# ITMD-361
# CLASS 4
# SEPTEMBER 12, 2017

# TONIGHT'S AGENDA

- **Lab 2**
- **Questions so far on how HTML works?**
- **HTML Forms**

# HTML5 SKELETON PAGE STRUCTURE

```
<!DOCTYPE html>
<html lang=“en”>

    <head>
        <meta charset="utf-8">
        <title>Title Here</title>

    </head>
    <body>

    </body>
</html>
```

ITMD-361

# HTML FORMS

# HTML FORMS

- **Forms collect & send data back to a server or like processer**
- **Forms CANNOT be nested inside other forms**
- **Forms collect data with form controls**
- **Two Parts To All Forms:**
  1. Client side form element & form controls in the HTML
  2. Server side resource the form will transmit the data to

- **Form controls can exist outside a form but will not transmit data without some type of JavaScript to read and submit the data. So don’t do it.**

# HTML FORMS

- **Form element contains all form controls**
  - <form></form>
- **Has two primary attributes you need**
  - action & method
  - <form action="URLToScript" method="post"></form>
  - Example: <form action=“getform.php” method=“get”>
- **Action attribute**
  - The URI of a program that processes the form information.
- **Method attribute**
  - The HTTP method that the browser uses to submit the form.
- **HTML Forms Guide-MDN**

# HTML FORM METHOD

**The HTTP method that the browser uses to submit the form. Possible values are:**

- **POST**
  - Corresponds to the HTTP POST method
  - Form data is included in the body of the HTTP Request and sent to the server.
  - Most often used
  - No character or size limit to data
- **GET**
  - Default if no method is specified
  - Corresponds to the HTTP GET method
  - Form data is appended to the action attribute URI with a '?' as separator, and the resulting URI is sent to the server.
    - http://server.com/script.php?name=Bob&city=chicago
  - Only use this method if the form has no side-effects and contains only ASCII characters.
  - URL length is limited and varies by browser & server

**Neither method does any encryption of the data. That is the responsibility of TLS (SSL) during the connection.**

# HTML FORM CONTROLS

# FORM CONTROLS

- **Form controls are used to collect input data from the user and submit them to some resource that processes that data.**
- **In basic HTML forms, the form controls need to be nested inside the <form></form> tags.**
- **Form controls can be used outside a <form> tag in an HTML page but there is no way for it to submit data without using JavaScript to read and submit the data.**
- **HTML5 offers additional capabilities for control placement**
  - See form attribute in docs of a form control
  - Control can only be associated with one form on a page
- **All form controls need a name attribute or the data will not be submitted.**
- **Form controls need a type attribute to set the type of control (there are a couple exceptions).**

# FORM CONTROLS ATTRIBUTES

**Form controls have numerous global attributes that change behavior.**

- List of Global Form Attributes-MDN

**There are two attributes that are very important**

- **name attribute**
  - The name of the control; gets submitted with the form data. Used as a variable in the server side resource that processes the script.
  - All form elements must have a name attribute except the submit and/or reset buttons.
  - The name attribute specifies the variable name for the form control in the HTTP Request.
  - If the processing script is already created, the form name attributes must match the variables that the server side script expects to use.
- **type attribute**
  - Only used on <input> form controls
  - The type of control to display. The default type is text, if this attribute is not specified.

# FORM CONTROLS

- Full list of HTML From controls-MDN
  - Form Controls are <type> attributes.
  - Each changes the type of control displayed.

# FORM CONTROLS FOR TEXT ENTRY

- **Single line text entry field**
  - <input type="text" name="variable" >
  - Default type
  - Must have name attribute
  - value attribute can set default value. Better to use HTML5 placeholder attribute
  - maxlength attribute specifies the maximum number of characters (in Unicode code points) that the user can enter
- **Multiline text entry field**
  - <textarea name="variable" ></textarea>
  - multi-line plain-text editing control
  - Must have name attribute
  - Text inside the tag will display in the area by default and will send to the server...be careful. Better to use HTML5 placeholder attribute
  - cols and rows attributes can set the size. Better to size in CSS.

# FORM CONTROLS FOR SPECIAL TEXT ENTRY

- **Password entry field**
  - <input type="password" name="variable" >
  - Behaves the same as single line text control
  - Text characters are obscured from view with asterisk or bullet character
  - **IMPORTANT**: Data is not obscured in the HTTP Request

# FORM CONTROLS SUBMIT AND RESET

- **Submit button control**
  - <input type="submit">
  - Button needs to be nested in <form></form> tags to submit that form to it’s action via it’s method
- **Reset button control**
  - <input type="reset">
  - Resets all form controls to default state
- **value attribute on either button sets the visible text on the button**
  - <input type="submit" value="Send Me">
  - Button would say **Send Me** on it

# FORM CONTROLS RADIO BUTTONS

- **Radio Buttons should be used when you only want someone to choose one value from a small list of values.**
- **<input type="radio" name="variable" value="thevalue">**
- **No visual text associated with the button. Need to add text next to it. Best to use a label form control (next slide).**
- **name attribute sets the group of buttons & variable name**
- **value attribute sets the value of the variable if that option is selected that is transmitted to the server**
- **Only one button per group can be selected**
- **Use the checked attribute to preset a selection.**
  - Boolean attribute. Doesn’t need a value, but if you provide a value just provide checked. ( checked="checked" )

# FORM CONTROLS LABELS

- **Provide a textual label for your form control**
- **Better than using just plain text next to your control because it associates the label with the control. Good for accessibility.**
- **Gives semantic meaning to the controls**
- **Implicit or explicit association**
  - Implicit, label wraps form control
    - <label>Full Name:<input type="text" name="variable"></label>
  - Explicit, associates label with form control’s **id (not name)**
    - <label for="name-form">Name:</label>
    - <input type="text" name="variable" id="name-form">
- **Allows clicking of labels to make selections for radio buttons and checkboxs.**

# FORM CONTROLS CHECKBOXES

- **Checkboxes should be used when you want someone to choose zero or more values from a small list of values.**
- **<input type=”checkbox" name="variable" value="thevalue">**
- **No visual text associated with the checkbox. Need to add text next to it. Best to use a label form control.**
- **Must have name attribute, it sets the variable name.**
- **value attribute sets the value of the variable if that box is selected that is transmitted to the server**
- **If form processing is PHP it is common to set the name attribute's value in array syntax (square backets) if there is more than one with the same name. They will come over as an array of values.**
  - name="var[]"
- **Use the checked attribute to preset a selection.**
  - Boolean attribute. Doesn’t need a value, but if you provide a value just provide checked. ( checked="checked" )
- **Only transmits the variable and value if it is checked**

# FORM CONTROLS FOR MENUS

**There are two main types of menus, drop-down and scrolling.**

- **Uses the <select name="variable"></select> tags to define**
- **Select tag contains <option></option> tags for each option**
- **Option tags can have a value attribute**
  - Sends the value attribute instead of the contents of the tag to server
- **Option tags can have a selected attribute (boolean attribute)**
  - <option selected="selected"></option> or <option selected></option>
- **Option tags can be grouped with <optgroup label="cat"></optgroup>**
- **Select tags size attribute controls size of menu**
  - Size of 1 is a standard pull down menu
  - Size > 1 displays that many options, see p164
- **multiple attribute allows multiple selections (boolean attribute)**
  - <select multiple="multiple"></select> or <select multiple></select>

# FORM CONTROLS
# FILE UPLOADS

- **<input type="file" name="variable" >**
- **If a form includes a file control it must:**
  - Be POST method
  - Must include enctype="multipart/form-data" attribute
- **Server resource must be setup to handle a file upload. Handled a little differently depending on the language.**

# FORM CONTROLS HIDDEN FIELD

- **<input type="hidden" name="variable" value="thevalue">**
- **You can use a hidden field control when you need to send data to the form processing application that does not come from the user.**
- **It is not visible in the page but is visible if you view the page source.**
- **Some pre-built script files may require you send information to them via a hidden control.**
- **Example:**
  - <input type="hidden" name="success-link" value="http://www.example.com/link.html">
  - This would send the URL value to the server using the success-link variable.

# FORM CONTROLS FOR ACCESSIBILITY

- **Labels**
  - We already looked at labels
  - You should always use them to give a text label to your form controls
- **Fieldset and legend**
  - The fieldset element indicates a logical group of form controls. A fieldset may also include a legend element that provides a caption for the enclosed fields.
  - Allows you to group form controls

```
<fieldset>
<legend>Field title</legend>
</fieldset>
```