Csci 4131 Internet Programming

Fall 2018
Lecture 3, September 12

Instructor: Dr. Dan Challou

Logistics

- HW 1 Due Friday at 2PM – 3 page web site: Class Schedule, Form, and "Widget Page" – HTML and basic CSS
- Last Time:
  - How Computers Represent Text Recap
  - Brief Look at HTML
  - Hardware / SW Models of WWW function
    - URI's, URL's
    - Intro to HTTP / HTTPS protocols
- Reading:
  - Chapters 2 and 3 in Sebesta
  - HTML and CSS tutorials at

    http://www.w3schools.com/

Office Hours will be posted today on Moodle

- Dr. Dan's Office hours are after class today from 3:45 – 5:00pm. I have to add another hour, which will be sometime this Friday after 2pm this week.
- I will probably get to my office about 4:10pm, since I answer questions after class here in beautiful Anderson 370

Today

- Intro to HTTP wrap-up
- More HTML and Some CSS
  - Lists
  - Tables
  - Some CSS
  - Navigation
  - Forms

But first:
Review Exercise 1 from last class

- Create an HTML 5 Webpage that:

1. Displays your name on the TAB opened by the browser
2. Has a header that says: My Favorite Site
3. Has a link that says: click here, and when you do, it opens up your favorite site

HTTP Wrap- up

HTTP Get and Post Requests

- The two most common HTTP request types (also known as request methods) are **`get`** and **`post`**.
- A `get` request typically gets (or retrieves) information from a server, such as an HTML document, an image or search results based on a user-submitted search term.
- A `post` request typically posts (or sends) data to a server.
- Common uses of `post` requests are to send form data or documents to a server.
- An HTTP request often posts data to a server-side form handler that processes the data.
- `Get` requests and `post` requests can both be used to send data to a web server, but each request type sends the information differently.

HTTP: Get

- A get request appends data to the URL, e.g., `www.bing.com/search?q=challou.`
- In this case, ***search*** is the name of the routine on the server side, ***q*** is the name of a variable in bing's search form and ***challou*** is the search term.
- The ? in the preceding URL separates the query string from the rest of the URL in a request.
- A *name/value* pair is passed to the server with the *name* and the *value* separated by an equals sign (=).
- If more than one *name*/*value* pair is submitted, each pair can be separated by an ampersand (&).
- ***E.g., www.bing.com/search?q=challou&q=csci4131 …***
- ***Or a + sign***
- ***E.g, www.bing.com/search?q=challou+csci4131 …***

- The server uses data passed in a query string to retrieve an appropriate resource from the server.
- The server then sends a response to the client.
- A get request may be initiated by submitting an HTML form whose method attribute is set to "get", or by typing the URL (possibly containing a query string) directly into the browser's address bar.

From last class: Activity 1– you can use your computing device (and work with your neighbors)

- In bing (https://bing.com) – type: **golden gophers** in the search bar
- Can you specify the routine is called on the server side?
- What are the name/value pairs passed to the routine called on the server side?
- What are the names and what are the values?

HTTP: Post

- A post request sends form data as part of the HTTP message, not as part of the URL.
- A get request typically limits the query string (i.e., everything to the right of the ?) to a specific number of characters, so it's often necessary to send large amounts of information using the post method.
- The post method is also sometimes preferred because it hides the submitted data from the user by embedding it in an HTTP message.
- If a form submits several hidden input values along with user-submitted data, the post method might generate a URL like www.searchengine.com/search.
- The form data still reaches the server and is processed in a similar fashion to a get request, but the user does not see the exact information sent.

HTTPS on the World Wide Web

- URLs of websites that handle private information, such as credit card numbers, often begin with https://, the abbreviation for Hypertext Transfer Protocol Secure (HTTPS).
- HTTPS is the standard for transferring encrypted data on the web.
- It combines HTTP with the Secure Sockets Layer (SSL) and the more recent Transport Layer Security (TLS) cryptographic schemes for securing communications and identification information over the web.

Back to HTML

- Recall the purpose of HTML?
  - Structure and Content

Lists

-   - ordered (numeric lists)
-   - unordered (bullet lists)
- Each Element

  Example: ListEx.html

Exercise 1

1. Make an ordered list of your 2 favorite movies
2. Each list item from part 1 should contain an unordered nested list of your two favorite character associated each of the movies you chose for part 1 of this exercise

For example:

1. Lord of the Rings Trilogy:
    - Frodo
    - Arwen
2. The Godfather
    - Tom
    - Vito

Tables

- <table> </table>
- All elements below used after start tag for table
- <thead> </thead> - table heading
  - Use <th> </th> for a bold fond heading
- <tbody> </tbody>
  - <tr> </tr> - elements specify a row
  - <td> </td> - data element in a row (column)
- <tfoot> </tfoot>
  - Same as head if you want bold font in footer
- <caption> </caption> - caption for table

Simple Table Example

- L2_table_ex1.html

Exercise 2 – Create a valid Web Page That Displays the Following:

- Two row table , each row has 2 columns
- First Column in each row contains the name of your favorite Movie, second column contains your favorite Character in that movie

Let's Digress, Talk **Presentation**/Style

- CSS – cascading style sheets

Methods for Adding Style (CSS) to HTML 5 documents

- Inline – applied via the HTML style attribute to a particular element
- Embedded – use HTML style tag:

  <style> </style>in the <head> section of the HTML document. Define styles for particular elements, and classes that can be applied to elements in between the style start and end tag
- Separate File that is included in HTML (separates structure and content from presentation)

  Use the HTML <link> tag to include a style file

CSS Inline

- The style attribute
- Here is a paragraph
 with a line break

css_inline_ex.html

Example

- Using the inline style tag, make the **font-size** of the table we did in the previous exercise 16 pixels (i.e., **16px**) and make the font **color** red and collapse the border

1. <table border = "1" style="font-size:16px;color:red;border-collapse:collapse;">
2. <table style="font-size:16px;color:red;border-collapse:collapse; border: 1px solid black;">

Useful HTML attributes and CSS Style properties

- HTML table **border** attribute (boolean – set to 0 or 1, default is off) – it is depricated though!!!
- CSS border properties
  - http://www.w3schools.com/css/css_border.asp
- CSS border-collapse property
  - http://www.w3schools.com/cssref/pr_border-collapse.asp
- CSS **color** property
- CSS **background-color** property

Example

- Turn the HTML border attribute on for the table you developed in the previous exercise

Embedded and External CSS Files

Embedded CSS using HTML Style Tag

https://www.w3schools.com/html/html_css.asp

Source: http://www.w3schools.com/tags/tag_style.asp

External Including the CSS properties from an External StyleSheet

```
<head>
<link rel="stylesheet" type="text/css" href=“
        style.css" />
</head>
```

Make an External Style Sheet of the Embedded CSS we specified Earlier?

Don’t forget to validate your HTML AND CSS

- https://validator.w3.org/

- https://jigsaw.w3.org/css-validator/

Mutiple Page Websites

- Use the navigation element (i.e., <nav> tag)

```
<nav>
    <a href="homepage.html">Home</a> |
    <a href="Page1.html">Page 1</a>
    …
</nav>
```

homepage.html

Target Attribute for Navigation

- http://www.w3schools.com/tags/att_a_target.asp

Next Class

- Forms and Beyond – see class schedule for reading (Finish Chapter 2)
- Finish CSS – Chapter 3