CS 134: Pre Midterm Study Guide

Chapter 1: Introduction to web development

- The client uses a ***web browser*** to access web sites and applications
- A ***web server*** is used to host web sites and applications
- The core content and structure of a web site is encoded in ***HTML (Hypertext Markup Language)***
- Styling (e.g. colors and fonts) and layout for modern web sites are encoded in ***CSS (Cascading Style Sheets)***
- The protocol most frequently used to upload or publish web sites is ***FTP (File Transfer Protocol)***
- A HTTP URL consists of three major parts: ***protocol***, ***domain***, ***path*** (e.g. http://www.mysite.com/my/path/)

Chapter 2: How to code, test, and validate a web page

- HTML consists of nesting, bracketing regions called ***elements***
- Within elements one can define **attributes** with **values** in the like so: ***<element attribute = 'value'>***
- Elements typically are of the form ** … ** for elements that take content and **
** for those that don't
- In HTML the ***<body>*** element is used to contain the visible ***content*** of the site.
- In HTML ***whitespace*** consists of ***spaces, tabs, and line returns*** and generally does not affect the output
- A CSS ruleset generally consists of a ***selector*** and a bracketed region {} of one or more ***rules***
- It is a good idea to ***validate*** both your HTML and CSS to ensure it is formatted correctly and abides standards
- The simplest HTML document contains a ***<html>*** element containing one ***<head>*** and one ***<body>*** element
- Each CSS rule consists of a ***property*** and a ***value*** of the format: ***property: value;***

Chapter 3: How to use HTML to structure a web page

- The ***<meta>*** element is used in the head to provide background data to search engines and the browser
- ***Block*** elements generally display on a new line, and take parameters such as margins and padding
    - Examples: *** , <main>, <nav>, <header>, <footer>, <h1>, ***
- ***Inline*** elements get rendered such that they flow along with text content
    - Examples: ***, <a>, , , <img>***
- Characters such as & and © are encoded in html as ***character entities*** of the format ***& ©***
- An element can be identified uniquely with an ***id*** attribute. There can only be one element with a given ***id***
- An element can also be classified with a ***class*** attribute. Multiple elements can have the same ***class***
- ***id*** and ***class*** are used for example by CSS or JavaScript to point to a particular element or set of elements
- New to HTML5 are ***semantic elements*** such as <main>, <nav>, <header>, <footer>, <section>, <aside>, etc.
- ***Semantic elements*** act the same as does, but provides more information about the content to search engines and developers, and makes HTML and CSS code easier to read.
- ***Absolute paths*** include the entire URL including domain
- ***Root-relative paths*** begin with a / character and are relative to the root of the web site
- ***Relative paths*** begin with either "..", a folder, or a file name and are relative to the current file
    - ***".."*** in a path navigates one level upward in the file tree
- ***Links*** are defined with the ***<a>*** element with the format: ***…***
- ***Lists*** can be ordered with numbers ***<ol>*** or unordered with bullets ***<ul>***
    - Both styles take list elements ***<li>*** nested within them

Chapter 4: How to use CSS to format the elements of a web page

- CSS rules can be defined in three ways:
  - An ***external stylesheet*** linked to in the <head> with a ***<link>*** element (this is the most common)
  - An ***internal stylesheet*** in the <head> in a ***<style>*** element
  - Directly applied ***inline*** to an element with a ***style attribute***
- CSS rules ***cascade*** and will add to and potentially override each other. The order of precedence is:
  - First rules in an ***external stylesheet*** apply
  - ***normalize.css*** is a commonly used generic stylesheet to ensure a uniform baseline across browsers
  - These can be overridden by an ***internal stylesheet***
  - Which can be overridden by an ***inline style attribute***
  - More ***specific rules override more general ones***.
  - For ***two competing rules with the same specificity, the later rule takes precedence***
  - Units in CSS can be ***absolute (px, pt)*** or ***relative (%, em)***, the former are always consistent and can aide in pixel-perfect or ***fixed-layout design***, the latter are more flexible to different environments and screen sizes are important for ***fluid-layout design*** or ***responsive design*** (e.g. for mobile)
  - One ***em*** is equal to the height of the current font-size
  - ***Colors*** can be defined in CSS with ***Red, Green, Blue*** values of the sort ***rgb(R%,G%,B%),*** or in hexadecimal, ***#RRGGBB*** or can be named if they are part of the list of web colors
  - ***CSS Selectors:***
    - ***Element selectors*** consist simply of the element name with no brackets: ***h1***
    - ***Id selectors*** consist of the ***#*** symbol followed by the id: ***#content***
    - ***Class selectors*** consist of a period followed by the class: ***.myClass***
    - ***Descendent*** consist of the ancestor followed by the descendent: ***header h1***
    - Multiple selectors can trigger the same ruleset if put together with commas: ***h1, h2, h3***
    - The universal selector ******* targets all elements, but gets overridden easily
    - ***Pseudo-class selector*** such as ***:hover*** and ***:visited*** trigger based on user behavior
    - These rules can be used together to make more specific selectors:
      - E.g. ***div#content a:hover*** would target links being hovered over that are descendents of a div element with the id "content"

Chapter 5: How to use the CSS box model for spacing, borders, and backgrounds

- CSS block elements render using the ***CSS box model***
  - From outside to in, the box model consists of ***margin, border, padding, content***
- It's often a good idea to use a ***reset selector : * {margin 0; padding 0;}*** to ensure a consistent baseline
- Borders can be defined together with the format: ***border: width style color;***
- Margins, borders, and padding can be defined with individual parameter for example with a ***"-left"*** suffix, or they can be defined in one rule with the following patterns:
  - ***margin: 5px;*** defines all four margins together
  - ***margin: 5px 4px;*** defines the top and bottom margins together followed by the right and left
  - ***margin: 5px 4px 3px;*** defines the top margin, then right and left together, then the bottom
  - ***margin 5px 4px 3px 2px;*** defines in clockwise order, top, right, bottom, and left margins
- ***auto*** when used as a size, will do its best to center an element within its parent
- Foreground colors are defined with the ***color*** property, background colors with ***background-color***

Chapter 6: How to use CSS for page layout

- ***Floating*** an element to the left or right takes it out of the regular flow of the document, and causes unfloated elements to flow around the element. This is very commonly done with block elements to generate layout.
- If you have an element you want to ensure has no element floating to its right, left, or on either side (e.g. a footer) you can set the ***clear*** property to ***right, left,*** or ***both*** accordingly
- Common page layouts consist of a header at the top, a footer at the bottom, and either two or three columns of content defined with float
- A fluid layout will define its elements' widths with percentages and ems rather than pixels

Chapter 7: How to work with links and lists

- To link to a place within a long website, you can use "**#id**" in the href attribute of a link to go to the element with the id specified. If nothing precedes the #, this will link to somewhere on the current page. This can also be used with external pages as well e.g. ***"newpage.html#myID"***
- To force a link to open in a new tab or window, you can specify the following attribute: ***target='_blank'***
- It's possible to link to media formats other than HTML. When doing so, it's possible to set a ***MIME*** attribute to tell the browser what sort of media you are linking to.
- It's also possible to link using a protocol other than HTTP, for instance when sending an e-mail with the ***mailto:*** protocol
- Navigation bars are very frequently built using re-styled unordered lists.
    - Some rules of thumb to do so include:
        - Turning off bullets with ***list-style-type: none;***
        - Changing the display type of link in the list to block format with ***display: block;***
        - Floating list items to the left so the stack horizontally: ***float: left;***
        - The current and final menu items will often be given classes to allow them to be styled differently than the other menu items
    - It's also possible to make creative use of the ***:hover*** pseudo-class selector and nested lists create two and three tier navigation menus.

Chapter 8: How to use Responsive Web Design

- When designing with responsive web design and mobile support in mind, it's important to make use of a ***fluid layout*** wherever possible. This may mean converting absolute sizes to relative ones.
- It is possible to use the ***"min-"*** and ***"max-"*** prefixes on sizes to define absolute limits to relatively defined widths and heights.
- Modern browsers include a ***developer toolbar*** that make testing responsive layouts on different screen sizes and devices much easier.
- It's important to set a ***meta viewport*** element with your desired viewport width (e.g. to device-width) as well as the desired initial scaling to ensure consistent behavior across devices.
- It's also a good idea to make use of ***@media*** queries e.g. ***@media only screen and (max-width: 479px){…}*** to redefine your stylesheet and layout to better display on larger or smaller screens.
- When used, it's best to either set a default stylesheet for large screens, and then define successively smaller ones using @media queries; or alternatively set a default stylesheet for mobile devices, then define successively larger ones up to a desktop device.