Csci 4131 Internet Programming
Fall 2021
Lecture 3
September 15th

Instructor: Dr. Dan Challou

Logistics – Csci 4131 Lecture 3, 9/15

- HW Assignment 1 Due **THIS Friday 9/17** at 11:59pm
  - You can get the assignment write up in the assignment item:
    - **Programming Homework 1: Assignment Specification and Submission Link** item in the week 1 module on the class Canvas site

- The weekly readings, tutorials, homework due dates and exam dates are in the item:
  - **Course Schedule: Weekly Class Readings and Tutorials, Exam Dates, and Programming Assignment Due Dates**

Accessible via the Resources Module at the top of the Home Page on the class Canvas site

- ***Don't forget to complete your zyBook Lecture Prep and Homework Assignments listed in the Assignments tab in your zyBook!!!***

- ***I will extend zybooks lecuture preparation activities for lecture 2 (Monday 9/13, and Wednesday 9/15 through end of day Saturday 9/18). Extensions are not likely to happened again, so please get your lecture prep activities done before lecture meets).***

- Programming Homework Assignment 2 will be out tonight or tomorrow morning and will be due next Friday, 9/24 at 11:59pm

Grading Policy – added to Syllabus Monday

- **20%** Exam 1
- **20%** Exam 2
- **12%** Final Project
- **28%** Seven Individual HW Programming Projects
- **5%** Zybooks Class Preparation Assignments (Typically due before lecture starts - but subject to change)
- **10%** Zybooks Homework Assignments (Typically due on Saturdays at the end of the day - but subject to change)
- **5%**- Randomly Selected In-class exercises or quizzes

This week's readings and tutorials

Zybooks assignments (in zyBook),
www.w3schools.com:
CSS Tutorial, JavaScript Tutorial
https://www.w3schools.com/css/default.asp
http://www.w3schools.com/js/
Optional Textbook: Sebesta Chapters 3,4

Questions?

Agenda

- Last Time:
  - How Computers represent text revisited
  - Intro to HTML
- Today
  - HTML Revisited
  - Forms
  - HTTP
  - Start CSS

Review Exercise 1 from Lecture 2

- Use a text editor for example, NotePad , or Notepad++ (**Not** MS Word), or TextEdit (Mac) and 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

An Answer: Exercise1.html

Review Exercise 3 from Lecture 2

- Create a Two row table, that has two columns
  - The first column in each row should contain the name of one of your favorite movies,
  - The second column in each row should contain your favorite actress or actor in that movie.

If you don't have a computer, write your answer on a piece of paper, transfer it to electronic format using your favorite text editor, and upload your answer to Canvas

An Answer: L2_table_exercise_answer.html

A better looking table (with 3 rows): L2_table_exercise.html

Recall, some Other Useful HTML5 Tags – block and inline elements

- Div, Span, and Navigation tags
- Nav Useful for creating a multi-page Website
  (see homepage.html / Page1.html example)
  homepage.html
- Can alternately use div if no browser support
- Span is an inline element – useful for inline styling!
- See - http://www.w3schools.com/html/html_blocks.asp

Homework 1

- Demo
- ***And note, make sure to pick a non-copyrighted YouTube playlist for your HW, or it may not work!***
- There is a way to get around it, but we can't endorse using copyrighted YouTube videos

Questions?

HTML Forms

- HTML5 provides **forms** for collecting information from users
- HTML forms enable you to do some syntactic validation on the client side before sending the information collected from the user to the server side (your Zybook, and http://www.w3schools.com/ for more details)
- A form typically has a collection of input fields that can be submitted for further processing

Form Input Types (note, type is an HTML attribute of the input element)

- Several Input Types Available :
  - Text input (textarea is an HTML tag (element))
  - Submit input
  - Password input
  - Checkbox input
  - Radio Button input
  - Select Element input
  - Text Area input (element not a attribute of input)

HTML5 Form input types – (Introduced with HTML 5)

- input Type color
- input Type date
- input Type datetime
- input Type datetime-local
- input Type email
- input Type month
- input Type number
- input Type range
- input Type search
- input Type tel
- input Type time
- input Type url
- input Type week

Notes

- These types are not universally supported by all browsers (but the most popular browsers support them)
- HTML 5 input types are ***somewhat*** *self validating* on the client side.
- Benefits
  - eliminate the need to add (some) JavaScript code to your web pages to validate user input,
  - reduces the amount of invalid data submitted and thus reduces Internet traffic between the server and the client to correct invalid input.
- *The server should still validate most all user input.*
- When a user enters data into a form then submits the form the browser immediately checks the self-validating elements to ensure that the format (syntax) of the input data is correct

HTML Self Validating Input Types

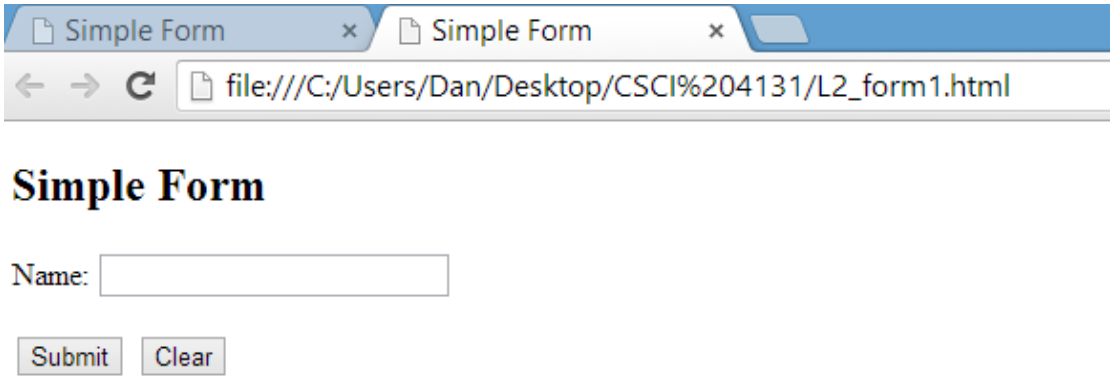
input type	Format
color	Hexadecimal code
date	yyyy-mm-dd
datetime	yyyy-mm-dd
datetime-local	yyyy-mm-ddThh:mm
month	yyyy-mm
number	Any numerical value
email	name@domain.com
url	http://www.domain-name.com
time	hh:mm
week	yyyy-Wnn

Fig. 3.5 | Self-validating input types.

Example – a simple Form (Code Along)

- Build a Simple Form
  - L2_form1.html

Exercise 1 – Add an email input type and URL input type to the Simple Form



Add an email input type, and a URL input type to the Simple Form

L2_exercise1.html

Here are the constructs to add to the form we did in the Example

```
<label> Email
        <input type = “email”  name=“emailAddress”
            placeholder = “name@domain.com” required />
            (name@domain.com)
</label>

<label> My Favorite URL
        <input type = “url”   name=“aUrl”
            placeholder = “http://www.domainname.com” required />
            (http://www.domainname.com)
</label>
```

Submit through the Lecture 3, Exercise 1 link in the week 2 Module on Canvas

Please Close your computer (almost) when done!

Let’s take a Look

- Dan’s Attempt (note, this will not be posted with the lecture materials until after the exercise submission due date.

L2_exercise1.html

Questions?

Hidden Fields

- L2_form1a.html

The World Wide Web, Revisited

- Recall - after the Internet was in place, the world wide web – which uses the Internet, emerged.
  - In 1989, Tim Berners-Lee of CERN (the European Organization for Nuclear Research) began to develop a technology for sharing information via hyperlinked text documents (aka web pages) called HyperText Markup Language (HTML).
  - Berners-Lee also wrote the Hypertext Transfer Protocol (HTTP)—a communications protocol used to send information over the web.
  - Each web page on the Internet is associated with a unique Uniform Resource Locator (URL).
  - The URL specifies the address (i.e., location) of the web page displayed in the browser window.
  - You get or put (post) information to a URL!!!!
  - URLs usually begin with **http://** (why? What else can they begin with???)
  - Web use increased dramatically with the availability in 1993 of the Mosaic browser, which featured a user-friendly graphical interface – its inventor, Marc Anderson, went on to found Netscape

Components of a URL

Consider the following URL:

- https://twin-cities.umn.edu/about-us
- **PROTOCOL: https://** indicates that the Secure HyperText Transfer Protocol (HTTPS) should be used to obtain the resource.
- Next in the URL is the server's fully qualified hostname (for example, twin-cities.umn.edu)—the name of the web-server computer on which the resource resides.
- This computer is referred to as the host, because it houses and maintains resources.
- The hostname twin-cities.umn.edu is translated into an IP (Internet Protocol) address—a numerical value that uniquely identifies the server on the Internet
- An Internet Domain Name System (DNS) server maintains a database of hostnames and their corresponding IP addresses and performs the translations automatically.

Components of a URL

- The remainder of the URL (homepage/about-UI/index.html) specifies the resource's location (/about-us) and name (default is index.html) on the web server.
- The location could represent an actual directory on the web server's file system. For *security* reasons, however, the location is typically a *virtual directory*.
- When the request is received by the web server, it translates the virtual directory into a real location on the server, thus hiding the resource's true location.

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 Request

- https://www.bing.com/search?q=challou&form=QBLH&sp=-1&pq=challou&sc=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&…***
- ***Or a + sign***
- ***E.g, www.bing.com/search?q=challou+ …***
- 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.

Next Time –

- Wrap up Intro to HTML, HTTP
- CSS
- DOM
- JavaScript?

  – Make sure to do the zybooks reading; homework, and lecture prep assignments in your Zybook:
  – And do tutorials (and optional reading) for Week 3, specified in the document:

  Course Schedule: Weekly Class Readings and Tutorials, Exam Dates, and Programming Assignment Due Dates

  in the Resources Module at the top of the Home Page on the class Canvas site