers
  - target="***target***"
    - Tells where to open the page sent as a result of the request
    - ***target***= _blank means open in a new window
    - ***target***= _top means use the same window

The <input> tag

- Most, but not all, form elements use the input tag, with a type="..." argument to tell which kind of element it is
  - type can be text, checkbox, radio, password, hidden, submit, reset, button, file, or image
- Other common input tag arguments include:
  - name: the name of the element
  - value: the "value" of the element; used in different ways for different values of type
  - readonly: the value cannot be changed
  - disabled: the user can't do anything with this element
  - Other arguments are defined for the input tag but have meaning only for certain values of type

TYPE Attribute

TEXT type:

- Specifies a single line text entry field.
- Can be used with the maxlength and size attributes (maxlength >= size)

```
<p><b> first name:</b> <input name="fname" type = text maxlength=30 size =30></p>
<p><b> last name:</b> <input name="lname" type = text maxlength=30 size =30></p>
```

PASSWORD Type:

- Same as text except the text entered by the user is obscured.
- Use the maxlength and size attributes.

```
<p><b> enter your password:</b>
<input name="password" type = password maxlength=30 size =30></p>
```

Text input

A text field: <input type="text" name="textfield" value="with an initial value">

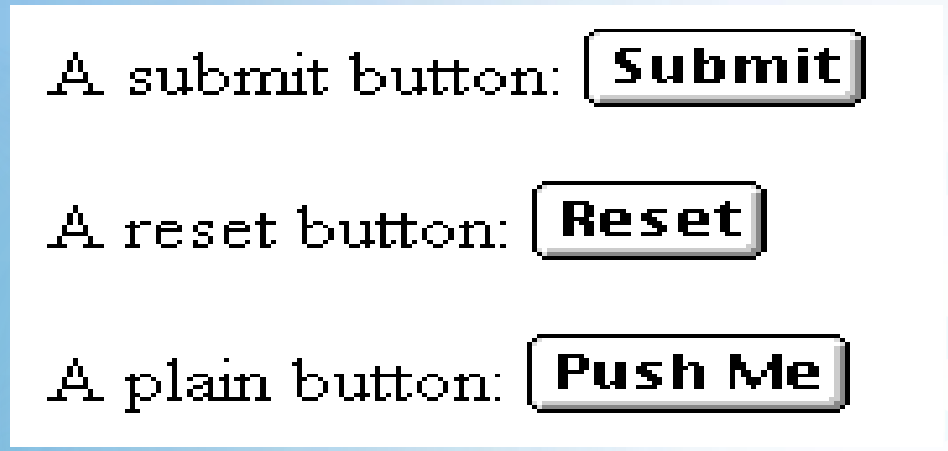
A multi-line text field <textarea name="textarea" cols="24" rows="2">Hello</textarea>

A password field: <input type="password" name="textfield3" value="secret">

- Note that two of these use the input tag, but one uses textarea

Buttons

- A submit button: <input type="submit" name="Submit" value="Submit">
- A reset button: <input type="reset" name="Submit2" value="Reset">
- A plain button: <input type="button" name="Submit3" value="Push Me">



- submit: send data
- reset: restore all form elements to their initial state
- button:
  - Creates a button whose use can be defined through scripting and **on Click event**.
  - Use to create a back button.
  - Only useful to browsers that support scripting.

- Note that the type is input, not "button"

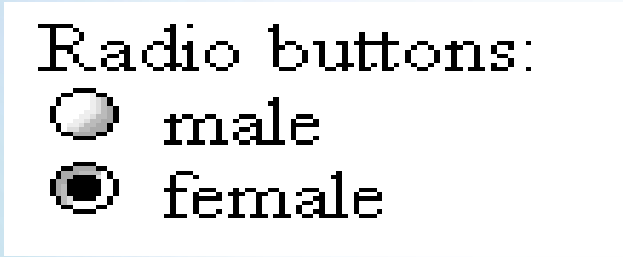
Checkboxes

- A checkbox: <input type="checkbox" name="checkbox" value="checkbox" checked>

  - type: "checkbox"
  - name: used to reference this form element from JavaScript
  - value: value to be returned when element is checked
  - Note that there is *no text* associated with the checkbox—you have to supply text in the surrounding HTML

Radio buttons


```
Radio buttons:<br>
<input type="radio" name="radiobutton" value="myValue1">male<br>
<input type="radio" name="radiobutton" value="myValue2" checked>female
```



- If two or more radio buttons have the same name, the user can only select one of them at a time
  - This is how you make a radio button "group"
- If you ask for the value of that name, you will get the value specified for the selected radio button
- As with checkboxes, radio buttons do not contain any text

Drop-down menu or list

- A menu or list:

```
<select name="select">
    <option value="red">red</option>
    <option value="green">green</option>
    <option value="BLUE">blue</option>
</select>
```

- Additional arguments:
  - **size**: the number of items visible in the list (default is "1")
  - **multiple**: if set to "**true**", any number of items may be selected (default is "false")

Hidden fields

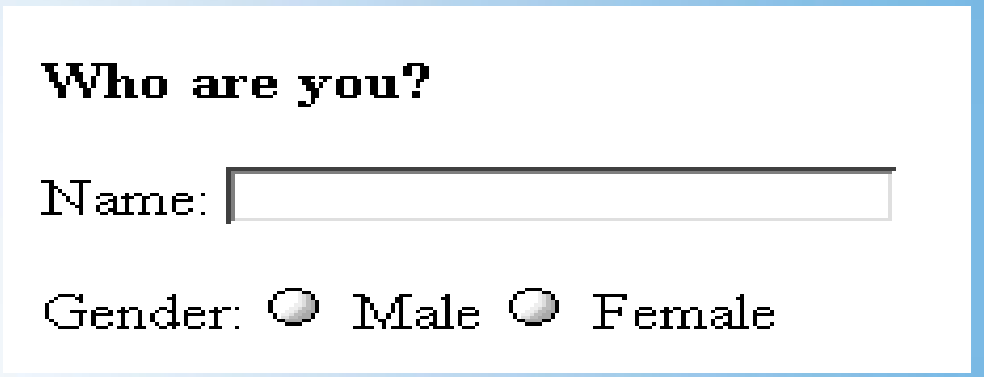
- <input type="hidden" name="hiddenField" value="nyah">
  <-- right there, don't you see it?

A hidden field: <-- right there, don't you see it?

- What good is this?
  - All input fields are sent back to the server, including hidden fields
  - This is a way to include information that the user doesn't need to see (or that you don't want her to see)
  - The value of a hidden field can be set programmatically (by JavaScript) before the form is submitted

A complete example


```
<html>
<head>
<title>Get Identity</title>
<meta http-equiv="Content-Type" content="text/html;
        charset=iso-8859-1">
</head>
<body>
<p><b>Who are you?</b></p>
<form method="post" action="">
  <p>Name:
    <input type="text" name="textfield">
  </p>
  <p>Gender:
    <input type="radio" name="gender" value="m">Male
    <input type="radio" name="gender" value="f">Female</p>
  </form>
</body>
</html>
```



TEXTAREA

- Let users enter more than one line of text.
- Uses attributes ROWS and COLS to size.
- WRAP Attribute:
  - OFF: No wrapping
  - VIRTUAL: Display wraps but long lines are sent as one line.
  - PHYSICAL: Word wraps and text is sent with wrap points.

PullDown Menu

- Use SELECT and OPTION to create pulldown menu.
- SELECT:
  - Allows the user to choose one (or possibly more) items from a list.
  - Attributes: MULTIPLE, SIZE, and NAME.
- OPTION:
  - Specifies the list items.
  - Attributes: SELECTED, VALUE, and LABEL

PullDown Menu

- Example:

```
<P><B>Pick your favorite baseball team:</B>
<BR><SELECT NAME="team">
<OPTION>Dodgers
<OPTION>Braves
<OPTION>Cardinals
<OPTION>Yankees
</SELECT>
```

Form Programming

- Handling GET Forms:
  - A typical invocation of a GET-style applicatoin might use a URL like this:

http://www.kumquat.com/cgi-bin/dump_get?name=bob&phone=555-1212

  - When the server processes this URL, it invokes the application named dump_get stored in the directory named cgi-bin. Everything after the question mark is passed to the application as parameters.
  - The parameters are placed in a variable named QUERY_STRING.

Form Programming

- Handling POST Method:
  - Data is sent in the format:

name=bob&phone=555-1212

  - No URL is sent.

Thank U