1. **Know the Tags/Attributes**

* <html>
* <title>
* <header>
* <body>
* h1, h2, h3, h4, h5 and h6
* p
* hr
* strong
* em
* img
* a
* br
* div
* <table>
* <tr>
* <td>
* <th>
* <ul>
* <ol>
* <li>
* <map>
* <area>
* <form>
* <input>
* <textarea>

NOTES: Do not get the HTML <em> tag and the em unit of measure confused.

1. **HTML Concepts**
   1. Know how to use HTML tags
   2. What is the basic HTML Page Structure?
   3. What is a  <!DOCTYPE html> declaration
   4. Adding HTML and/or CSS comments
   5. Know the three main components of a basic HTML page are
      * Tags
      * Attributes
      * Values
   6. Know how to setup and list using the appropriate tags
   7. Know how to map an image and what tags are used when mapping.
   8. Know and understand the tag attributes
      * href
      * shape
      * cords
   9. What image attributes can be use.
   10. Understand Absolute & Relative File Path
2. **Know the concepts and tags associated with a form**
3. **CSS Concepts**
4. Know the three key concepts required to understand the basics of CSS structure and know how to use them. We will learn about **selectors**, **properties,**and **values**.
5. There are three ways of inserting a style sheet.

Inline

Internal (embed)

External

1. Review the CSS Selectors (i.e. element selector, class selector, id selector, descendant selector, multiple selectors, etc.)
2. Know how to use the tags and attributes with CSS
3. Review Box Model & Positioning Concepts (i.e. positioning float, clear, z-index, & more)
4. Review Navigation Menu Concepts
5. **Review the Layout concepts**
6. Related to Layout Structure Tags (i.e. <nav>, <header>, <footer>, <aside>, <section>, <main>)
7. Layout Modes

* Block
* Inline, for text
* Table
* Positioned

1. Review Flexbox concepts
2. **Review Unit of Measure & Overflow Property**
3. **Review Responsive Design**
4. **Keywords**

**HTML CSS Fundamentals**

**A element:** Used to define a link to another page. Also known as the anchor element.

**Absolute URL:** Looks exactly the same, whether the reference is on your server or another server.

**Alt attribute:** Displays text if image does not load.

**Attributes:** Contains information about content without being content itself.

**Block-level:** HTML elements that display on their own line.

**DOCTYPE:** Identifies the version of HTML in page's content.

**In-line:** HTML elements that render in the same line as other content.

**Markup:** HTML elements that describe text content and make references work.

**Elements:** Describe different parts of a webpage.

**Em element:** Means “stress emphasis.” Dictates semantic, not visual, emphasis.

**Heading element:** Provides six heading levels to structure page's outline.

**Img element:** Primary choice for displaying an image.

**P element:** Used to mark up a paragraph.

**Relative URL:** URLs are relative to the file’s location on the server. Locates file by referring to the location of the file that contains the URL reference.

**Semantic:** Describes content’s meaning, not its display.

**Semantic HTML:** Refers to content that is marked up with the HTML elements that best describe it and without regard for how the content should look.

**URL:** Uniform Resource Locator (URL) is a technical name for a web address. It contains information about a ﬁle’s location and instructions for the browser.

**Values:** Numerical values paired with attributes.

**Text editor:** Used to create and edit HTML code and content for webpages.

**Web browser:** Identifies and displays webpages and their content.

**Notepad:** Free Windows text editor.

**.htm:** A webpage file name extension.

**.html:** Recommended webpage file name extension.

**File name extension:** Identifies file type for the web browser.

**Default page:** Top-level page in a webpage directory, typically: index.html.

**Directory:** Organizes webpages and assets into folders and subfolders.

**Sub-folders:** Separate folder in directory for a major section of site, along with individual sub-folders within those, as necessary.

**View Source:** A browser command that allows you to view the webpage's code. Sometimes called Page Source command.

**Adobe Photoshop:** Industry standard image editor for both print and web images.

**GIF:** Lossless web image format that is best for icons and logos because it compresses areas of continuous colors or repetitive patterns better than JPEG.

**JPEG:** Lossy web image format that is best for color photographs.

**PNG:** Lossless web image format that is best for icons and logos. Like GIF, it is better at compressing areas of continuous colors or repetitive patterns than JPEG. It also has better compression algorithm for smaller file sizes and superior transparency support (alpha transparency).

**Anchor:** Anchors define the specific location in a webpage where a link leads when triggered.

**Building Blocks - CSS**

**CSS (Cascading Style Sheets):** Defines the content’s appearance.Controls basic formatting, layout, print controls, and dynamic properties.

**.css:** Cascading Style Sheetfile name extension.

**Style sheet:** A text file containing rules for displaying elements in a webpage.

**Declaration block:** Describes the formatting to be executed.

**Comments:** Organizational or descriptive notes added to style rules in CSS.

**Cascade:** When more than one style rule applies to an element, CSS uses this principle to determine which of the conflicting rules wins.

**External style sheet:** Preferred method of applying CSS to HTML page. It is a separate page that holds all CSS styles to be loaded into one or more pages.

**Embedded style sheet:** Second method of applying CSS to HTML page. Sets the styles directly in HTML document. Typically goes in the page’s head. Because styles are only in HTML file, the CSS will not apply to other pages and lacks same caching benefits.

**Inline style:** Least desirable method of applying CSS to HTML because it mixes content (HTML) and presentation (CSS). Each inline style instance only affects one element, which even simple changes are time consuming to make. Inline styles are not for regular use but can be employed for a quick test before placing a style in an external style sheet.

**Selector:** The part of a CSS style rule that determines which elements will receive formatting.

**Class selector:** Preferred selecting method because it can be reused and applied to many elements.

**id selector:** Less preferred selecting method. Associated styles cannot be reused on other elements because id can only appear on one element in a page.

**Parent:** Selector for the element that directly contains the element (the child) you wish to format.

**Descendant:** The selector for the element you wish to format.

**Layout with Styles**

**Absolute positioning:** Takes elements out of the normal flow entirely by specifying the precise position with respect to the body or to the nearest positioned ancestor element.

**Block-level:** Starts on a new line like a new paragraph.

**Border:** Outside edge of the padding.

**Box model:** Invisible box encloses each element made up of content area, padding, border, and margin.

**Clear property:** Can prevent the floating effect.

**Fixed-layout:** Layout has pixel-based widths for whole page and for each column of content. Width does not change on smaller devices or browsers.

**Float property:** Makes elements float in a sea of text (or other elements).

**Floated element:** Forces subsequent content to wrap around it.

**In-line:** Does not generate a new line.

**Inline-block:** Hybrid display that allows an element to appear inline with other content, while otherwise behaving like a block-level element.

**Margin:** Invisible space around the border that separates one element from the next.

**Padding:** Space surrounding the content area.

**Relative positioning:** Means moving the element with respect to its original location. Surrounding elements are not affected at all.

**Forms**

**Form tag:** Includes the URL of the script that will process the form and its method (post or get).

**Form elements:** Actual form elements where visitor enters information. Includes fields, checkboxes, drop-down menus, and radio buttons.

**Submit button:** A trigger that activates the script that collects the data from the form and sends the collected data to the server.

**Form method=“post”:** Correct form method for saving, adding, and deleting data in a database.Method sends more data to server, and the information in the form is not shown in the URL.

**Form method=“get”:** Data will show in browser’s address bar, which means the user can bookmark results. Form method used by most search engines in search forms because it allows a search query to be saved or sent to a friend.

**PHP:** Open-source scripting language written specifically for making Web pages interactive. Stands for Hypertext Preprocessor.

**Radio buttons:** A set of buttons of which only one can be selected at a time.

**Select boxes:** Offers visitors a choice from a given set of options. Made up of two HTML tags: select and option.

**Checkboxes:** A set of select boxes where a visitor can select as many options in a set as they like.

**Text areas:** Give visitor more room to write than text boxes. May be as large as page and will expand as needed. Perfect for eliciting questions and comments.

**Hidden fields:** Store data in form without showing it to visitor. Think of them as invisible text boxes.

**Video, Audio, and Other Multimedia**

**Audio element:** Adds audio controls to your webpage.

**Autoplay attribute:** Makes an audio file start playing when the page loads.

**Controls attribute:** Informs the browser to add a set of default controls to the video.

**Loop attribute:** Use with autoplay to set video to play automatically and continuously until stopped.

**Source element:** Allows you to define more than one source for a media element.

**Src:** Specifies URL to the audio or video file.

**Publishing Your Pages on the Web**

**Domain name:** Name associated with site.

**FTP client:** Used to upload pages to web host’s server.

**Web host:** Serve the site to anyone who visits the domain in a browser.

1. **HTML5 Structural Elements**

|  |  |
| --- | --- |
| **Code** | **Diagram** |
| Header Element  <header></header> Contains the headings  Nav Element  <nav></nav> Contains the main navigation  Main Element  <main></main> Contains the main content  Footer Element  <footer></footer> Contains the footer | **A wireframe contains elements from top to bottom as follows: header; nav; main, containing three div elements; footer.** |

Note: this information is not a complete list of concepts which you are expected to know. Make sure to review the information in Canvas in the various Modules.