Kevin Chen

3/13/16

HTML Notes (abbreviated)

* DOCTYPE declaration helps the browser display the page correctly.
  + Example: <!DOCTYPE html> Typical for HTML 5
* Attributes: come in name/value pairs like so: [name]=“value”
  + lang (language) attribute in the html tag ("en-US" for American English)
  + title attribute in paragraph elements (displays text when mouse rolled over)
  + disabled attribute specifies that an input element should be disabled
  + id attribute specifies a unique id for that element
  + Recommended to use lowercase attributes and quote attribute values
  + Double quotes is most common, but single quotes can be used.
* Horizontal rule: <hr /> tag used to separate content (adds a horizontal line)
* Paragraph element: <p>. Can’t change output by adding extra spaces or lines in code: the browser will remove them.
  + A line break separates between paragraphs in actual output.
* Break element: <br /> puts a line break without starting a new paragraph.
* Preformatted text element: <pre>. Preserves spaces and line breaks in the source code. Displayed in a fixed-width font (usually Courier).
* Style attribute: Can be used in a variety of tags. Syntax: style="property:value;" Property is a CSS property. Value is a CSS value. List of different properties:
  + Background color: <body style="background-color:lightgrey;">
  + Text color: <h1 style="color:blue;"> (or put it in a paragraph element)
  + Fonts: <h1 style="font-family:verdana;"> (or put it in a paragraph element)
  + Font-size <h1 style="font-size:300%;">
  + Text Alignment:<h1 style="text-align:center;">
* Formatting text: <b> for bold text and <strong> for strong importance, <italic> for italics and <em> for emphasized text, <small> for small text, <marked> for highlighted text, <del> for deleted text (adds a line through the text), <inserted> for added text (adds an underline), <sub> for subscript, <sup> for superscript
* Quotations: <q> for short quotes, <blockquote> for quoted sections (indents whole paragraph), <abbr> for abbreviated text, <address> to display contact information (italicized), <cite> for titles of works (italicized), <bdo> defines bi-directional override (overrides current text direction).
  + Example of <bdo>: <bdo dir="rtl"> (Text written from right to left)
* Computercode elements: Doesn’t preserve spaces/line breaks from code. Can surround computercode elements with <pre> to fix this.
  + <code> shows piece of computer code.
  + <kbd> for keyboard input. (Use | in between steps)
  + <samp> defines sample output from a computer program
  + <var> for variables in both mathematical expressions and in programming context
* Comments: <!--Comment -->. Some cases comments may be executed!
  + Conditional comments: <!--[if IE 8]> …some HTML here… <![endif]-->
  + HTML comments tags can also be generated by various HTML software programs.
* Colors: Various colors can be accessed through different means
  + By name. Example: <h2 style="background-color:red">
  + By RGB. Example: <h2 style="background-color:rgb(0,255,0)">
  + Hexadecimal colors. Example: <h2 style="background-color:#FF0000">
* Using CSS: Can be added to HTML in 3 ways:
  + Inline Styling: Used for defining style for a single element. Use <style> attribute. Example: <h1 style="color:blue;">
  + Internal Styling: Used to define style for a single HTML page. In the <head> section, use the <style> element.
  + External Styling: Can be used for defining the style for many pages. In the <head> section, add <link rel="stylesheet" href="[relative URL]">
* Use id attribute (for single element) and class attribute (for groups of elements) to target a specific subset of elements.
  + Can be referenced by CSS using #[id/class name] followed by brackets with CSS code inside.
* Links:
  + Local links: links to the same website. Don’t include <http://www>.
  + Target attribute: specifies where to open the linked document. Example: target=“\_blank” opens link in a new browser window or in a new tab. If your webpage is locked in a frame, use target=“top” to break out of frame.
  + Bookmarks: used to jump to sections of a page. Use id attribute, and then use href=“#[id name]”
* Images: empty tag, self-closing, contains only attributes.
  + Attributes: src (source file), alt (alternative text used for screen readers), width, height
  + width and height can be specified in style attribute (this is recommended). Example: style="width:128px;height:128px;"
  + gif standard (example: src="programming.gif") allows animated images
  + Can be used as links when placed within <a> element
  + Float attribute: image flows to the left or right of a text (Ex. float=“right”)
  + Map: Use <map> tag to define an image-map, which is an image with clickable areas. In <img> tag, add the attribute usemap=“#[name]” and then in <map> tag, add the attribute name="name"
* Tables: Defined using <table> tag, divided into rows with <tr> tag, rows are divided into table data with <td> tag. A table row can also be divided into table headings with the <th> tag instead of <td> tag.
  + Add border in CSS (recommended). Can be done in HTML with border attribute. Example: border=“1”
  + Use colspan attribute in <td> or <th> to span more than one column. Example: colspan="2". Similar for rowspan attribute to span more than one row.
  + Captions: use <caption> tag after <table> opening tag to add a caption (a title) to a table.
* Lists: <ul> for unordered list, <ol> for ordered list, <li> for each list item.
  + Description lists: defined by <dl>. <dt> for each term (name), and <dd> for each definition of the term (this is indented)
  + Horizontal lists: add display:inline as a style attribute for each list item.
* Blocks: Block-level elements (e.g. <div>, <h1>-<h6>, <p>, <form>): start on new line and take up full width. In-line elements (e.g. <span>, <a>, <img>): do not start on new line and take up as much width as necessary.
  + <div>: a container for other HTML elements
  + <span> a container for some text
* Layout: can be done using <div> (using CSS for positioning), HTML 5 (has various predefined locations for typical page elements, such as <header>), and tables (not recommended)
* Responsive Web Design: can manipulate size/location of content to make it look good on all screen sizes. Can be done by yourself or by using W3.CSS.
* Iframe: display a webpage within a webpage. Use <iframe> tag along with src attribute
  + Can set width/height, can remove border via style=“border:none”
  + Can be used as a target frame for a link. <iframe> tag must have a name attribute, and the target attribute in the link tag needs to be that same name.
* Use <script> tag to define a script (JavaScript). Either contains scripting code or points to an external script file through the src attribute.
  + <noscript> tag to provide alternate content is JavaScript is not supported or disabled.
* HTML Head: container for metadata (data about data). <html>, <body>, and <head> can be omitted in HTML5, though not recommended.
  + <link> element defines the page relationship to an external source (often used to link to style sheets)
  + <meta>: specify page description, keywords, author, and other metadata.
    - <meta name="keywords" content="[keywords separated by commas]"> to define keywords for search engines
    - <meta name="description" content="Free Web tutorials on HTML and CSS"> to add description of webpage.
    - <meta name="author" content="Hege Refsnes"> to add author
    - <meta http-equiv="refresh" content="30"> to refresh page every 30 sec.
  + <base>: specifies base URL and page target for all relative URLs on the page.
* Character Entities: Reserved characters in HTML must be replaced with character entities. Characters, not present on your keyboard, can also be replaced by entities.
  + Character entity syntax: &[entity name]; or &#[entity number];
  + For entity number, type number normally for decimal or place an “x” before the number for hexadecimal.
  + Non-breaking space (&nbsp; or &#160;) is a space that won’t break the word before and after it. It will make sure they are on the same line.
* Diacritical marks: glyphs (e.g. accents) added to a letter. Example: é.
  + Construct syntax: [letter]&#[mark number – consult a table];
* Charset. In HTML5, specify charset using the charset attribute in the <meta> element.
  + Limited charsets: ASCII, ANSI, ISO-8859-1
  + Default charset: UTF-8
* Uniform Resource Locator (URL) Encoding.
  + Syntax: [scheme]://[prefix].[domain]:[port]/[path]/[filename]
    - scheme defines type of Internet service (commonly http or https)
    - prefix defines domain prefix (often http)
    - domain - defines the Internet domain name (w3schools.com)
    - port - defines the port number at the host (default for http is 80) – this (and the colon in front) is often omitted.
    - path - defines a path at the server (If omitted: the root directory of the site)
    - filename - defines the name of a document or resource
  + Common URL schemes: http (common web pages, not encrypted), https (secure web pages, encrypted), ftp (for downloading/uploading files), file (file on a computer)
  + URLs can only have ASCII characters; otherwise, they have to be converted by replacing them with a “%” followed by hexadecimal digits.
    - Cannot have spaces. Spaces replaced by “+” or “%20”
* XHTML: HTML written as XML
  + XML is a markup language where documents must be marked up correctly.
  + Follow strict guidelines of what is more lax in HTML.
    - Must have DOCTYPE, xmlns attribute in <html> as so: xmlns="http://www.w3.org/1999/xhtml", <head>, <title>, etc.
    - Must be properly nested, must be closed, elements & attribute names must be in lower case.
    - Attribute values must be quoted, attribute minimization is forbidden

Notes Source: <http://www.w3schools.com/html/>