

Internet Programming

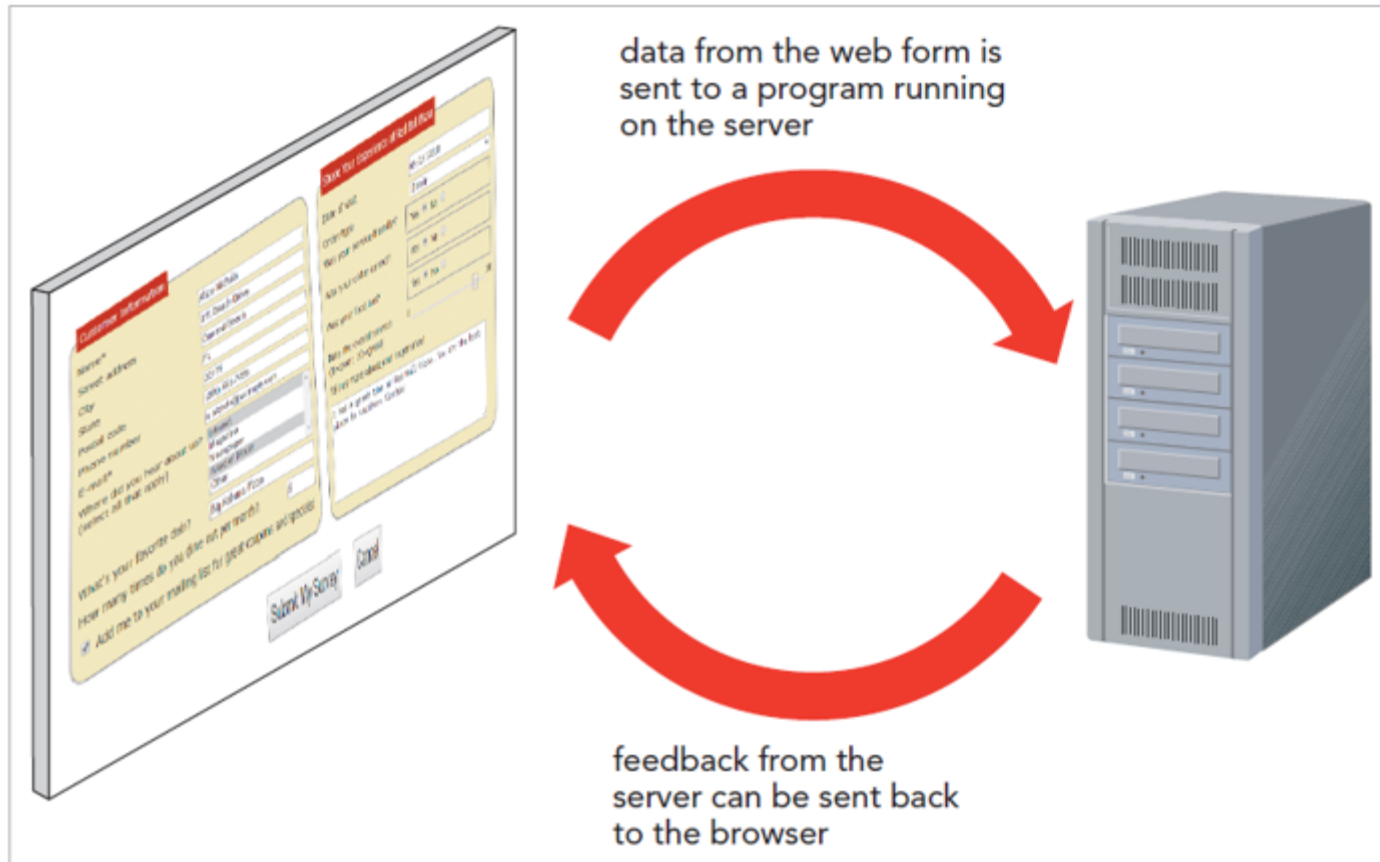
Forms and Advanced HTML

Dan Brandon, Ph.D., PMP

Forms

- A “form” is a group of **data capturing widgets (controls)** placed on an HTML page
- The form typically communicates the information (the user enters into the widgets) over the communication link to the Internet server via HTTP protocol encoded in a certain way (URLEncoded)
- Part of that HTTP protocol is the designation of a server side program which processes the form information; that program (in C/C++/Perl) is often called a **CGI** (Common Gateway Interface) **script** or **server page** (in PHP, ASP, etc.)

Forms (con't)



Forms (con't)

- Forms start with a `<form>` tag and end with `</form>`
- Within the `<form>` tag arguments are needed for the method of data transfer (post or get) to the server program (method=) and the name of the server program (action=):
 - `<form method="post" action="...path of server program">`
- For the form data to be **routed to the mail server**, the ACTION is "mail to: jdoe@mail.cbu.edu"

```
<form id="survey" action="http://www.example.com/redball/survey" method="post">  
</form>
```

location of server
processing the form

method by which
the form data is
sent to the server

Forms (con't)

- There are several widgets that can be placed on the form such as text fields, text areas (multi line), radio buttons, check boxes, selection lists, submit buttons, and reset buttons
- For text fields, submit buttons, reset buttons, and radio buttons, you use the INPUT tag
- For text fields, the INPUT tag has arguments for Type (“text”), Size, ID, and Name:
 - `<input type=“text” size=15 name=“City”>`
- The ‘name’ above does not show on the web page, it is used to identify the keyed-in info when it is sent to the server

Forms (con't)

- For buttons, the INPUT tag has arguments for Type, Name, and Value
- For radio buttons, use the same name:
 - `<input type="radio" name="paytype" value="cash">`
 - `<input type="radio" name="paytype" value="check">`
 - `<input type="radio" name="paytype" value="credit">`
- By default, **submit** buttons send the form data to the server, **reset** buttons clear the data
 - `<input type="submit" name="submit" value="submit">`
 - `<input type="reset" name="reset" value="reset">`

INPUT Types

Type Value	Control Displayed by the Browser
button	A button that can be clicked to perform an action
checkbox	A check box for yes/no or true/false responses
color	A widget from which users can select a color
date	A widget from which users can select a calendar date
datetime-local	A widget from which users can select a calendar date and time
email	An input box used for email addresses
file	A widget from which users can select a local file
Hidden	A control that is hidden from the user
image	An image that can be clicked to perform an action
month	A widget from which users can select a calendar month and year
number	A spin box from which users can select a numeric value

INPUT Types (con't)

Type Value	Control Displayed by the Browser
password	An input box in which the entry value is hidden by * symbols
radio	A radio or option button that can be clicked by the user
range	A slider from which users can select a numeric value within a defined range
reset	A button that can be clicked to reset the web form
search	A widget that can be used to search for a defined term
submit	A button that can be clicked to submit the form for processing
tel	An input box used for telephone numbers
text	An input box used for text entries (the default)
time	A widget from which users can select a time value
url	An input box used for entering URLs
week	A widget from which users can select a week value

Sample Form

- `<hr> Please enter the following customer data: <br>`
- `<form method="post" action="...path of cgi script">`
- `Name:<input type="text" size=30
name="cust_name">`
- `City:<input type="text" size=15 name="cust_city">`
- `State:<input type="text" size=2 name="cust_state">`
- `<br><br>`
- `<input type="submit" name="submit" value="submit">`
- `<input type="reset" name="reset" value="reset">`
- `</form><hr>`

Data Transfer Methods

- Both the **GET and POST method** is used to transfer data from client to server in HTTP protocol
- The main **difference between POST and GET method** is that **GET** carries request parameter appended in URL string while **POST** carries request parameter in message body
- POST is a more secure way of transferring data from a client but less efficient (for short strings)
- There is also the optional **enctype** attribute which specifies how the form data should be encoded as it is sent to the server

Encoding Type

Value	Description
<code>application/x-www-form-urlencoded</code>	The default format in which the data is encoded as a long text string with spaces replaced by the + character and special characters (including tabs and line breaks) replaced with their hexadecimal code values
<code>multipart/form-data</code>	The format used when uploading files in which no encoding of the data values occurs
<code>text/plain</code>	The format in which data is transferred as plain text with spaces replaced with the + character but no other encoding of the data values occurs

Tab Order and Access keys

- Form fields can also be assigned a “tab order”, “access key” via “label”/“id”
 - `<label for="ln" accesskey="L">`
 - Last Name:
 - `</label>`
 - `<input id="ln" name="lastname" type="text" tabindex="1">`
- There is also a “button” tag for making your own buttons and including images
- Uploading Files (type=file):
 - Subject Photo (Local File Path):`<input type="file" name="sPhoto" size="40" maxlength="60">`

[illegible]

Forms (con't)

Sample Form - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print W

Address  C:\mydata\470\html\sampleForm.htm

Search For > Search Web Get paid to fight spam Block Popups

Sample Form

Please enter the following info:

Name:

City:

State:





Form Position and Layout

- A `form` element can be placed anywhere within the body of a page
- Forms can contain page elements such as tables, paragraphs, inline images, and headings
- Forms can be laid out using a table(s) or CSS, and this is a common way to design a web form
- To really do anything significant with HTML forms you have to write the program on the server that will process the data from the form such as updating a database, querying a database, etc.

Common Layouts

The diagram illustrates two common form layouts for a 'Customer Information' form. The left layout is a 'one-column layout' where all fields are stacked vertically. The right layout is a 'two-column layout' where fields are arranged in two columns, with the first column containing labels and the second column containing input fields.

one-column layout

Customer Information

Name *

Street address

City

State

two-column layout

Customer Information

Name *

Street address

City

State (abbr.)

Field Sets

- A **Field set** is a group of widgets that share a common purpose
- Field sets are created using the `fieldset` element

```
<fieldset id="id">  
    content  
</fieldset>
```

where *id* identifies the field set and *content* is the form content within the field set

Field Sets (con't)

- A legend describes the content of a field set using the `legend` element

`<legend>text</legend>`

where *text* is the text of the legend

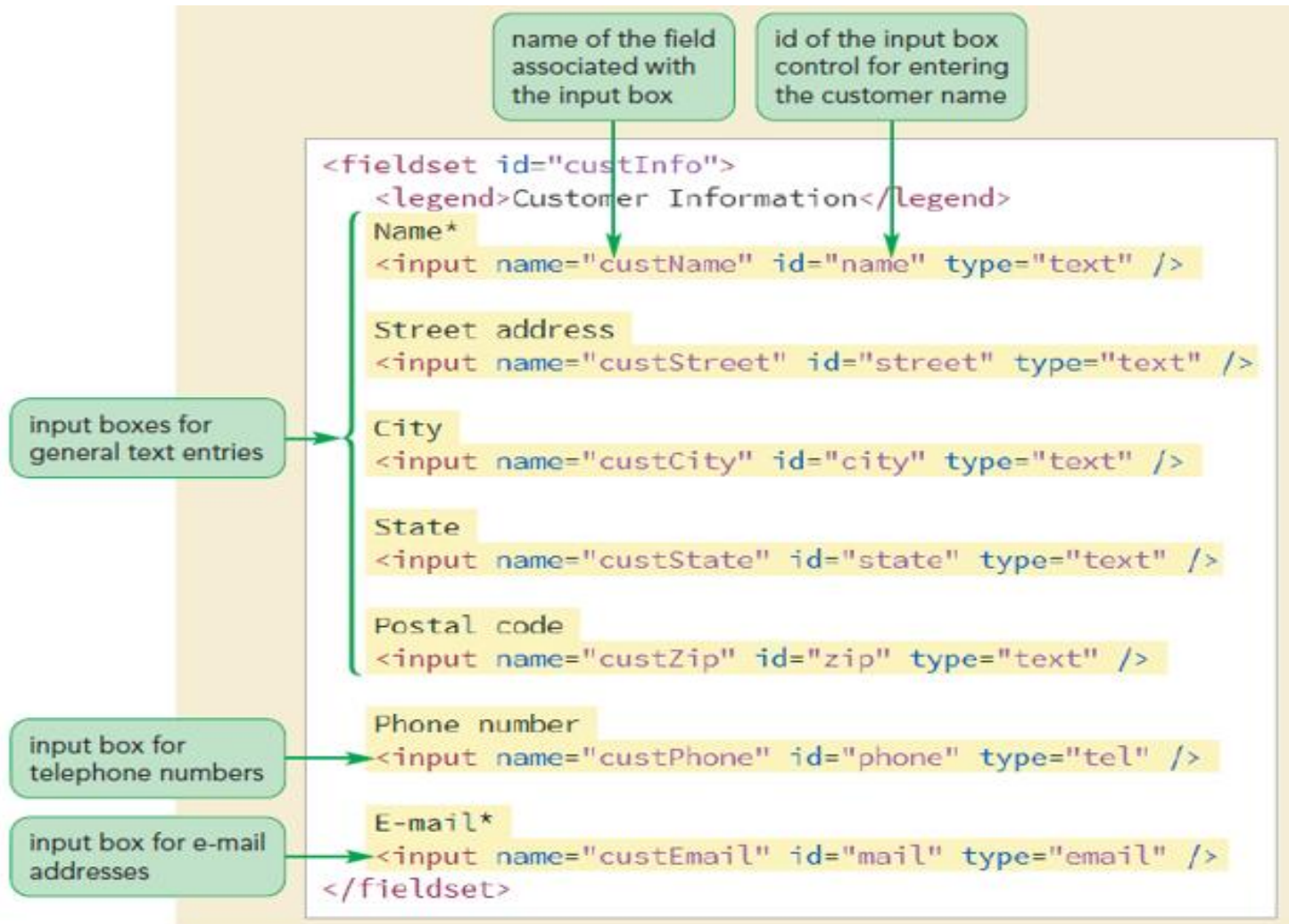
- The `legend` element contains only text and no nested elements
- By default, legends are placed in the top-left corner of the field set box and can be moved to a different location using the CSS positioning styles

Field Sets (con't)

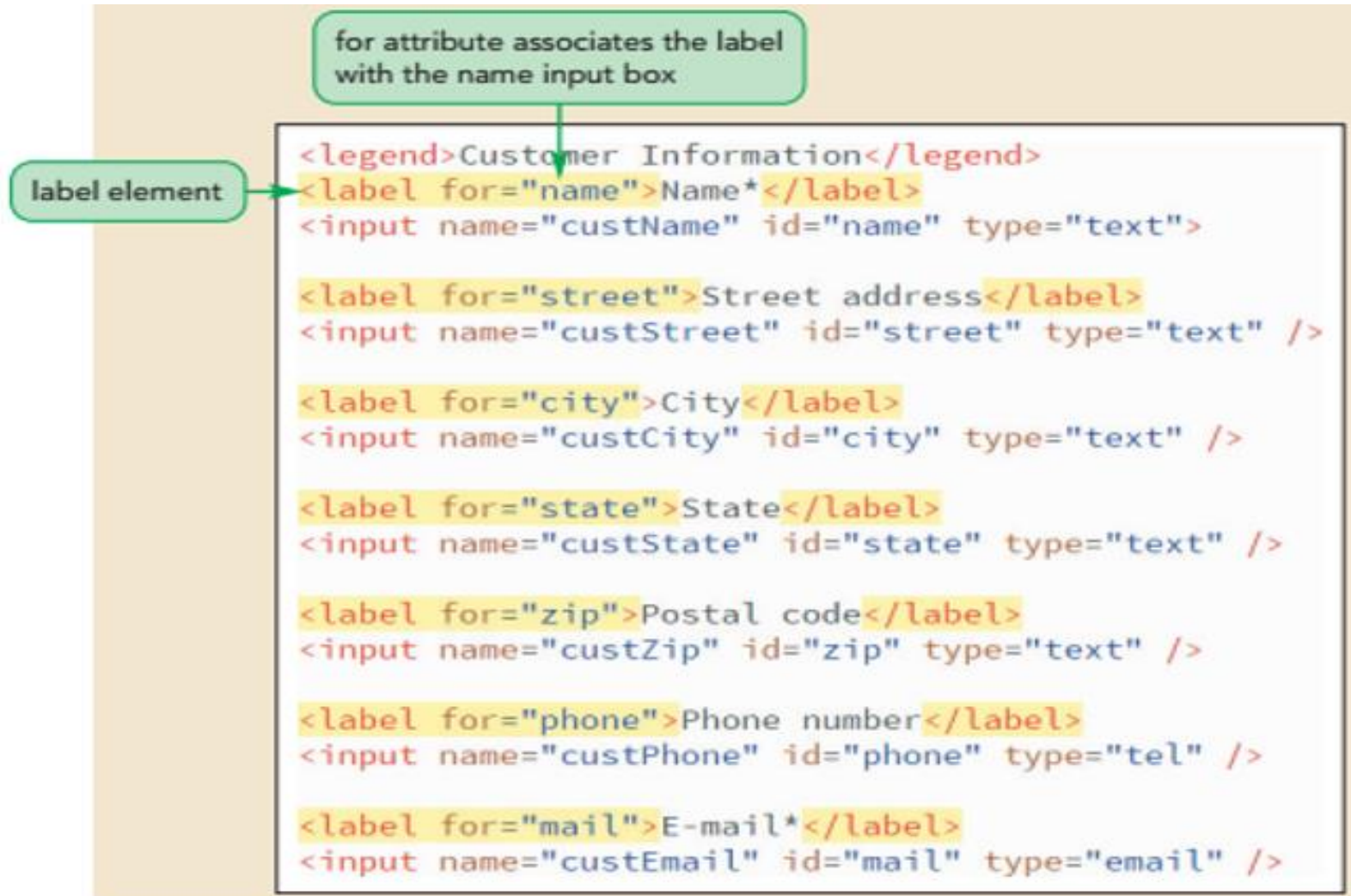
legend associated
with each field set

```
<form id="survey" action="http://www.example.com/redball/survey" method="post">
  <fieldset id="custInfo">
    <legend>Customer Information</legend>
  </fieldset>
  <fieldset id="expInfo">
    <legend>Share Your Experience at Red Ball Pizza</legend>
  </fieldset>
</form>
```

Field Sets and INPUT Widgets



Legend and Labels



Default Values

```
<div class="formRow">
  <label for="city">City</label>
  <input name="custCity" id="city" type="text" value="Ormond Beach" />
</div>

<div class="formRow">
  <label for="state">State</label>
  <input name="custState" id="state" type="text" value="FL" />
</div>
```

sets the default value
for the custCity field

sets the default value
for the custState field

Placeholders

```
<div class="formRow">
  <label for="name">Name*</label>
  <input name="custName" id="name" type="text" placeholder="first and last name" />
</div>
```

placeholder text for
the name input box

```
<div class="formRow">
  <label for="zip">Postal code</label>
  <input name="custZip" id="zip" type="text" placeholder="nnnnn (-nnnn)" />
</div>
```

placeholder text for
the zip input box

```
<div class="formRow">
  <label for="phone">Phone number</label>
  <input name="custPhone" id="phone" type="tel" placeholder="(nnn) nnn-nnnn" />
</div>
```

placeholder text for
the phone input box

Example Placeholders and Default Values

The diagram illustrates a form titled "Customer Information" with various input fields. Annotations explain the placeholders and default values for several fields:

- formatted field set legend**: Points to the "Customer Information" title.
- placeholder text for the name input box**: Points to the "Name*" field, which contains the placeholder text "first and last name".
- default value for the custCity field**: Points to the "City" field, which contains the default value "Ormond Beach".
- placeholder text for the phone input box**: Points to the "Phone number" field, which contains the placeholder text "(nnn) nnn-nnnn".
- placeholder text for zip input box**: Points to the "Postal code" field, which contains the placeholder text "nnnn (-nnn)".
- default value for the custState field**: Points to the "State" field, which contains the default value "FL".

The form fields and their current content are:

- Name*: first and last name
- Street address:
- City: Ormond Beach
- State: FL
- Postal code: nnnn (-nnn)
- Phone number: (nnn) nnn-nnnn
- E-mail*:

Testing Forms

[ACTION="http://www.danbrandon.com/echo.php">]

Echo site: <http://www.danbrandon.com/echo.php>

John Doe's Home Page - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Ed

Address <http://www.cbu.edu/~dbrandon/htmlab8.htm>

John Doe

 This is a picture of me.

I am so happy to be in this class.

I always do my reading and homework on time, and I never miss a class.

This home page contains information about my [address](#), my [hobbies](#), my [girlfriend](#), and my [pets](#).

Please enter the following info:

Name: City: State:

Form Echo - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit

Address <http://www.cbu.edu/~dbrandon/echo.php>

Form Echo

Key: Name Value: jane doe

Key: City Value: jackson

Key: State Value: ms

Key: submit Value: submit

Lab 9

■ Add a simple form, and use the “echo site” (on an earlier slide) to test out your form

The screenshot shows a Windows Internet Explorer browser window displaying a web page titled "John Doe's Home Page". The page content includes a greeting "This is my girlfriend!", a list of activities, and a section titled "My Pets" which contains a table of pet names and types. Below the table is a form for user input with fields for Name, City, and State, and submit/reset buttons. Overlaid on the right is a Notepad window showing the HTML source code for the page, including the table and form elements.

John Doe's Home Page - Windows Internet Explorer

Search

John Doe's Home Page

This is my girlfriend !

These are the things we like to do:

- Ride in my 1983 Ford Pinto
- Help pick up litter on our campus
- Eat in the school snack bar
- Sign up for 8am classes together
- Discuss student affairs with our school administration
- Study our math class notes together « What Fun »
- Surf the Net !

My Pets

Pet Types and Names	
Name	Type
Fred	Pigeon
Alice	Skunk
Mike	Aligator

Please enter the following info:

Name: City: State:

htmlab9.htm - Notepad

```
<ul type=circle>
  <li>Ride in my 1983 Ford Pinto
  <li>Help pick up litter on our campus
  <li>Eat in the school snack bar
  <li>Sign up for 8am classes together
  <li>Discuss student affairs with our school administration
  <li>Study our math class notes together &#171 What Fun &#187
  <li>Surf the Net !
</ul>
<br clear=right>
<p><hr>
<a name="pets"><h2>My Pets</h2>
<p>
<table border width=400>
  <caption align=top>Pet Types and Names</caption>
  <tr>
    <th width=40%>Name</th><th width=60%>Type</th>
  </tr>
  <tr>
    <td>Fred</td><td>Pigeon</td>
  </tr>
  <tr>
    <td>Alice</td><td>Skunk</td>
  </tr>
  <tr>
    <td>Mike</td><td>Aligator</td>
  </tr>
</table>
<p><hr>
Please enter the following info:<br><br>
<Form method="post" action="http://www.cbu.edu/~dbrandon/echo.php">
  Name: <input type="text" size=30 name="Name">
  City: <input type="text" size=15 name="City">
  State: <input type="text" size=2 name="State">
  <br><br>
  <input type="submit" name="submit" value="submit">
  <input type="reset" name="reset" value="reset">
</form>
</html>
```

Selection List

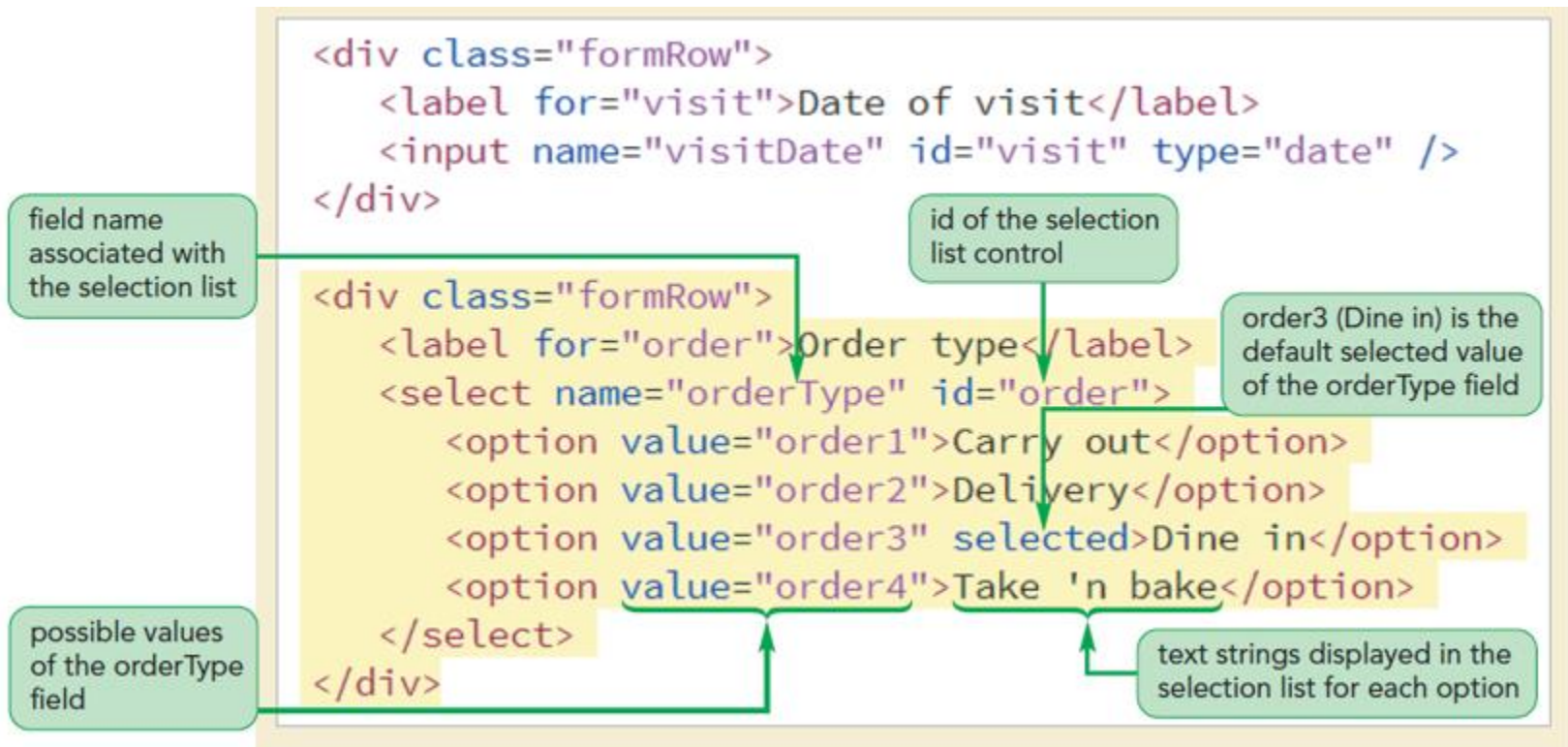
- A selection list is a list box that presents users with a group of possible values for the data field
- The list is created with the `select` and `option`

```
<select name="name">  
<option value="value1">text1</option>  
<option value="value2">text2</option>  
...  
</select>
```

■ Where

- `value1, value2,...` are the possible field values
- `text1, text2,...` are the text of the entries in the selection list that users see on the web form

Selection List (con't)



Selection List (con't)

- By default, a selection list appears as a drop-down list
- To display a selection list as a **scroll box**, use the `size` attribute to the `select` element:

```
<select size="value"> ... </select>
```

where *value* is the number of options that the selection list displays at one time
- By default, selection lists allow only one selection from the list of options
- To **allow more than one item to be selected**, add the `multiple` attribute

```
<select multiple> ... </select>
```

Selection List (con't)

- Two ways for users to **select multiple items**
 - For non-contiguous selection, press and hold the Ctrl key while making the selections
 - For contiguous selection, select the first item, hold the Shift key, and then select the last item in the range
- **Organize selection list** options by placing them in option groups using the `optgroup` element

```
<select>
  <optgroup label="label1">
    <option>text1</option>
    <option>text2</option>
  </optgroup>
</select>
```

SELECT and Option Group

```
<label for="appetizers">Starter Menu</label>
<select name="meal">
  <optgroup label="Appetizers">
    <option value="sms">Spicy Mozzarella Sticks</option>
    <option value="pr">Pepperoni Rolls</option>
    <option value="tr">Toasted Ravioli</option>
  </optgroup>
  <optgroup label="Salads">
    <option value="sms">Pasta Salad</option>
    <option value="tbs">Tuscan Bread Salad</option>
    <option value="pr">Caesar Salad</option>
  </optgroup>
</select>
```



Starter Menu

Spicy Mozzarella Sticks ▼
Appetizers
Spicy Mozzarella Sticks
Pepperoni Rolls
Toasted Ravioli
Salads
Pasta Salad
Tuscan Bread Salad
Caesar Salad

Radio (option) Buttons

- Unlike selection lists, the options appear as separate controls in the web form
- They are created with a group of `input` elements with a `type` attribute value of “radio”

```
<input name="name" value="value1" type="radio" />  
<input name="name" value="value2" type="radio" />  
...
```

where *value1*, *value2*, ... are the field values associated with each

- Set an option button to be selected as the default by adding the `checked` attribute to the `input` element
`<input name="name" type="radio" checked />`

Radio Buttons (con't)



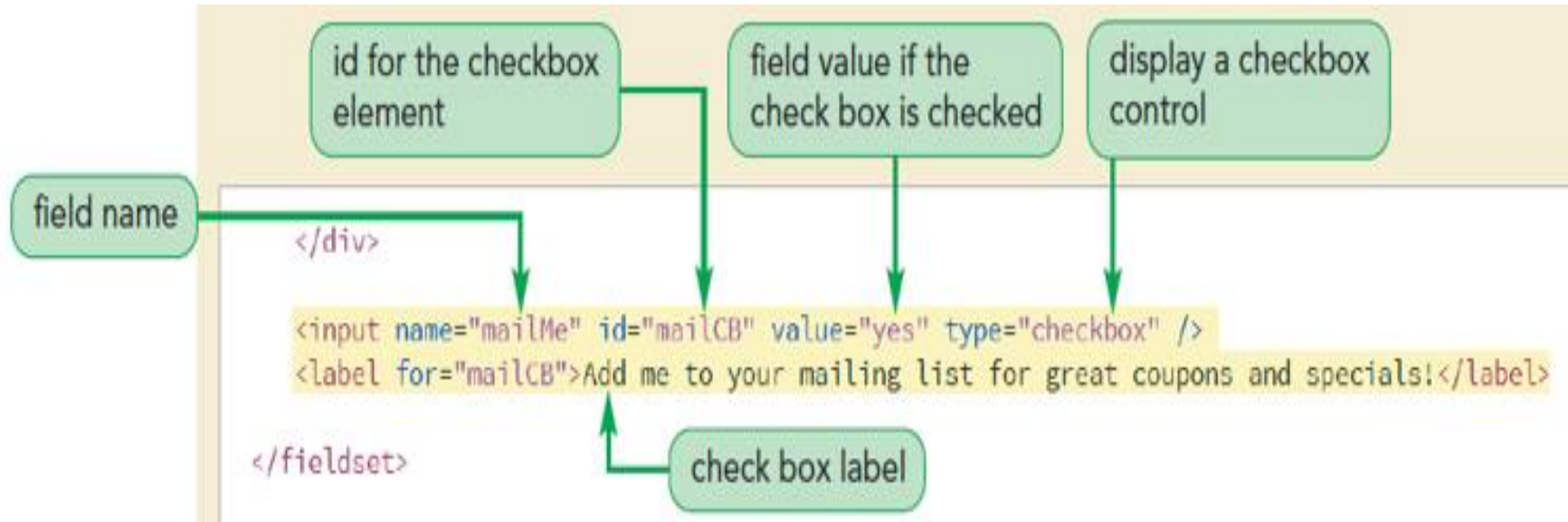
Check Boxes

- Check boxes are designed for fields that record the presence or absence of an something
- They are created using the `input` element with the `type` attribute set to “checkbox”

```
<input name="name" value="value"  
      type="checkbox" />
```

where `value` attribute contains the value of the field when the check box is checked and `type` attribute indicates a check box

Check Box (con't)



Text Area

- Text area is created using the `textarea` element

```
<textarea name="name">  
    text  
</textarea>
```

where *text* is the default value of the data field

- HTML supports the `rows` and `cols` attributes to set size

```
<textarea rows="value" cols="value">  
    ...  
</textarea>
```

Where the `rows` attribute specifies the number of lines in the text area box and the `cols` attribute specifies the number of characters per line

Text Area (con't)



DataList Suggestions

- A data list can be used to “suggest” some possible input:

The diagram illustrates how a data list is used in an HTML form. It features a code block with the following HTML structure:

```
<div class="formRow">
  <label for="dish">What's your favorite dish?</label>
  <input name="favDish" id="dish" type="text" list="dishType" />
  <datalist id="dishType">
    <option value="Anitpasto Pizza" />
    <option value="Big Kahuna Pizza" />
    <option value="BBQ Chicken Pizza" />
    <option value="Mediterranean Herb Pizza" />
    <option value="Pasta Rolls" />
    <option value="Pasto Artichoke Pizza" />
  </datalist>
</div>
```

Two callouts provide context:

- A green box on the left labeled "data list containing suggested values" has an arrow pointing to the `<datalist id="dishType">` block.
- A green box on the right labeled "links the favDish field to the dishType data list" has an arrow pointing to the `list="dishType"` attribute in the `<input>` tag.

Hidden Fields

- Hidden fields do not appear in the browser, but let the web page send information to the server
 - `<INPUT type="hidden" name="xxx" value="zzz">`
- Hidden fields let web developers include data that cannot be seen or modified by users when a form is submitted
- Hidden fields were used to maintain state between pages such as session info, login info, or storing what database record that needs to be updated when the form is submitted; server languages such as PHP provide better ways now

HTML 5 Extensions for Forms

- HTML 5 provides new types of form controls which may provide both error checking and/or entry aids such as “pickers”:
 - range (exact number not important)
 - number
 - telephone number
 - color
 - date, time, datetime (UTC), datetime-local, month, week
 - email
 - image
 - search
 - url
 - password
- New attributes and functions are also available

HTML5 Forms (con't)

The diagram illustrates an HTML5 form titled "Customer Information" and "Share Your Experience at Red Ball Pizza". It features several form controls, each labeled with a green box and an arrow pointing to it:

- input box**: Points to the "Name*" field.
- option buttons**: Points to the "Was your service friendly?", "Was your order correct?", and "Was your food hot?" questions, which each have "Yes" and "No" radio button options.
- calendar control**: Points to the "Date of visit" field, which includes a calendar icon.
- selection list box**: Points to the "Where did you hear about us?" dropdown menu.
- spin box**: Points to the "How many times do you dine out per month?" field, which includes up and down arrow controls.
- check box**: Points to the "Add me to your newsletter for great specials" checkbox.
- text area box**: Points to the "Tell us more about your experience!" text area.
- slider control**: Points to the "Rate your overall service (0 = poor; 10 = great)" slider, which ranges from 0 to 10.

The form is divided into two main sections:

- Customer Information**: Includes fields for Name*, Street address, City, State, Postal code, Phone number, E-mail*, and a dropdown for "Where did you hear about us?".
- Share Your Experience at Red Ball Pizza**: Includes a "Date of visit" field, a dropdown for "Order type", three questions with radio button options ("Was your service friendly?", "Was your order correct?", "Was your food hot?"), a slider for "Rate your overall service (0 = poor; 10 = great)", and a text area for "Tell us more about your experience!".

Time/Date Fields

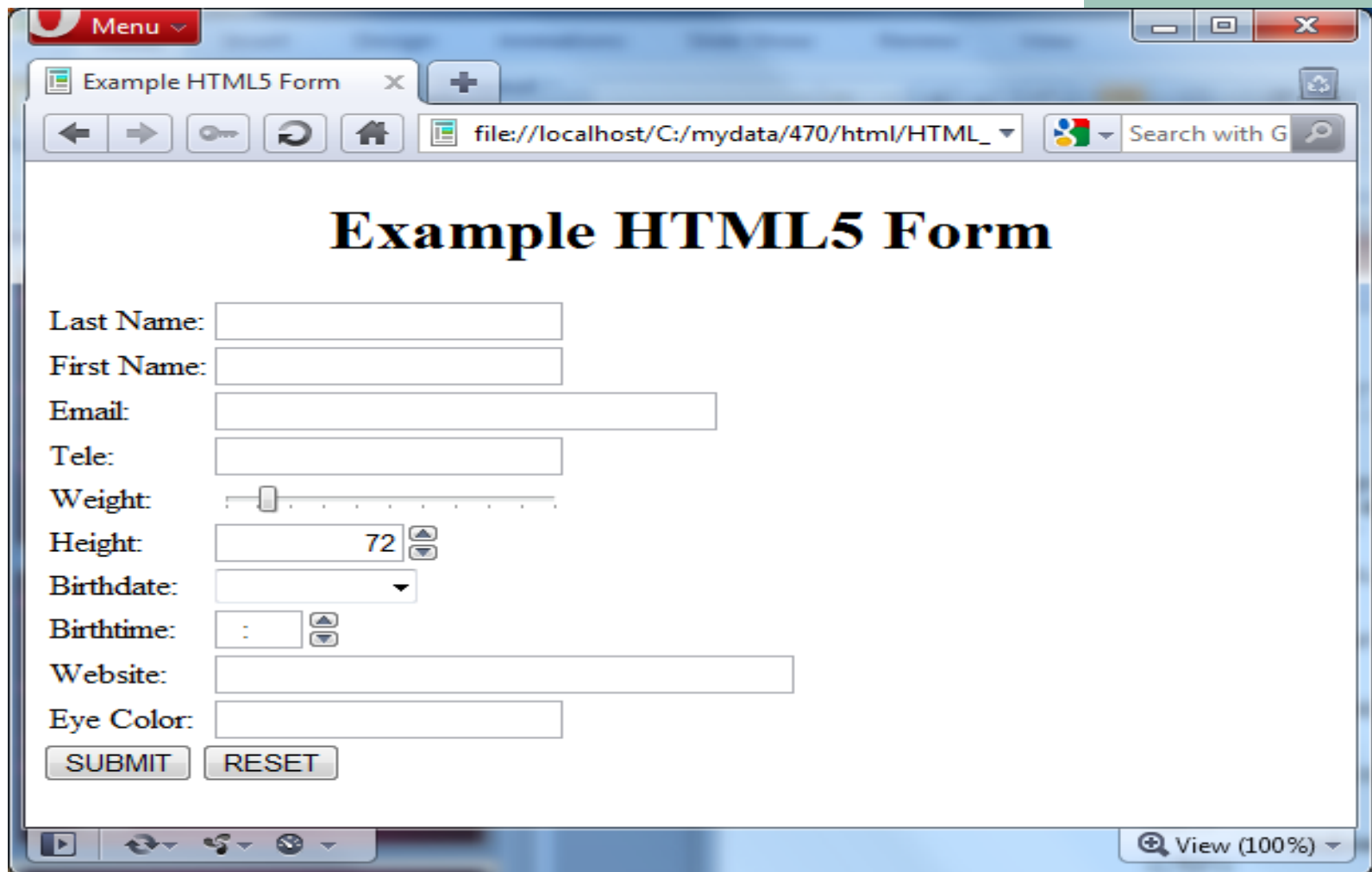
- Indicated using `type` attributes:
 - `date`
 - `time`
 - `datetime-local`
 - `month`
 - `week`

```
<fieldset id="expInfo">
  <legend>Share Your Experience at Red Ball Pizza</legend>

  <div class="formRow">
    <label for="visit">Date of visit</label>
    <input name="visitDate" id="visit" type="date" />
  </div>
</fieldset>
```

sets the data type of the visitDate field to "date"

Example HTML5 Form



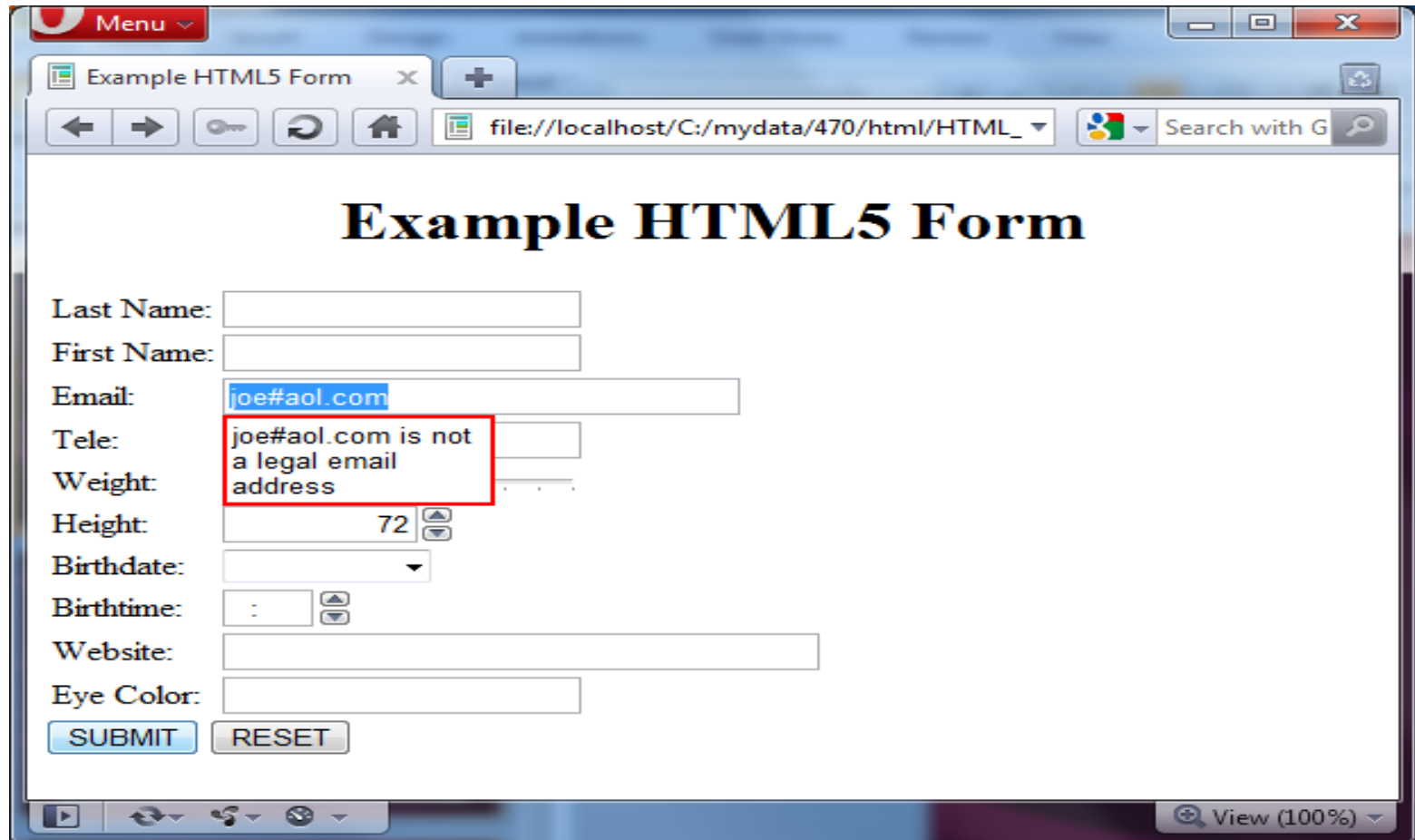
The screenshot shows a web browser window with a single tab titled "Example HTML5 Form". The address bar shows the file path: `file:///localhost/C:/mydata/470/html/HTML_`. The page content features a form with the following elements:

- Last Name:** A text input field.
- First Name:** A text input field.
- Email:** A text input field.
- Tele:** A text input field.
- Weight:** A range input (slider) with a value of 72.
- Height:** A text input field containing the value 72, with up and down arrow buttons on the right.
- Birthdate:** A date picker control.
- Birthtime:** A time picker control.
- Website:** A text input field.
- Eye Color:** A text input field.
- SUBMIT** and **RESET** buttons at the bottom of the form.

The browser's status bar at the bottom shows a "View (100%)" button.

Example HTML5 Form

[showing email validation]



The screenshot shows a web browser window with a single tab titled "Example HTML5 Form". The address bar displays the file path: `file:///localhost/C:/mydata/470/html/HTML_`. The page content features a title "Example HTML5 Form" and a form with the following fields:

- Last Name:
- First Name:
- Email:
- Tele:
- Weight:
- Height: (with up/down arrow icons)
- Birthdate:
- Birthtime: (with up/down arrow icons)
- Website:
- Eye Color:

At the bottom of the form are two buttons: "SUBMIT" and "RESET". A red rectangular box highlights the email field and the validation message: "joe#aol.com is not a legal email address". The browser's status bar at the bottom right shows "View (100%)".

Example HTML5 Form

[showing “date picker”]

Menu

Example HTML5 Form

file://localhost/C:/mydata/470/html/HTML_

Example HTML5 Form

Last Name:

First Name:

Email:

Tele:

Weight:

Height:

Birthdate:

Birthtime:

Website:

Eye Color:

January 2011

Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
52	27	28	29	30	31	1	2
1	3	4	5	6	7	8	9
2	10	11	12	13	14	15	16
3	17	18	19	20	21	22	23
4	24	25	26	27	28	29	30
5	31	1	2	3	4	5	6

Today None

View (100%)

Example HTML 5 Form Code

```
<!DOCTYPE html>
<html>

    <head>

        <meta charset="utf-8">
        <title>Example HTML5 Form</title>

    </head>
    <body>

        <h1 style="text-align:center">Example HTML5 Form</h1>
        <form method="post" action="echo.php">
            <table align="left">

                <tr>

                    <td>Last Name:</td><td><input name="lname" placeholder="Enter your last name" autofocus></td>

                </tr>
                <tr>

                    <td>First Name:</td><td><input name="fname" placeholder="Enter your first name"></td>

                </tr>
                <tr>

                    <td>Email:</td><td><input type="email" name="email" placeholder="Enter your email address"></td>

                </tr>
                <tr>

                    <td>Tele:</td><td><input type="tel" name="tele" placeholder="Enter your telephone number"></td>

                </tr>
                <tr>

                    <td>Weight:</td><td><input type="range" min="50" max="1000" step="10" value="170" name="weight" placeholder="Enter your
weight in pounds"></td>

                </tr>
                <tr>

                    <td>Height:</td><td><input type="number" min="36" max="96" step="2" value="72" name="height" placeholder="Enter your weight
in pounds"></td>

                </tr>
                <tr>

                    <td>Birthdate:</td><td><input type="date" name="bdate" placeholder="Enter your birthdate"></td>

                </tr>
                <tr>

                    <td>Birthtime:</td><td><input type="time" name="btime" placeholder="Enter your birth time (24 hour)"></td>

                </tr>
                <tr>

                    <td>Website:</td><td><input type="url" name="web" placeholder="Enter your web site"></td>

                </tr>
                <tr>

                    <td>Eye Color:</td><td><input type="color" name="ecolor" placeholder="Enter your eye color"></td>

                </tr>
            </table>
            <br clear="all">
            <input type="submit" value="SUBMIT">
            <input type="reset" value="RESET">
        </form>

    </body>

</html>
```

Pattern Attribute

- If none of these new input types suits your needs, HTML 5 provides the pattern attribute for input elements with type="text"
- The value of the pattern attribute is a **regular expression**, as defined in [ECMAScript](#) and used in JavaScript (discussed later in this course)
- For example, if one wanted to match a five-digit or nine-digit US ZIP code or a six-character Canadian postal code, one could use this pattern:
 - `([0-9]{5}(-[0-9]{4})?)|([A-Z][0-9][A-Z]\s+[0-9][A-Z][0-9])`

Pattern Attribute (con't)

- `<input type="text" name="postCode" required="required"`
- `pattern="([0-9]{5}(-[0-9]{4})?)([0-9][A-Z][0-9]\s+[A-Z][0-9][A-Z])"`
- `title="US: 99999-1234; Canadian: 0A1 B2C" />`

Form Validation

- **Validation:** Process of ensuring that a user has supplied valid data
- Types of validation
 - **Server-side validation** – validation occurs on the web server
 - **Client-side validation** – validation occurs in the user's browser
- Client-side validation is preferred where possible

Validation (con't)

- The first validation test is to verify if data is supplied for all the required data fields
- Add the `required` attribute to the control to identify the required data fields
- If a required field is left blank, the browser will not submit the form returning an error message to indicate the unavailability of data

Validation (con't)

- The syntax to define the `maxlength` attribute is `<input maxlength="value" />`

where *value* is the maximum number of characters in the field value

- Example:

```
<input name="custZip" maxlength="5" />
```

- The `maxlength` attribute does not distinguish between characters and digits

Validation (con't)

- A form fails the **built-in validation** test if the data values entered into a field do not match the field type
- Example:
 - A data field with the `number` type will be rejected if nonnumeric data is entered
 - Fields with `email` and `url` types will be rejected if a user provides an invalid e-mail address or text that does not match the format of a URL
- **To go beyond built-in validation, one needs to use JavaScript which is covered later in the course**

Validation (con't)

- Inline validation: The technique of immediate data validation and reporting of errors
- The `focus` pseudo-class is used to **change the display style of fields that currently contain invalid data**
 - **Focus:** The state in which an element has been clicked by the user, making it the active control on the form

Validation (con't)

Pseudo-Class	Matches
checked	A check box or option button that is selected or checked
default	A default control, such as the default option in a selection list
disabled	A control that is disabled
enabled	A control that is enabled
focus	A control that has the focus (is actively selected) in the form
indeterminate	A check box or option button whose toggle states (checked or unchecked) cannot be determined
in-range	A field whose value lies within the allowed range (between the min and max attribute values)
invalid	A field whose value fails the validation test
optional	A field that is optional (not required) in the form
out-of-range	A field whose value lies outside the allowed range (outside the min and max attribute values)
required	A field that is required in the form
valid	A field whose value passes the validation test

Validation (con't)

- Example: To create styles for all of the checked option buttons in the form, apply the checked **pseudo-class**

```
input[type="radio"]:checked {  
    styles  
}
```

where styles are the CSS styles applied to checked option buttons

Validation (con't)

- The `valid` and `invalid` pseudo-classes are used to format controls based on whether their field values pass/fail a validation test
- Example: To display `input` elements containing valid data with a light green background, use the following style rule:

```
input:valid {  
  background-color: rgb(220, 255,  
    220) ;  
}
```

Validation (con't)

- Example: To display `input` elements containing invalid data with a light red background with focus, use the following style rule:

```
input:focus:invalid {  
    background-color: rgb(255, 232,  
233) ;  
}
```

Validation (con't)

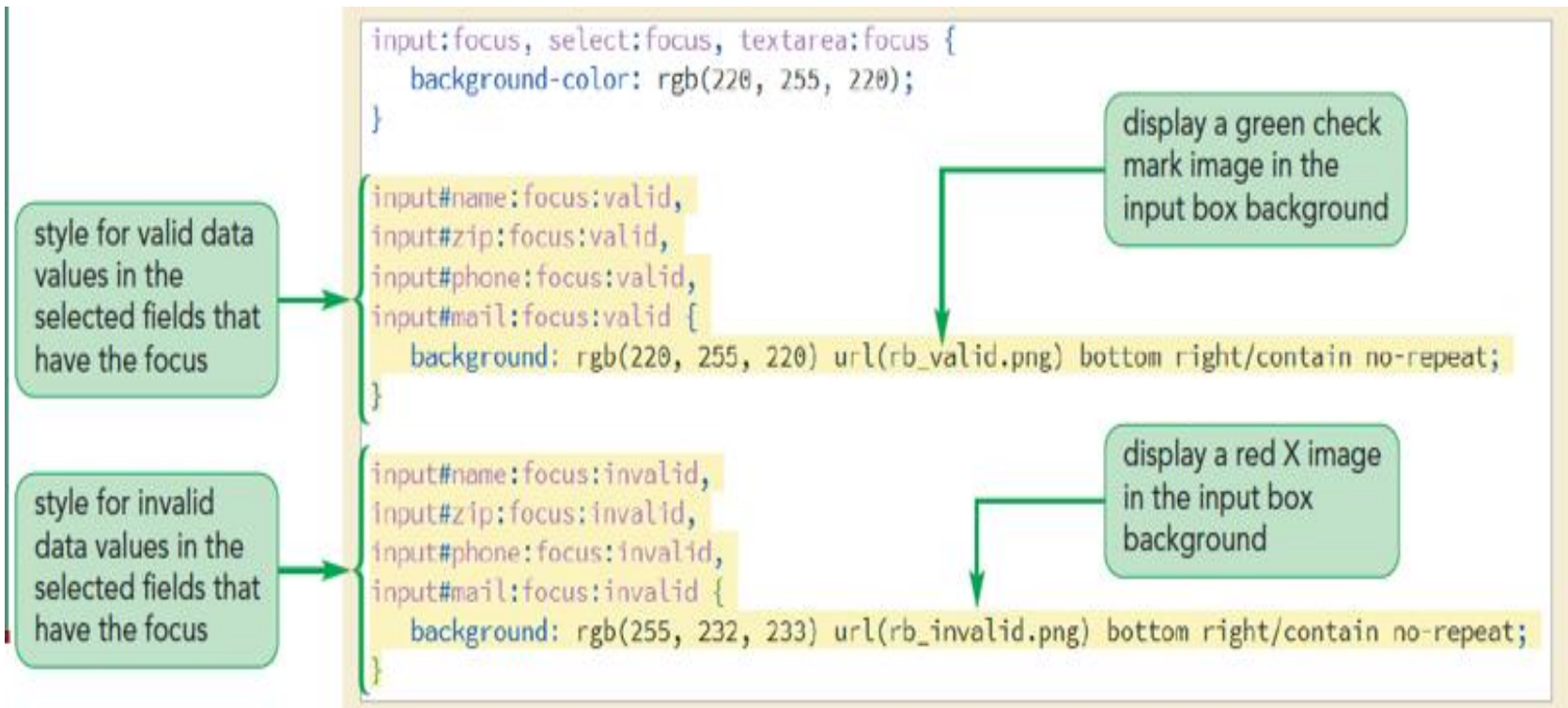


Image Maps

- Image maps allow you to associate links with portions of an image
- To associate an image with a map, simply set the property 'USEMAP' to the name of the image map:
 - `<img src=myImage.gif usemap="#myMap">`
- myMap works like the label for intrapage links
 - the imagemap img link must match a MAP tag

Defining the Map

- The map identifies areas within the image using relative pixel coordinates
- The relative coordinate systems has the origin (0,0) at the top left of the image
- MAP and nested AREA tags are used:
 - `<MAP NAME="myMap">`
 - `<AREA SHAPE=RECT COORDS="5,5,45,45" HREF="myDoc1">`
 - `<AREA SHAPE=RECT COORDS="105,105,195,195" HREF="myDoc2">`
 - `</MAP>`

Defining the Map (con't)

- For the **rectangular** area, use the x,y coordinates of the top left point followed by the x,y coordinates of the lower right point
- You can also define **circle** areas using the x,y of the center followed by the radius:
 - `<AREA SHAPE=CIRCLE COORDS=50,50,30" HREF="myDoc3">`
- You can also define **polygons** by giving a list of x,y pairs:
 - `<AREA SHAPE=POLY COORDS="0,100,0,50,100,100" HREF="myDoc4">`

Map Layout

- You can lay out the map areas manually
- Or there are commercial products which you download or execute over the Internet that allow you to visually define the image map
 - www.image-maps.com

www.image-maps.com

Online [image map](#) tool



www.image-maps.com
online image mapper

Free Online HTML Image Mapping Tool

Image Mapper Links

- Image Map Home
- Image Map Forum
- Notes, Updates, Changes
- United States Mapped
- Image Map Rollover Example
- Contact Us



Basic Online Image Mapping Tool

Basic online html **image mapping** tool. No software to download. Use images from the web or your PC. Load the image, map out your links and get the code! The Best and most Simple online **image mapper** and free.

New Now outsourcing **CSS** style image maps with **HTML** image maps.

[Tutorial on mapping myspace 2.0 image map - vdec](#)

Upload your image to be mapped

From your PC

- OR -

From a URL

[Use a test image](#)

[Share This](#)

[Post a Comment Email](#)

Image Mapper Friends

People who have helped the image map site out.

Web Development Tutorials	2008/08/31
Web Site Development Resources	2008/09/05
Client Side Image Map Example	2008/09/18
Making a Clientside Image Map	2008/10/29
Windows Tips and Tricks	2008/10/30

[Get your link here](#)

Image Map Comments From The

Posted By: Arvinc

2008-08-14 - 06:20:52

Rating: ★★☆☆☆ (Okay, but not great)
Great stuff! But, it can be greater.)

[Post Comment](#)

Posted By: Green Fitness

2008-08-16 - 11:54:38

Ads by Google

[Develop on Google Maps](#)

Develop your mapping application with Google Maps API Premier.
[www.google.com/center](#)

[Data Mapping Tool](#)

Powerful and easy to use Data Mapping. Online demos. Free Trial.
[www.sdsara.com/center](#)

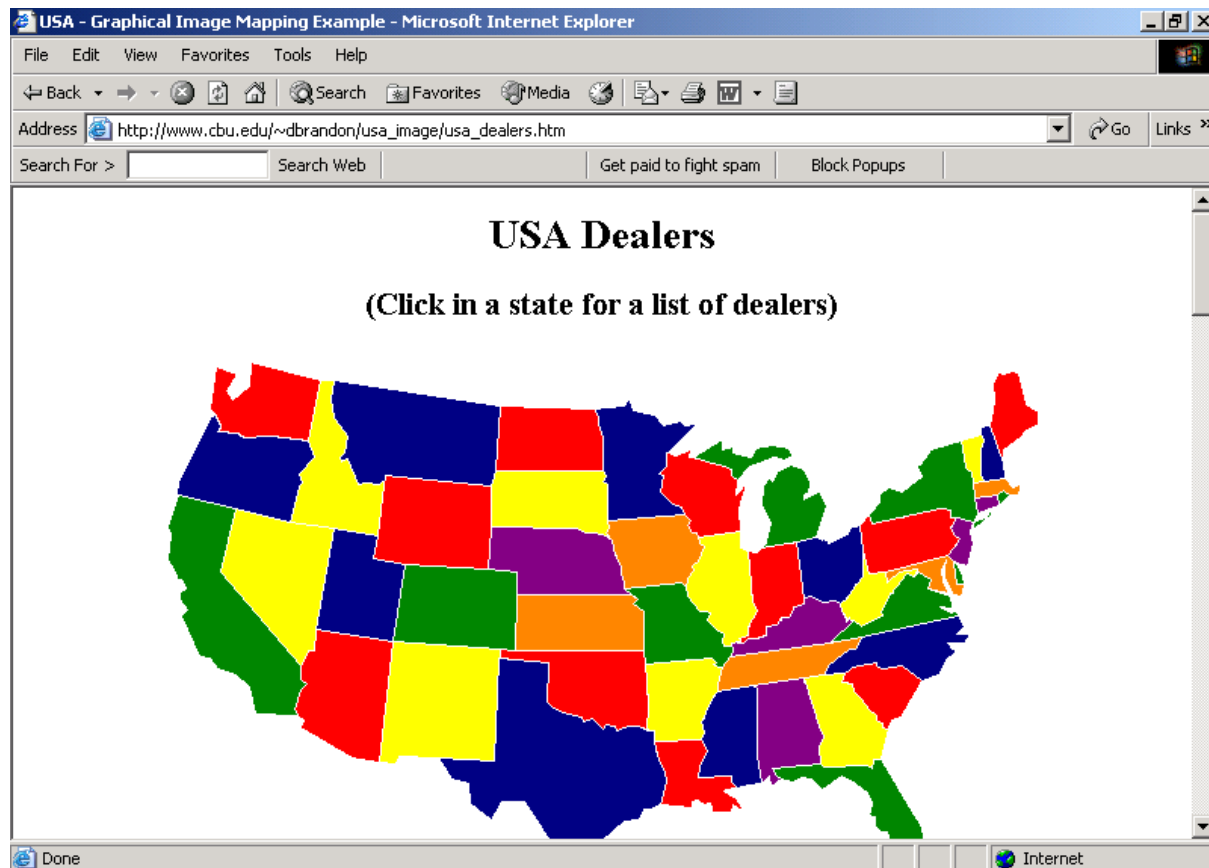
[Online Network Map](#)

Free Network Map. Identify rogue devices, & VPN servers.
[www.cul3y.com/Free-N](#)

[Green IT](#)

Best Practices in removing waste in the data center. Free whitepaper.
[www.ticvay.com](#)

http://facstaff.cbu.edu/dbrandon/usa_image/usa_dealers.htm



In Line Frames

- You can create “in line” frames in a similar manner as setting up images:
 - `<iframe src="myPage.htm" width="300" height="300" align="right"></iframe>`
- You can also give the frame a name and then use that name as a target to dynamically change the content of the frame (via JavaScript)

Lab 10

Add an inline frame to you page

John Doe's Home Page - Windows Internet Explorer

CaCourses\231\HTML_Labs\htmlab10.htm

John Doe

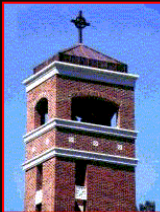
సుసావగత (Hello in Japanese)

This is a picture of me.

I am so happy to be in this class.

I always do my reading and homework on time, and I never miss a class.

This home page contains information about my [address](#), my [hobbies](#), my [girlfriend](#), and my [pets](#).



Click on tower to go to CBU home page !

[Click here to send me e-mail !](#)

[Click here to get a copy of my resume !](#)

My address

11 Peach Street

Example of In-Line Frame

 Here is some text in the frame.

htmlab10a.htm - Notepad

```
<html>
<head>
<title>IN LINE FRAME</title>
<body>
<h1 style="text-align:center">Example of In-Line Frame</h1>

Here is some text in the frame.
</body>
</html>
```

htmlab10.htm - Notepad

```
<body bgcolor="101010" text="ffffff" link="ff0000" vlink="0000ff">
<h1 style="text-align:center">John Doe</h1>

<p style="color:green">
This is a picture of me.
<p>
I am so happy to be in this class.
<p>
I always do my reading and homework on time
<p>
<br clear=left>
<hr>
<iframe src="htmlab10a.htm" width="400" height="200">
This home page contains information about my <a href="#hob">
my <a href="#hob">hobbies</a>, my <a href="#girl">girlfriend</a>
and my <a href="#pets">pets</a>.
<p>
<a href="http://www.cbu.edu">
Click on tower to go to CBU home page !
<p>
<a href="mailto:jdoe@cbu.edu">Click here to send me e-mail !
<p>
<a href="http://www.cbu.edu/~jdoe/resume.doc">Click here to get a copy of my resume !
<br clear=left>
<hr>
<a name="add"><h2>My address</h2>
<div style="font-style:italic">
11 Peach Street
Memphis, TN 38108
</div>
<hr>
<a name="hob"><h2>My hobbies</h2>
```

“Don’t let me be framed”

- `<script>`

- `if (frames) {`

- `if (top.frames.length>0)`

- `top.location.href=self.location;`

- `}`

- `</script>`

- JavaScript later in course

XMP

[deprecated]

- `<XMP>` indicates a block of text where all HTML tags are ignored. The only tag that is not ignored is the ending tag `</XMP>`:

- Here's how to do an anchor:
- `<XMP> Cool Dude
</XMP>`

■ *Produces*

- Here's how to do an anchor:
- `<A HREF="http://www.idocs.com">Cool Dude</A>`

■ Another way to do this is:

- Text in a `<pre>` element is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks
- `<PRE> Cool Dude
</PRE>`

HTML Information

- There is a “META” tag available to indicate certain key information about your page
- There are two types of META tags
 - HTTP-EQUIV
 - NAME attribute tags
- The HTTP-EQUIV tags control the actions of the browser
- The NAME meta tag takes NAME and CONTENT arguments
 - The NAME argument is a code for the type of information in the CONTENT argument
- These tags should go just before your BODY tag in between the <HEAD> and </HEAD> tags

HTML Information (con't)

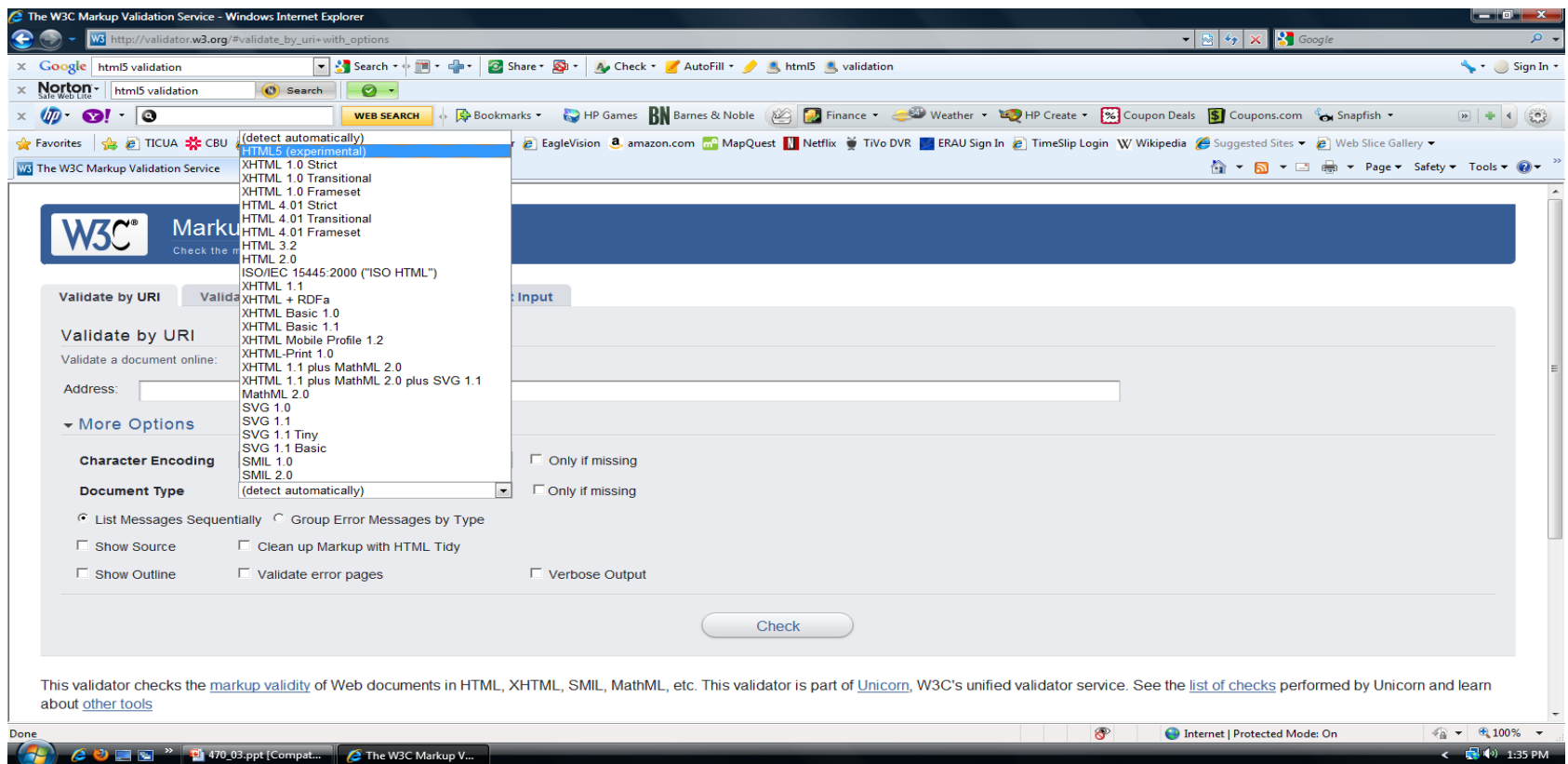
- There are several options for NAME, but one purpose of META is to **help search engines find out about your page:**
 - `<META NAME= "Author" CONTENTS="John Doe">`
 - `<META NAME= "Keywords" CONTENTS="college, training, computer, information technology">`
- Other NAMEs include date, generator, authoring.tool, description, etc.
- Some of these names are only used in proprietary authoring software or other applications such as Microsoft SharePoint or Microsoft Word

HTTP-EQUIV

- There are many of these other META capabilities
- They can control how and when your web pages are viewed including refresh rates, caching, slide shows, framing, setting cookies, rating content for violence, language, sex, nudity, etc.
- They can also override the normal server “content type” of ISO-8859-1 (Western, Latin-1):
 - `<META HTTP-EQUIV=“Content-Type” CONTENT=“text/html”; CHARSET=ISO-8859-5>`
 - This changes character set to Cyrillic
 - `HTTP-EQUIV=“Content-Script-Type”`
 - Sets default scripting language (later in course)
 - `HTTP-EQUIV=“Content-Style-Type”`
 - Sets default stylesheet (later in course)
 - `HTTP-EQUIV=“Content-Language”`
 - Sets the default natural language (later in course)

“Validating an HTML 5 Page”

- W3C provides a validation service for HTML:
 - <http://validator.w3.org/>



Protecting Your Site

- You may wish to protect your web site in several ways:
 - Keep images from being downloaded and saved
 - Keep source code (html & javascript) from being viewed or copied
 - Protect e-mail links so spammers can't get them with scanner programs
 - Protect all links so that scanners like Teleport or Webzip cannot store your website

Note: do not use any of these for your work in this class !!!

Protecting Your Site (con't)

- To disable local caching:

- `<meta http-equiv='pragma' content='no-cache'>`
- `<meta http-equiv='expires' content='-1'>`
- `<meta name='revisit-after' content='1 days'>`

- To disable robots indexing:

- `<meta name='robots' content='noindex'>`

- To disable frames:

- `<script>if(frames){if(top.frames.length>0)top.location.href=self.location;}</script>`

Disable Right Click

(see JavaScript later in course)

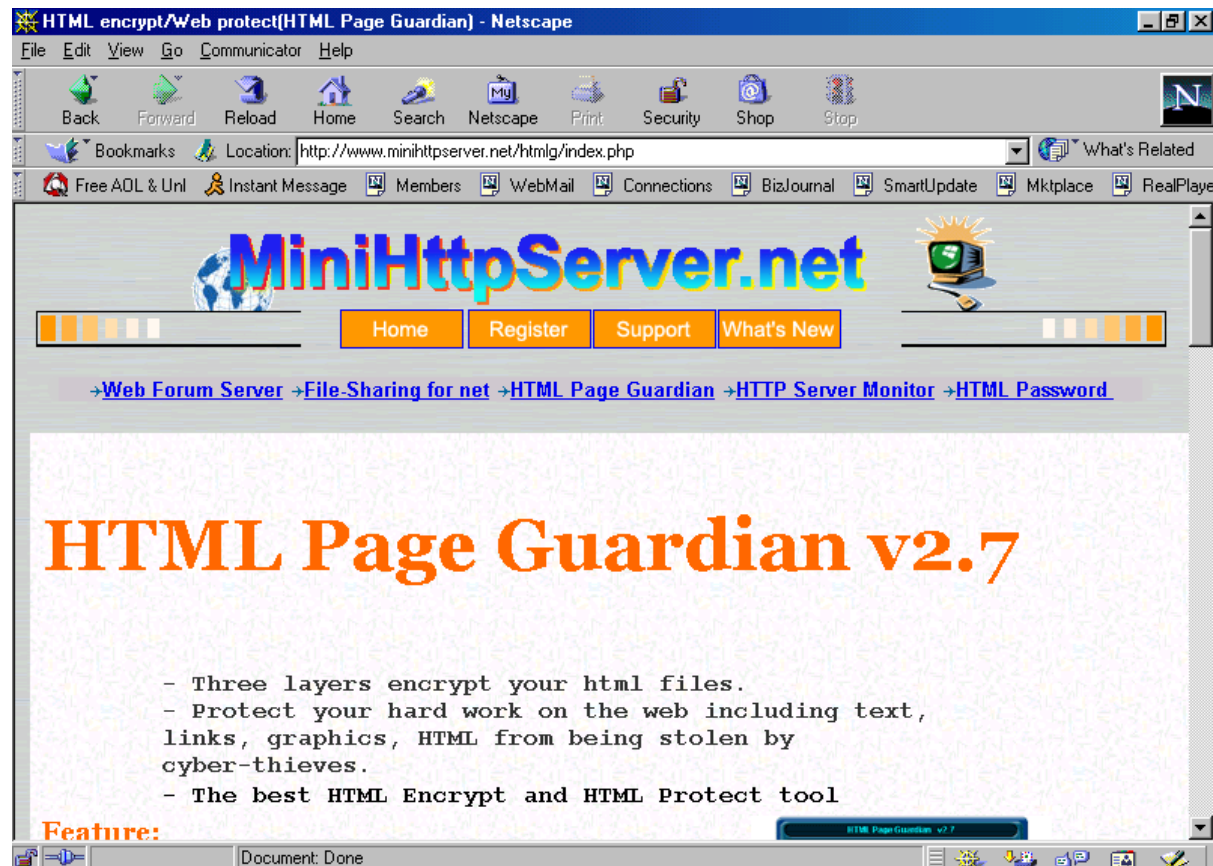
```
■ <SCRIPT language="JavaScript">
■     function clickIE() {
■         if (document.all) {alert("Copyright Source Code");return false;}
■     }

■     function clickNS(e) {
■         if (document.layers||(document.getElementById&&!document.all))
■             {if (e.which==2||e.which==3) {alert("Copyright Source Code");return
false;}}
■     }

■     if (document.layers) {
■         document.captureEvents(Event.MOUSEDOWN);document.onmousedown=clickNS;
■     }
■     else {
■         document.onmouseup=clickNS;document.oncontextmenu=clickIE;
■     }

■     document.oncontextmenu=new Function("return false")
■ </SCRIPT>
```

Protecting Your HTML [encryption]



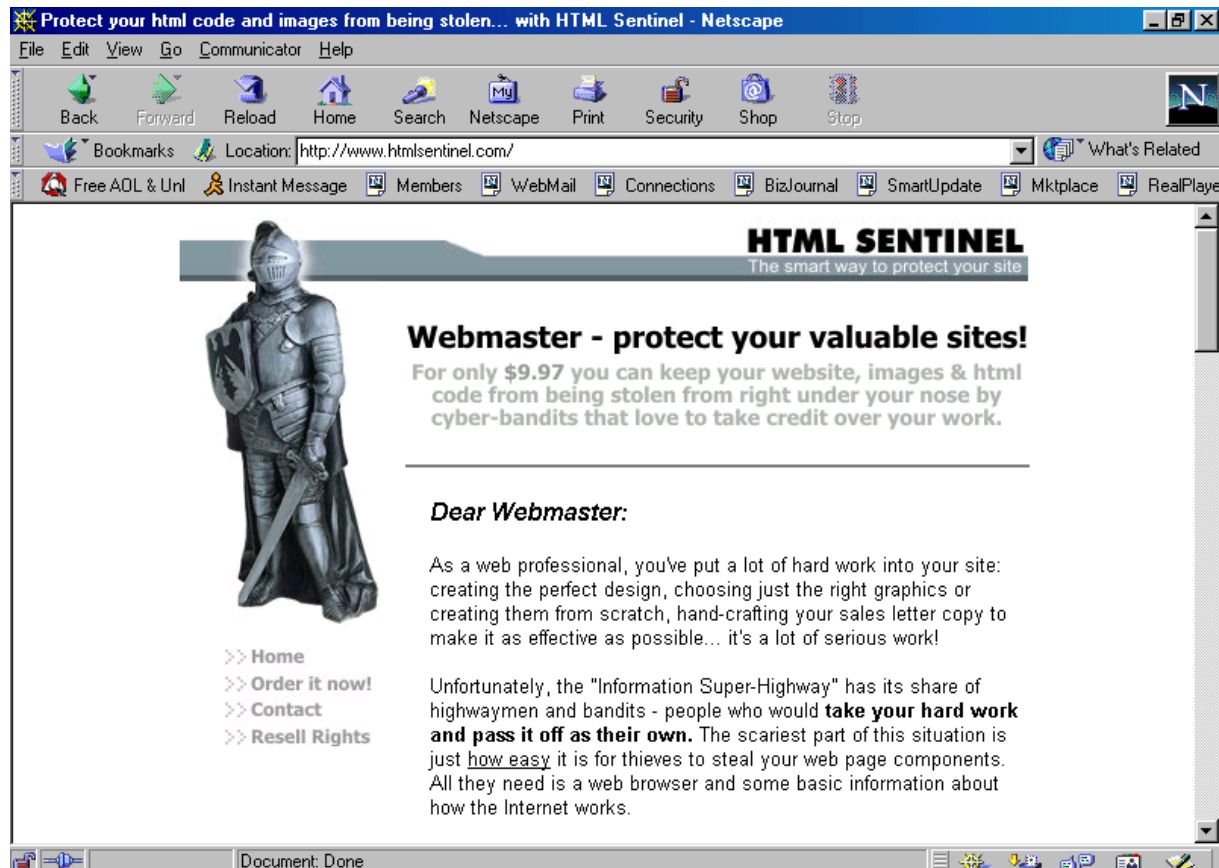
Protecting Your HTML (con't)



Protecting Your HTML (con't)



Protecting Your HTML (con't)



Posting Your Home Page

- Create a home page document in HTML using a general local text editor (or a specific HTML editor). Save the file with the extension “.htm” or “.html” (as a text file)
- View that HTML document in your browser (ie browser “open file”); go back to step one and revise as necessary
- If necessary: log onto CBU Server (Telnet) and create a subdirectory below your home directory called WWW
- If necessary, change its attributes to let everyone read it (chmod a+rx WWW); make sure your home directory has the execute permission set also:

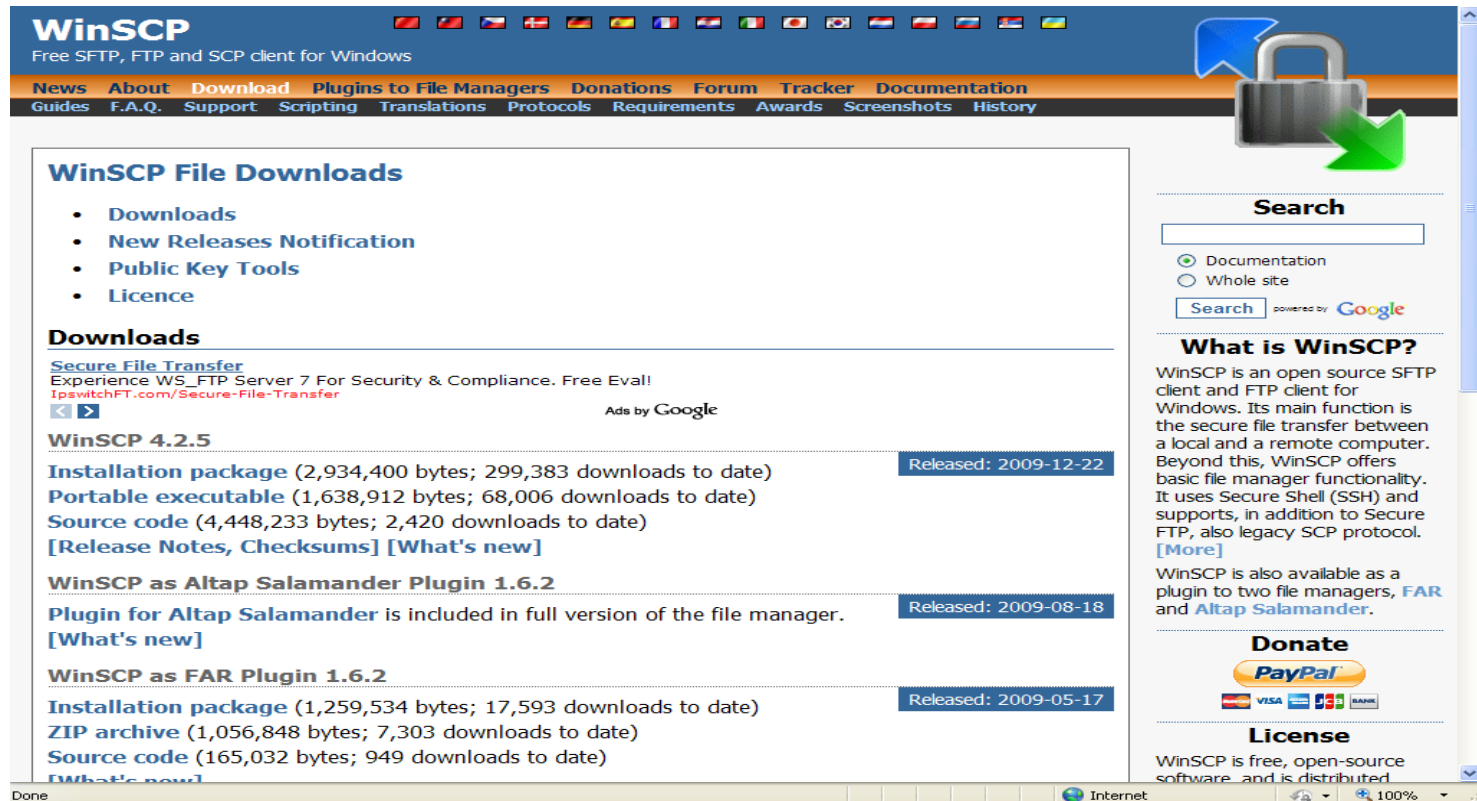
```
chmod 711 /home/stu1/jdoe
```

Posting a Home Page (con't)

- Copy the home page (all pages and images used) to CBU Server (via FTP or WinSCP)
- Name the home page Welcome.html (or index.html), put it in that WWW subdirectory, and, if necessary, change its attributes so that anyone can read it (“chmod a+r Welcome.html” or “chmod 755 Welcome.html”)
- Your URL will be: `stu.cbu.edu/~jdoe`
- See: <https://www.cbu.edu/it-services>

WinSCP (free download)

[<http://winscp.net/eng/download.php>]



The screenshot shows the WinSCP website interface. At the top, there's a navigation bar with links: News, About, Download, Plugins to File Managers, Donations, Forum, Tracker, Documentation, Guides, F.A.Q., Support, Scripting, Translations, Protocols, Requirements, Awards, Screenshots, and History. Below this, the main content area is titled "WinSCP File Downloads" and lists several download options: "Secure File Transfer" (Experience WS_FTP Server 7 For Security & Compliance. Free Eval! IpswitchFT.com/Secure-File-Transfer), "WinSCP 4.2.5" (Installation package, Portable executable, Source code), "WinSCP as Altap Salamander Plugin 1.6.2" (Plugin for Altap Salamander), and "WinSCP as FAR Plugin 1.6.2" (Installation package, ZIP archive, Source code). Each download link includes the file size and the number of downloads to date. On the right side, there's a search bar, a "What is WinSCP?" section explaining it's an open-source SFTP client, a "Donate" section with a PayPal logo, and a "License" section stating it's free, open-source software. The website is displayed in a browser window with a taskbar at the bottom showing "Done" and "Internet".

WinSCP
Free SFTP, FTP and SCP client for Windows

[News](#) [About](#) [Download](#) [Plugins to File Managers](#) [Donations](#) [Forum](#) [Tracker](#) [Documentation](#)
[Guides](#) [F.A.Q.](#) [Support](#) [Scripting](#) [Translations](#) [Protocols](#) [Requirements](#) [Awards](#) [Screenshots](#) [History](#)

WinSCP File Downloads

- [Downloads](#)
- [New Releases Notification](#)
- [Public Key Tools](#)
- [Licence](#)

Downloads

[Secure File Transfer](#)
Experience WS_FTP Server 7 For Security & Compliance. Free Eval!
IpswitchFT.com/Secure-File-Transfer

Ads by Google

WinSCP 4.2.5

Download Link	Released
Installation package (2,934,400 bytes; 299,383 downloads to date)	2009-12-22
Portable executable (1,638,912 bytes; 68,006 downloads to date)	
Source code (4,448,233 bytes; 2,420 downloads to date)	

[\[Release Notes, Checksums\]](#) [\[What's new\]](#)

WinSCP as Altap Salamander Plugin 1.6.2

Download Link	Released
Plugin for Altap Salamander is included in full version of the file manager.	2009-08-18

[\[What's new\]](#)

WinSCP as FAR Plugin 1.6.2

Download Link	Released
Installation package (1,259,534 bytes; 17,593 downloads to date)	2009-05-17
ZIP archive (1,056,848 bytes; 7,303 downloads to date)	
Source code (165,032 bytes; 949 downloads to date)	

[\[What's new\]](#)

Search

☒ Documentation
☐ Whole site

powered by [Google](#)

What is WinSCP?

WinSCP is an open source SFTP client and FTP client for Windows. Its main function is the secure file transfer between a local and a remote computer. Beyond this, WinSCP offers basic file manager functionality. It uses Secure Shell (SSH) and supports, in addition to Secure FTP, also legacy SCP protocol. [\[More\]](#)

WinSCP is also available as a plugin to two file managers, [FAR](#) and [Altap Salamander](#).

Donate

[PayPal](#)



License

WinSCP is free, open-source software and is distributed

Loaded on CBU lab computers !

Posting a Home Page (con't)

- Unix directories and files have **three rights in regards to three entities**: the owner, groups, everyone else
- Check that the rights/properties are set properly for your Sheba WWW directly (as well as all the files inside of the WWW directory) as shown below (rwxrwxrwx, octal 777) or 755



2/23/2017 7:32:27 AM

rwxrwxrwx

dbrand...

- For a directory (folder), the three permissions/properties (r,w,x) are: view, create/delete, enter/run
- For a file, the three permissions/properties are: read, write, execute

Posting a Home Page (con't)

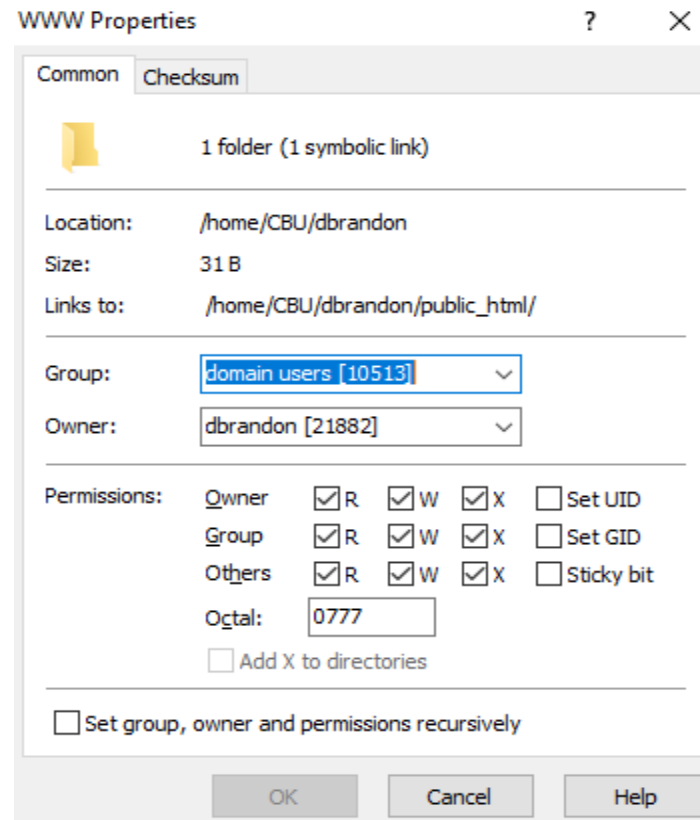
- Files inside of WWW directory should also be set to 777, 755, or 644 (rw-r--r--) for read only property

 Welcome.html	4,815 B	10/30/2019 5:58:14 PM	rw-r--r--
--	---------	-----------------------	-----------

- If they are correct and you still get an error viewing your page in a browser, go to the CBU ITS Help desk as there may be a problem with the way your account is set up

Posting a Home Page (con't)

- In WinSCP, one can change directory or file properties/permissions:



Posting a Home Page (con't)

- Setting permission for a file inside WWW folder:

The screenshot displays a Windows File Explorer window with two panes. The left pane shows the 'C:\Courses' directory, and the right pane shows the '/home/CBU/dbrandon/public_html' directory. A file named 'Welcome.html' is selected in the right pane. A 'Welcome.html Properties' dialog box is open, showing the 'Security' tab. The 'Group' is set to 'domain users [10513]' and the 'Owner' is 'dbrandon [21882]'. The 'Permissions' section shows 'Owner' with 'Full control' (R, W, X) and 'Group' with 'Full control' (R, W, X). The 'Others' section shows 'Full control' (R, W, X). The 'Octal' permission is set to '0644'. A green arrow points to the 'Welcome.html' file in the right pane, and another green arrow points to the 'Octal' field in the 'Permissions' section of the dialog box.

Name	Ext	Size	Changed	Rights	Owner
unchained_melody.m...		31,030 B	5/9/2005 4:36:16 PM	rw-r--r--	dbrandon...
usa.gif		4,978 B	5/9/2005 4:36:16 PM	rw-r--r--	dbrandon...
UserInterface.doc		55,808 B	5/9/2005 4:36:17 PM	rw-r--r--	dbrandon...
valentin.gif		4,261 B	5/9/2005 4:36:17 PM	rw-r--r--	dbrandon...
velcro1.jpg		4,169 B	5/30/2008 10:39:33 AM	rw-r--r--	dbrandon...
velcro2.jpg		2,857 B	5/30/2008 10:39:35 AM	rw-r--r--	dbrandon...
VerticalScroll.class		4,290 B	5/9/2005 4:36:17 PM	rw-r--r--	dbrandon...
VerticalScroll.html		154 B	5/9/2005 4:36:17 PM	rw-r--r--	dbrandon...
WebLocal.doc		676 KiB	5/9/2005 4:36:18 PM	rw-r--r--	dbrandon...
WebQuiz102.xml		254 B	5/9/2005 4:36:18 PM	rw-r--r--	dbrandon...
Welcome.html		4,815 B	10/30/2019 5:58:14 PM	rw-r--r--	dbrandon...
Welcome_Old.ht					rand...
whats_happen.c					rand...
Who_Can_Explai					rand...
whoami.cgi					rand...
window.gif					rand...
WriteServerFile.c					rand...
WriteServerFile.h					rand...
WS_FTP.LOG					rand...
XITM694.html					rand...
xmas.gif					rand...
yawn.gif					rand...
znewfile.txt					rand...

0 B of 75,170 KiB in 0 of 71

4,815 B of 369 MiB in

AG Wed 3/31/2021 11:21 PM
To: Daniel Brandon
Good Evening Prof

Jessica L King
NEW Starfish Training Module: F... Tue 3:06 PM

0:07:47

Posting a Home Page (con't)

- **One can also set permissions via the command line via chmod**
 - **chmod {a,u,g,o} {+,-} {r,w,x} files**

The plus ("+") sign indicates give permission. The minus ("-") sign indicates remove permission.
- **Permission examples:**
 - **chmod a+r** files are readable by all
 - **chmod a-r** files cancels the ability for all to read the file
 - **chmod a-rwx** cancels all access for all
 - **chmod g+rw** files give the group read and write permission
 - **chmod u+rwx** files give the owner all permissions
 - **chmod og+rw** files give the world and the group read and write permission
- : For example, If you want all people to read the home page of your Web site (the files which reside in your public_html directory), but do not want to give permission for viewers to alter your files, you would type:
- **chmod a+r index.html**

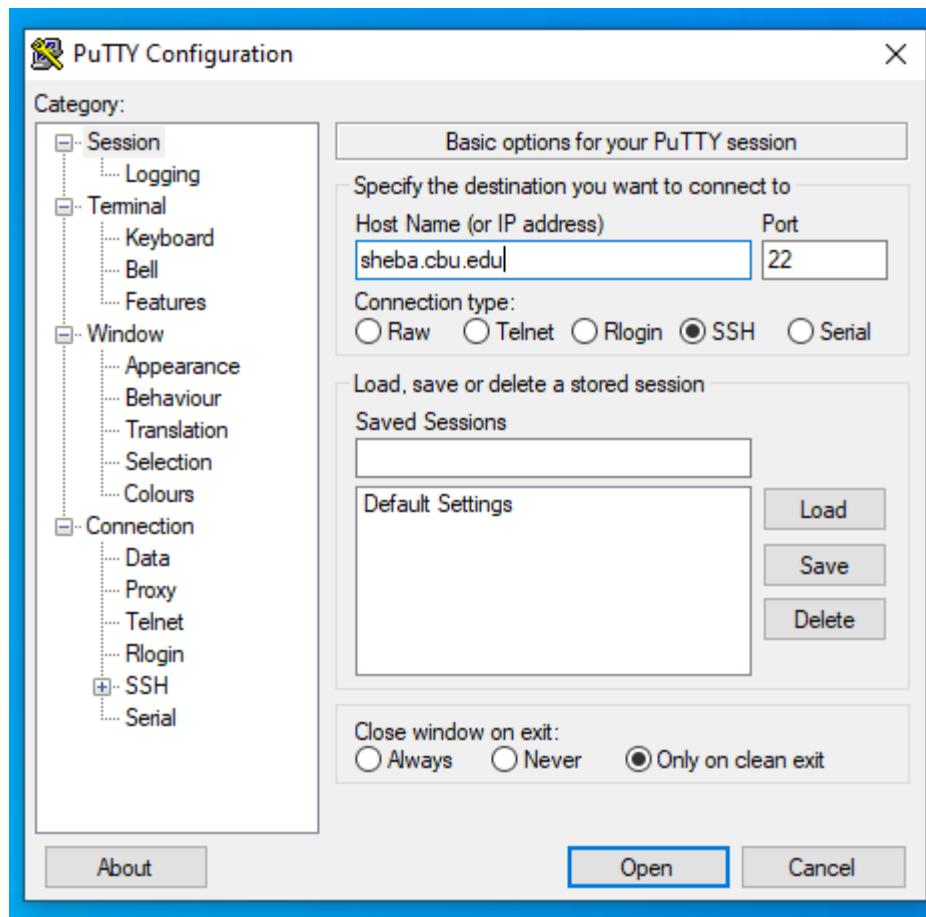
Posting a Home Page (con't)

- Home directory permissions
 - Now on the student server (sheba), the student's home directory permissions have to be set to 755 in order to see their web page (not get a "forbidden" error)
 - This may or may not have been done when your account was first created
 - This is in addition to the 755 permissions on the WWW directory and all files and folders inside of it (that can be done with WinSCP)
 - Setting the home directory permissions cannot be done in WinSCP because one can't back out of their home directory (go up the directory tree) to change it
 - One can use the PuTTY program to log onto your account on sheba and issue unix/linux commands
 - PuTTY is a program that is loaded onto the CBU computers, and it is free to install on your own device

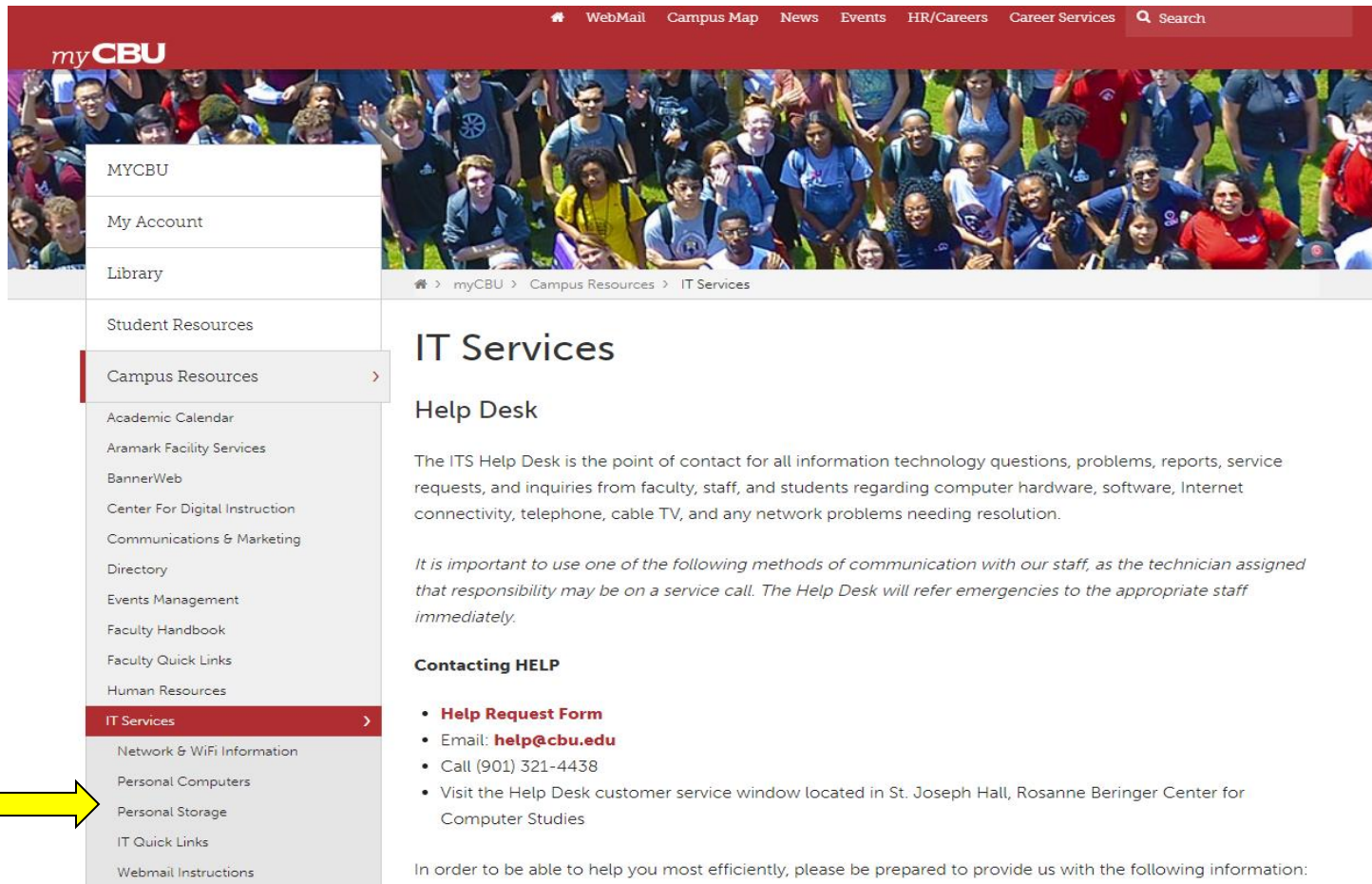
Posting a Home Page (con't)

- When PuTTY is started you specify the target server as sheba.cbu.edu
- Then you enter your CBU username and password, and now you are ready to enter unix commands
- You can check permissions on the home directory with the command:
 - `ls -ld`
- You should see
 - `drwxr-xr-x`
- Which is 755; If not, you can set permissions correctly with the command:
 - `chmod 755 ~`

Posting a Home Page (con't)



<https://www.cbu.edu/it-services>



The screenshot shows the myCBU website interface. At the top, there is a navigation bar with links for WebMail, Campus Map, News, Events, HR/Careers, Career Services, and a Search bar. Below this is a large banner image of a diverse group of students. On the left, a sidebar menu is open, listing various resources. A yellow arrow points to the 'IT Services' link in this menu. The main content area on the right is titled 'IT Services' and contains a 'Help Desk' section with a description of its purpose and contact information. Below this is a 'Contacting HELP' section with a list of contact methods.

myCBU

WebMail Campus Map News Events HR/Careers Career Services Search

MYCBU

My Account

Library

Student Resources

Campus Resources >

Academic Calendar

Aramark Facility Services

BannerWeb

Center For Digital Instruction

Communications & Marketing

Directory

Events Management

Faculty Handbook

Faculty Quick Links

Human Resources

IT Services >

Network & WiFi Information

Personal Computers

Personal Storage

IT Quick Links

Webmail Instructions

IT Services

Help Desk

The ITS Help Desk is the point of contact for all information technology questions, problems, reports, service requests, and inquiries from faculty, staff, and students regarding computer hardware, software, Internet connectivity, telephone, cable TV, and any network problems needing resolution.

It is important to use one of the following methods of communication with our staff, as the technician assigned that responsibility may be on a service call. The Help Desk will refer emergencies to the appropriate staff immediately.

Contacting HELP

- **Help Request Form**
- Email: help@cbu.edu
- Call (901) 321-4438
- Visit the Help Desk customer service window located in St. Joseph Hall, Rosanne Beringer Center for Computer Studies

In order to be able to help you most efficiently, please be prepared to provide us with the following information:

Getting Your Page “Noticed”

- Use “META” HTML tags to describe the content of a document within the <HEAD> area, so that search engines can readily get that information
 - META uses name/value pairs
 - Typical names are: Author, Keywords, Generator, Classification, Description
 - Examples:
 - <META name=“Author” content=“John Doe”>
 - <META name=“Keywords” content=“lady’s man, stud, buff, cool, bad”>

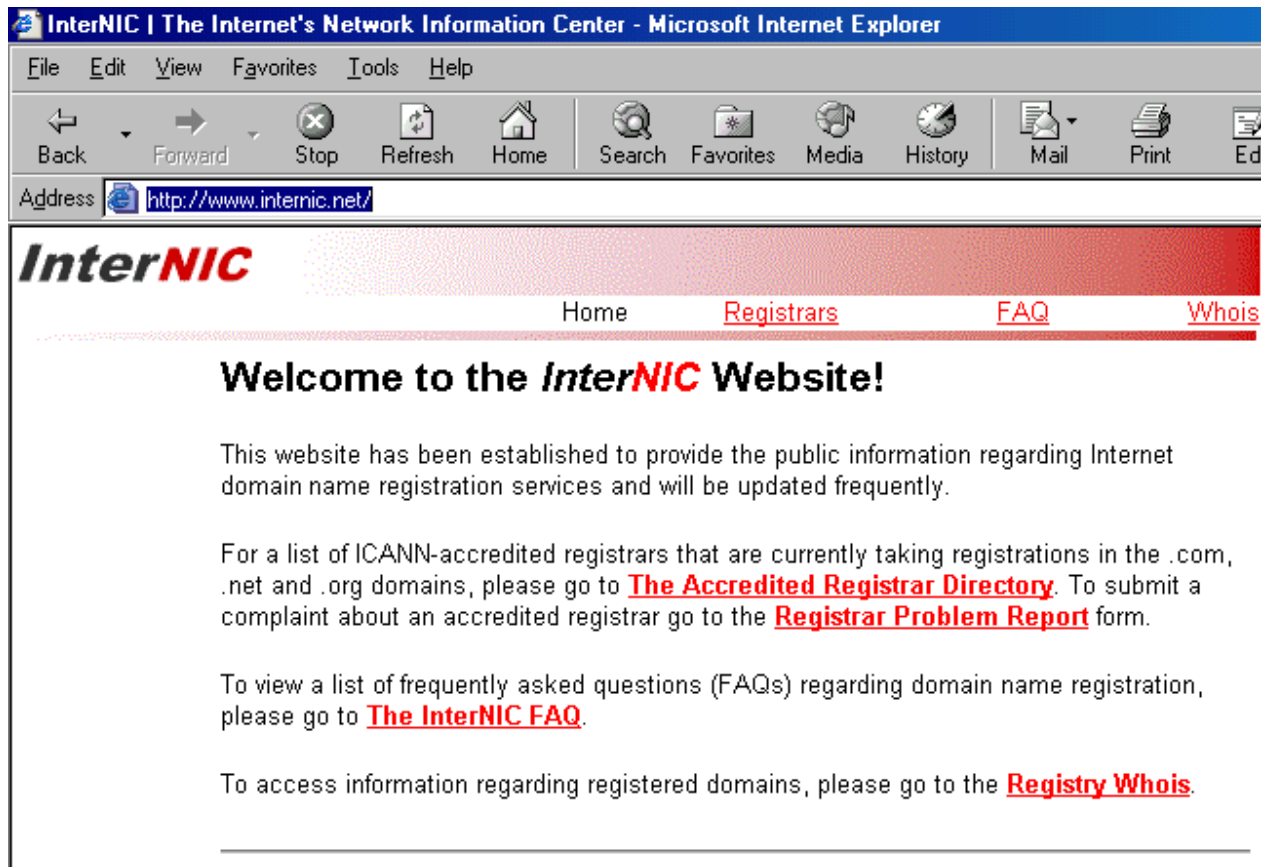
Getting Your Page “Noticed” (con’t)

- List keywords in META tag in order of importance; keywords should be in lower case; set the language if necessary
- Use both plural and singular forms if appropriate
- Quickly describe your site in the text, and use some “keywords” (do not use keywords more than five times on a page)
- Put your keywords in the TITLE tag also
- Also use the NAME=“DESCRIPTION” capability to give a concise description of your web site (under 200 characters)

Getting Your Page “Noticed” (con’t)

- Register your domain name with NIC (<http://www.internic.net/>)
- Pick a great domain name; available names can be found via NIC search; some names taken now are for sale (see online brokers)
- Title your pages descriptively
- Limit pages searched, if necessary, by using a robots.txt file in your web directory
- Use a hosting service, such as **GoDaddy**, which will also register your domain name

www.internic.net



www.godaddy.com

The screenshot shows the Godaddy website interface. At the top, there's a navigation bar with the Godaddy logo and a search bar for domain names. Below this is a green navigation bar with links to Products, Domain Names, Websites, Hosting, Web Tools, Support, My Account, and Cart. The main content area features a large banner for creating a professional website, highlighting a 50% discount and a starting price of \$1.00/month. Below the banner are three smaller promotional tiles: 'New names are coming' (for .MBA, .EAT, .FAN domains), 'Secure online payments' (for SSL certificates), and 'Improve search rankings' (for Google, Yahoo!, and Bing listings). The bottom of the page shows a Windows taskbar with various application icons and a system clock indicating 2:21 PM on 10/24/2013.

Domain Names | The World... x

GoDaddy
It's go time

Find a domain Go

Create an account or Log In

24/7 Support: (480) 505-8877
Hablamos Español

Products Domain Names Websites Hosting Web Tools Support My Account Cart 0

Start your domain search Go Bulk Search **
\$12.99/yr

Create your own professional website

- Easy-to-use tools & hundreds of designs
- Includes free domain and hosting
- No technical skills required

Get Started Starting at \$1.00/mo

50% OFF

New names are coming
From .APP to .WEB see which domains will be available soon.

Find your perfect name

Secure online payments
Keep customer information safe with an SSL Certificate.

Starting at \$69.99/yr

Improve search rankings
Get your business listed higher on Google®, Yahoo!®, and Bing®.

Starting at \$2.99/mo

2:21 PM
10/24/2013

Getting Your Page “Noticed” (con’t)

- Register directly with search engines that provide this capability and adhere to their catalog requirements (ie Yahoo & Inktomi)
- Also follow the “How to include Your Site” or “Add URL” link found on many search engines
- There are submission services that will do this for you (www.submit-it.com, www.netcreations.com/postmaster, www.register-it.com) and other sites that lists all registration forms (www.freelinks.com)
- Some charge a fee, and some are free (selfpromotion.com, www.addme.com)

Getting Your Page “Noticed” (con’t)

- Make sure each of your pages has a link back to your home page
- Add your URL to your letterhead, business cards, and all company materials
- Include your URL in all your e-mail correspondence (i.e. with a sig file)
- Send out a press release to traditional media
- Cross link to other sites, and get customers and vendors to link to you

12 KEY ON-PAGE SEO FACTORS THAT SEARCH ENGINES (AND USERS) LOVE

ON PAGE SEO GUIDE: THE BEST TIPS FROM THE PROS

1 LEVERAGE SEO-FRIENDLY PERMALINK URLS

You want your page's URL to be short and keyword rich. Avoid ugly URLs, like backlinko.com/p=123. Google has stated that the first 3-5 words in a URL are given the most weight.

Permalink: <http://backlinko.com/perfect-on-page-seo>

Edit

Add media:   



3 DAZZLE WITH MULTIMEDIA

Text can only take your content so far. Engaging images, videos, and diagrams can reduce bounce rate and increase time on site: two user interaction-based Google ranking factors.

5 DROP KEYWORD IN FIRST 100 WORDS

Your keyword should appear in the first 100-150 words of the article. Putting the keyword early in the content emphasizes that your article's content is about that keyword.

7 NAIL LOADING SPEED

Google is on the record as saying page loading speed is an important SEO ranking signal. You can boost your site speed by using a CDN, compressing images, and switching to faster hosting. Make sure it loads in 4 seconds or less for your page to load: a study by MunchWeb that 75% of users wouldn't re-visit a site that took longer than 4 seconds to load.

9 USE SOCIAL SHARING BUTTONS

Social signals are becoming a larger part of search engine algorithms. A study by BrightEdge found that prominent social sharing buttons can increase social sharing by up to 700%.

11 SLASH BOUNCE RATE

A bounce — when a visitor quickly leaves your site — is used by search engines to gauge a page's quality. Google can use the Google toolbar, Chrome browser, and Google Analytics data to determine a page's bounce rate. To lower your bounce rate add internal links, write compelling copy, and invest in a clean site design.

2 GOAL=START TITLE WITH KEYWORD

Your title tag is the most important on-page SEO element. An SEOmoz study found that pages that started their title with a keyword ranked higher than those with the keyword in the middle or end of the title.

4 USE OUTBOUND LINKS

Outbound links to related pages is a relevancy signal that helps Google zero-in on your page's topic. An industry study by SEOwizz.net found that adding outbound links to authority sites — like Harvard.edu and Wikipedia.org — boosts a page's rank in Google.

6 WRAP YOUR TITLE IN AN H1 TAG

The H1 tag is your "headline tag". Most blog platforms (like Wordpress) automatically add the H1 tag to your post title. But some themes override this setting. Check your site's code to make sure your title gets the H1 love it deserves.

8 ADD MODIFIERS TO YOUR TITLE

Adding modifiers like "2013", "best", "guide", and "review" can help you rank for long tail versions of your target keyword.

10 POST LONG CONTENT

The SEO adage "length is strength" was supported by a SERPiq.com industry study which found that longer content ranked significantly higher in Google. Aim for at least 1500 words when targeting competitive keywords.

12 SPRINKLE LSI KEYWORDS

LSI keywords are synonyms that Google uses to determine a page's relevancy (and possibly quality). Find LSI keywords using "Searches Related to..." at the bottom of Google search results or by entering your keyword into the Google Adwords Keyword Tool.

Attention Grabbing Techniques for Visitors

- Give something away “free”
- Automate e-mail with a “mailto” tag
- Use a guestbook to solicit contact info
- Embellish forms with drop downs and buttons
- Use multimedia effectively (ie HTML5 multimedia, RealAudio, Shockwave, Flash, or Java), discussed later in course

Search Engine Optimization (SEO)



References



New York Times Bestselling Author
DONALD MILLER

BUILDING A STORY BRAND



*Clarify Your
Message so
Customers
Will Listen*

Head First HTML and CSS: A Learner's Guide to Creating Standards-Based Web Pages

by Elisabeth Robson and Eric Freeman | Sep 11, 2012

★★★★☆ ~ 251

Paperback

\$26⁶² to rent

\$28.99 to buy

✓prime FREE Delivery Wed, Mar 25

More Buying Choices

\$12.82 (79 used & new offers)

Building a StoryBrand: Clarify Your Message So Customers Will Listen

by Donald Miller | Oct 10, 2017

★★★★☆ ~ 1,187

Hardcover

\$17²⁹ ~~\$24.99~~

✓prime FREE Delivery Wed, Mar 25

More Buying Choices

\$10.99 (59 used & new offers)

Web Copy That Sells: The Revolutionary Formula for Creating Killer Copy That Grabs Their Attention and Compels Them to Buy

by Maria Veloso | Feb 15, 2013

★★★★☆ ~ 118

Paperback

\$20⁰⁷ ~~\$21.95~~

✓prime FREE Delivery Thu, Mar 26

Only 20 left in stock (more on the way).

More Buying Choices

\$7.33 (39 used & new offers)

"Veloso has cracked the code for writing irresistible copy that gets people to buy without you begging on the web."
—Annette Kunkin, author of *Ultimate Persuasion*

WEB COPY THAT SELLS

third edition

The Revolutionary Formula
for CREATING KILLER COPY
That Grabs Their Attention
and COMPELS THEM TO BUY

maria veloso

Project 1



- Design and build your own personal home page with multiple sections
- Requirements:
 - Multiple Sections with internal links to them
 - Your photo
 - At least one list
 - Table
 - External links (text or graphic)
 - Colors
 - Form (**with echo**)
 - Image map and/or inline frame
- You can use either your version with or without CSS layouts
- Pure HTML, no special plug-ins, viewers, or web page builder apps in this assignment

Example Student Project 1

This home page contains information about my [address](#), my [interests](#), my [graphic design tools](#), and my [favorite NBA teams & players](#).



Photo 





List 

Established 1995

My Interests:

- Graphic Design
- Music Production
- Poetry
- Basketball
- Netflix & Chilling



Image Links 

Mr. Address:

2421 Raymond Road
Memphis, TN 38115

Graphic Design Tools:

Favorite NBA Team	Favorite Player(s)
Memphis Grizzlies Golden State Warriors Chicago Bulls	- Michael Conley, Jr. - Steph Curry - Michael Jordan

Table 

For more info, [CONTACT](#) [@mccall02@osu.edu](#)

Please enter the following information, I would love to know who visited my home page:

Name: City: State:

[CLICK](#) on the WUC to visit WUC's Homepage

[CLICK HERE](#) for Jansheng's Temperature Conversion

[CLICK HERE](#) to input a sentence & have it broken down into a list of words

[CLICK HERE](#) to see pictures of my favorite things to eat

[CLICK HERE](#) to view my HTML drawing

[CLICK HERE](#) to listen to some music! iTunes.com

[CLICK HERE](#) to view my graphic design web page! stuL1GationDesigns stuL1G0nDesigns



Form

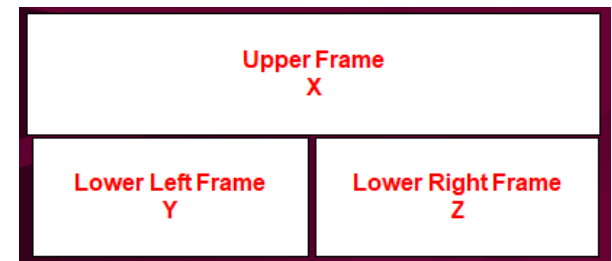
Links to Homework 



Inline Frame 

Homework

- Textbook Chapter 7
- Project 1 Completion
 - Including posting to student server
 - Email URL to instructor
- Appendix – SSL (https)
- Appendix - frames



Web Security

- We pass a tremendous amount of sensitive information back and forth via the internet, VPN's (Virtual Private Networks), LAN's (Local Area Networks), etc. every day
- What is to stop hackers and thieves from tapping in to that sensitive information and using it to their own gain?
- **Most important is SSL (Secured Socket Layer)**
- In these days of information there are a great number of reasons web designers and developers use secured pages:
 - Protect user identities and passwords
 - Protect credit card transactions and other sensitive information during online consumer purchases
 - Allows users to safely view personal and business financial information
 - Secure the transmissions of other sensitive personal information like your social security number
 - Keep secret sensitive corporate information being passed between branches, divisions, etc.
- Granted, there are many more reasons you may come across a need for security but the short list above gives you some of the highlights
- Without the ability for encryption many of the online resources we take for granted today would not be possible

SSL Process

- First: A client (that's the browser) requests a SSL connection with the server
- Second: The server sends a Certificate (more on that later)
- Third: The client validates the Certificate, creates a session key and encrypts the key, and sends it back to the server
- Fourth: The server decrypts the session key and establishes the encrypted connection
- However, you have yet to send any real information
- All you have right now is an established connection
- The Certificate that the server sent out is what makes this whole process work
- A Certificate is obtained from a Certificate Authority, which is sort of like a notary public that verifies the Certificates authenticity
- The Certificate contains the common name of the server, making it impossible to use on other servers
- It also uses keys, a public and private key, to **create and verify a secured connection**

SSL Setup

- The first is to do is to set up SSL on your own server
- Most of you are probably not running your own web server so we'll cover just the highlights
- In order to set it up you would need a server running Apache or IIS (Internet Information Services) and be familiar enough with servers and network configurations to set it up
- Next you need a Certificate, which you can get from a certificate authority like VeriSign (or you can even create a Certificate yourself)
- Once you have the pieces put together it's really as simple as creating a folder in your web where all of your secured pages are stored and routing all references to your secured pages through your secured server
- Or you could use the secured server that your web host has already set up

SSL (con't)

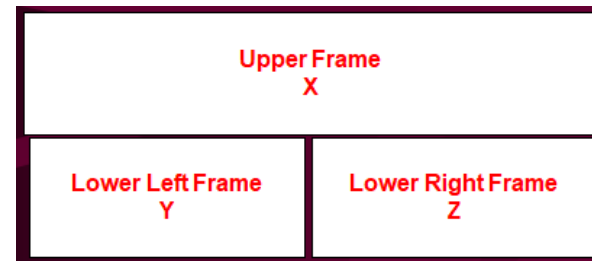
- Most web hosts already have a secured server set up and ready to use
- These are referred to as Shared Secure Servers or Shared SSL since the server takes care of security for several different webs
- If your site is hosted by a provider, odds are you have this service available to you at no additional charge
- If your site is being hosted for free (like on a personal web server with your ISP) you will probably not have a secured server available to you and if you do you can probably expect to pay for it
- As a general rule, the free hosting services don't come with the bells and whistles like Shared SSL
- Simply contact your provider and see if you have the service available to you
 - They will create a special folder within your web for your secured pages
- They should also provide you with a URL reference that routes through their secured server to your secured folder; the URL should look something like this:
<https://www.WebHostSecure.com/YourWeb/YourFolder/YourPage.html>
- Be sure that the URL reads https; that "s" on the end is what designates the link as secured

SSL (con't)

- Why wouldn't you just want to secure every page that you create as a matter of practice?
- Well, there's one simple and very good reason → Speed
- When you use the encryption capabilities every bit of information sent back and forth is encrypted and decrypted
- That includes graphics, photographs, text, data, etc. and it can create quite a drag on both upload and download time
- If you encrypted everything, your visitors would probably become very impatient with your site and surf off somewhere else
- A good rule of thumb is to only use encryption when absolutely necessary and consolidate your encryption needs in to as few pages as possible
- This should give you some good working knowledge of SSL and encryption; for most of you it will be as simple as emailing your host and having SSL set up for your web

Frames - Deprecated

- Your browser window can be divided into separate “frames” each of which can show separate HTML documents
- You can have links in one HTML document to control which HTML document shows in another frame (or even control which HTML document shows in a separate browser window)



Frames (con't)

- Browser windows can be split up either horizontally or vertically
- Frames can be nested, so that a window divided can be further divided
- HTML documents are either:
 - “frameset” documents
 - content or “body” documents
- For nested frames there can be one or several frameset documents

Frames (con't)

- Two vertical frames:
 - `<html>`
 - `<head><title>...</title></head>`
 - `<frameset cols="*,*">`
 - `<frame src="x.html" name="left">`
 - `<frame src="y.html" name="right">`
 - `</frameset>`
 - `</html>`
- You can give the frames names like we have done here with "left" and "right"; then they can be referenced in other frames

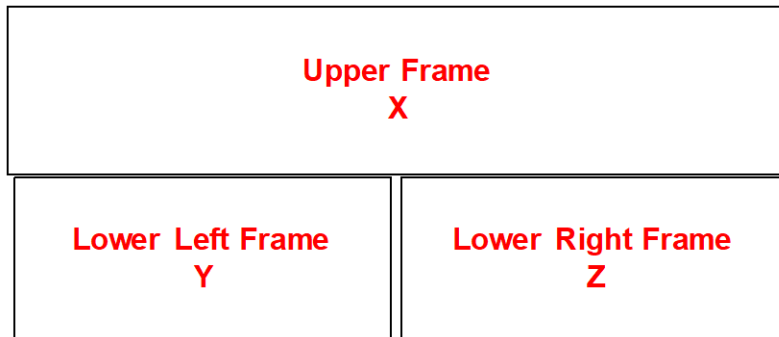
Frames (con't)

- Two horizontal frames:
 - `<html>`
 - `<head><title>...</title></head>`
 - `<frameset rows="*,*">`
 - `<frame src="x.html" name="top">`
 - `<frame src="y.html" name="bottom">`
 - `</frameset>`
 - `</html>`

Sizing Frames

- Dividing into rows:
 - `<frameset rows="s1,s2,s3...."`
- Dividing into columns:
 - `<frameset cols="s1,s2,s3...."`
- The sizes (s1...sn) can be:
 - * (divide up evenly)
 - absolute number - pixels (ie 300)
 - percent (ie 30%)
 - relative number (ie 2*, means twice as big as others)

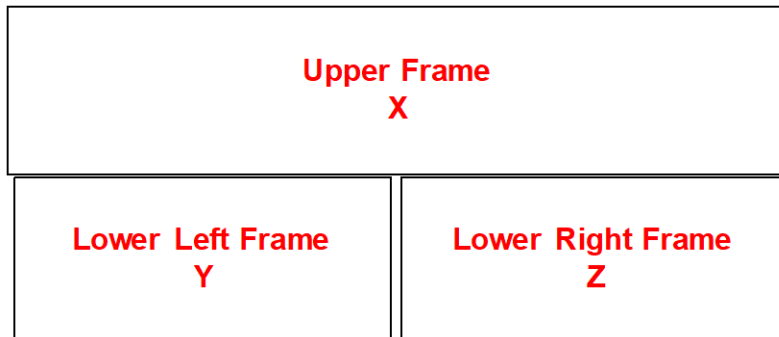
Nesting Frames - Directly



- `<frameset rows="*,*">`
- `<frame src="x.htm">`
- `<frameset cols="*,*">`
- `<frame src="y.htm">`
- `<frame src="z.htm">`
- `</frameset>`
- `</frameset>`

Nesting Frames - Indirectly

[can give frameset as name]



- `<frameset rows="*,*">`
- `<frame src="x.htm">`
- `<frame src="yz.htm"`
`name = "bottom">`
- `</frameset>`
- *Document yz.htm:*
- `<frameset cols="*,*">`
- `<frame src="y.htm">`
- `<frame src="z.htm">`

Targeting Frames

- Use “target” keyword in “a href” tag:
 - `<a href=mydoc.html target=“bottom”>...</a>`
- Targets may be:
 - HTML documents named in a “name” within the “frame” tag
 - `_self` (this document)
 - `_parent` (parent document)
 - `_top` (update entire window)
 - `_blank` (open new browser window)
- Use of any other “name” not specified in a “frame” tag will open a new browser window, and give that window a “name” !

Other Arguments

- <FRAMESET FRAMEBORDER="yes|no" ...>
- <FRAMESET BORDERCOLOR="\$\$\$\$\$\$" ...>
- <FRAME FRAMEBORDER="yes|no" ...>
- <FRAME BORDERCOLOR="\$\$\$\$\$\$" ...>
- <FRAME NORESIZE ...>
- <FRAME SCROLLING="yes|no|auto" ...>
- <FRAME MARGINWIDTH=\$\$...>
- <FRAME MARGINHEIGHT=\$\$...>