**HTML Essential Training, Part 1 - Eric Le**

**Questions and Responses**

***Intro to HTML***

**1) According to the author, what are the core languages a web designer needs to learn?**

HTML, CSS, JS

**2) [True or False] If we open a child element, we must close it before we close the parent element.**

True

**3) [Critical Thinking] Since there are now 2 different HTML specifications now being implemented at the same time, what is the authors advice regarding these standards?**

Just focus on learning HTML 5 syntax to make clean and well structured code. Don’t worry about the advanced features being added until you become more comfortable.

**4) What two websites does the author believe are the most useful to us as web designers and developers?**

These two sites are the most useful for web designers because they are geared more towards web designers/developers rather than browser developers, which is who the W3C and WHATWG specification documents are created for.

<http://www.webplatform.org>

<https://developer.mozilla.org/en-US/docs/Web/HTML/Reference>

**5) The author identifies four things that a code editor should do, at minimum. Name two of them.**

Line numbering, syntax highlighting

***Basic Page Structure***

**6) [Critical Thinking] What advantages does a tool like Chrome’s WebKit inspector have for us as developers?**

It allows us to take a look at the inner mechanisms that create web pages. We can visually examine how a web page is constructed and what makes it function. Useful for learning to recreate web pages or elements that we are interested in.

**7) In the author’s view, how is an HTML tag like a sandwich?**

Sandwich ingredients are encapsulated within slices of bread, like opening and closing tags. Each element is like a component of a sandwich. Header element is like condiments that improve a sandwich but do not define it. Body element is like the meat of a sandwich that contains all of the actual visual content required for a sandwich to exist.

**8) [Critical Thinking] What is the purpose of using a doctype declaration?**

Identifying what version of HTML is being used for a browser or user agent to be able to properly parse it.

**9) Why does the author advocate using the lang attribute inside the <html> tag?**

It is useful for screen readers and online translators to improve accessibility and accuracy of documents.

**10) What is the purpose of setting the charset attribute in the meta tag to utf-8?**

Some browsers may automatically set the charset for pages, but not reliably. Developers should set this so that web pages can reliably render the proper character set to reflect whatever language is being used in the document.

**11) Why does the author advocate using the name=”description” meta tag?**

Despite not being used by search engines when searching for content, it is used when search engine results are displayed to a user. It can also be useful for searching through your own sites for a specific web page.

**12) [Critical Thinking] What is the purpose of the <body> tag?**

The body tag contains all of the visual elements of a page, such as images, text, and video.

**13) [Critical Thinking] What is the purpose of the <p> (paragraph) tag?**

The <p> tag tells a browser or user agent to style the text contained in the element accordingly (as a standard paragraph). Used in conjunction with a CSS file, this can be used to quickly determine the style of a web page’s text.

**14) [Critical Thinking] What is the difference between the block level and inline level content models?**

Block level models render content in their own unique sections whereas inline level models concatenate content.

**15) Of the 7 new content models, identify which one is most like inline, and give 2 examples of elements it includes.**

Phrasing: <b>bolded text</><em> and italicized text</em> would be rendered as if they were in the same line.

Interactive: <a>anchored links</a> and button elements can exist as inline content

***Formatting Page Content***

**16) [Critical Thinking] What is the purpose of the <pre> tag?**

It tells a browser to display text as it exists within the confines of a tag. Useful for display snippets of code, or any other text where a specific formatting is necessary.

**17) [Critical Thinking] Why use an <em> tag instead of an <i> tag when they produce the same visual appearance?**

Users will comprehend the text the same way visually, but machines will understand them differently. Using a logical tag instead of a presentation tag will change the way a screen reader can comprehend text.

**18) [Critical Thinking] Although the heading tags have default have sizes related to them, what is the purpose of using more than one type of heading tag in our documents?**

Headings are useful for structuring document content logically, as in the case of a table of contents. Because their styles are malleable, it is important not to choose a heading based on how it is visually rendered.

**19) [Critical Thinking] Each block level element has margins that optionally create space around them. What does the author mean when he says, “vertical margins in CSS collapse?”**

If two vertically adjacent elements share a margin, they will overlap rather than rendering both margins, which would result in a double-wide space. If element1 has a 20px margin below it and element2, which is inferior to element1, has a 20px margin above it, they would render with a 20px space between them rather than 40px.

**20) [Critical Thinking] What does it mean when we say a tag is self closing?**

It means that it doesn’t require a separate closing tag. It contains no visual content, so rather than having two separate tags, they are combined into one <br />

***Video Notes***

[Welcome (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/welcome) (1:03)

[The importance of HTML (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/the-importance-of-html) (3:19)

* 3 core languages: Hyper Text Markup Language ( HTML), Cascading Style Sheets (CSS), JavaScript (JS)
  + HTML: Webpage structure
    - Contains all of a website’s essential content
  + CSS: Webpage style
  + JavaScript: Webpage behavior
* Markup Language
  + *Marks up* content to explain what content is and how it relates to other content on page
  + HTML is a very basic language, making it easy to adopt and learn

[Basic HTML syntax (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/basic-html-syntax) (8:42)

* HTML is basically just a TXT file with .htm or .html extension
* **Element** = Tags + Content
  + **Tags** identify page content
    - <p> = opening tag
      * Content follows this tag
    - </p> = closing tag
    - Not all tags have both opening *and* closing tags
    - Tags can be **nested** within each other
      * Creates parent-child relationships between tags
      * Certain tags can’t nest within others
      * Child element must close before parent element
  + **Attributes**
    - Tags can have attributes that are element-specific or global
    - <p lang=”en”>
    - <tag attribute=”value”>
    - Two types: Informative or Functional
      * **href** attribute is functional; tells browser where to go when link is clicked
  + If a browser sees text outside of an element, it’ll default to paragraph text

[The current state of HTML (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/the-current-state-of-html) (5:54)

* HTML grows quickly along with the development of the web
* History
  + Understand how language developed to see where it’s headed
  + Lots of parallel development in early days of HTML
  + HTML 4 was almost universally adopted by browsers in 1995
  + W3C released XHTML 1.0 as move from HTML to XML in 2000
  + 2009 W3C drops charter from XHTML 2.0
  + WHATWG forms to continue work on HTML 2001
    - Wanted to continue HTML 4.0 development
    - Eventually adopted by W3C as HTML 5.0
  + Two specifications of HTML concurrently developed by W3C and WHATWG 2019
    - W3C - Working to clear milestones to release HTML 5.1
    - WHATWG - Working on living standard with no version numbers
    - Versions beginning to diverge
  + HTML 5 vs. HTML 4
    - Focuses on building apps, drag’n’drop, location detection, drawing services
    - More of application development platform than just markup language
    - Big 3 varieties to use: HTML 4.0, XHTML 1.0, HTML 5.0
      * Most sites online use one of these three
        + Not many differences between pages using these three
      * Recommendation for beginners: Learn HTML 5 syntax, then add more HTML 5 features as skill grows

[HTML Resources (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/html-resources) (5:28)

* HTML Resources to learn from:
  + Specifications
    - [W3C Specifications HTML](http://www.w3.org/TR/html5/)
    - [WHATWG Specifications HTML](https://html.spec.whatwg.org/multipage/)
      * Gives overviews of how language works and functions
      * Written more for browser developers than web page authors
      * Gives detailed info on tags and elements
  + Learning resources/Knowledge bases
    - [www.webplatform.org](http://www.webplatform.org)
      * Open source wiki on variety of info regarding web development
    - [Mozilla HTML Reference](https://developer.mozilla.org/en-US/docs/Web/HTML/Reference)

[Choose a code editor (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/choosing-a-code-editor) (5:42)

* Creating a good website usually requires a variety of language, i.e. the big 3
* Good HTML editor can help catch syntax errors
* Miniumum requirements for code editor according to author:
  + Line numbers
  + Code formatting options
  + Syntax highlighting
  + Code support for necessary languages

***Basic Page Structure***

[Exploring an HTML document (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/exploring-an-html-document) (5:27)

* Most browsers have tools to inspect web documents
  + IE - Developer Tools
  + Firefox - Firebug
  + Webkit Browsers/Chrome - Inspector
* Open an inspector to examine the page’s HTML
  + Visual elements get highlighted when you hover the elements in inspectors
    - Inspecting them can tell you more information about the elements too
    - Allows you to see how specific web pages can be recreated
* Basic HTML structure:
  + DOCTYPE element
    - Tells browser what language is being used
  + Head element
    - Like condiments on sandwich
    - Improves a document but doesn’t define it
    - Can contain JS or CSS file references
    - Non-visual elements that help it function
  + Body element
    - Like the meat of a sandwich
    - All visual elements of a webpage

[DOCTYPE Declarations (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/doctype-declarations) (4:00)

* [W3 Doctype Declarations](http://www.w3.org/QA/2002/04/valid-dtd-list.html)
  + May be outdated (from 2002), but has information on what doctype declaration is and its history
  + History may be useful when examining older pages
* Function
  + Tells browser/**user agent** which version of HTML is being used
    - User agents are softwares that retrieveand relay information on behalf of a user
    - For knowing which version is used for parsing
  + Triggers “quirks mode”
    - Ensures page renders properly
  + **Doctype declaration is necessary**
* Not case sensitive “<!DOCTYPE HTML>” or “<!doctype html>”

[The document head (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/the-document-head) (6:11)

* Immediately after a doctype declaration is the **<head>** element
* Encapsulated within opening/closing html tags
  + Attribute **lang**=”en” can be added to html
  + Nice to do, but not necessary
    - Useful for screen readers and online translators
    - Good practice for accessibility
* Meta tag can be encapsulated within Header
  + Doesn’t require closing tag
  + Attribute **charset**=”utf-8” can be added to meta tag
    - Should be set for every html doc
    - Ensures that characters are properly rendered
    - Browsers may automatically set charset attribute, but not reliably
  + Attribute **name**=”description” **content**=”Test description” can be added to meta tag
    - Historically used by search engines to determine what type of content is on a page
      * Brief description of web page’s content
      * Good practice to have for search engines and personal indexing
        + Search engines will use this content when displaying search engine results
        + Personal indexing purposes such as searching your own site or content management system
* Title element
  + Title tag will give web page a title for search engines
  + Good practice to have
    - Without one, search engines will display site as “Untitled”

[The document body (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/the-document-body) (3:58)

* Where all of page’s visual content exists
  + Can be considered sibling of head tag
  + Exists on same level
* Indentations do not matter
  + Opening and closing tags can also exist on the same line
  + Make these decisions based on legibility for developers
* Without CSS, style is defaulted to browsers’ default rendering
* <h1> tag
  + Top-level heading
  + Can be used to encapsulate text within body element
* <b> tag
  + Bolds text

[Understanding content models (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/understanding-content-models) (6:39)

* Content Model
  + Almost every element in HTML belongs to at least one content model
  + Define type of content to expect within element and controls syntax rules such as element nesting
  + Only two content models existed before HTML 5
    - Block level
      * Take up own line within flow of doc
        + Headings, separate paragraphs
    - Inline
      * Appears within flow of other content
        + Bolded text, links, etc.
  + <a> element
    - Anchor element
    - Used for linking outside pages
  + HTML 5 Content Models:
    - [W3C Types of Content Models](https://html.spec.whatwg.org/multipage/dom.html#kinds-of-content)
    - Seven in total
      * Block level and inline level models still exist, but not as their own models
        + Some of these act as block level or inline level
      * Many of these models overlap with one another
    - Flow
      * Contains majority of elements
      * No specific rules on how to display or render this content
    - Metadata
      * Sets up presentation or behavior of rest of content
      * Primarily found in head of document
        + Meta tags, script tags, style tags
    - Embedded
      * Any content that imports other resources into an object
    - Interactive
      * Any content specifically intended for user interaction
      * Anything interactive like anchored links, buttons, audio\*, video\*, object\*
        + \* = Only interactive under certain circumstances, like having controls for a video player
    - Heading
      * Headings
    - Phrasing
      * Text of document & any elements used to markup text within paragraph-level structures
      * Very similar to inline elements from HTML 4
    - Sectioning
      * Defines scope of headings/footers
      * Creates new sections in document

[Formatting content with HTML (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/formatting-content-with-html) (6:58)

* HTML uses tags to markup content on a page
* If content is not marked up:
  + Some browsers/user agents may format it as default text
  + Others might not understand what it is at all and not render it properly
* <pre> tag
  + Preformatted text
  + Tells browser to display text as it exists within tags
  + Used for displaying code snippets
    - Many browsers default to Monospaced font
* <h1> tag
  + Heading
  + Goes from h1 (biggest) to h6 (smallest)
    - Don’t use different headings *just because* they vary in size
    - **Bad practice**
  + Headings can be styled and personalized
    - If not, they use browser defaults
* Presentation tags
  + Bold <b> and italics <i> can be used to mark up text
  + HTML 4 moved away from these to have all styling handled by CSS
  + Reintroduced in HTML 5 because they are still useful for specific cases
* Logical tags
  + Emphasis <em> and italicized <i> looks the same to users
    - Screen readers will change vocalization based on <em> or <i>
    - Emphasis will make readers add inflexion while <i> will not
  + Strong <strong> and bold <b> produce visually similar results, but change readers’ comprehension

[Using headings (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/using-headings) (7:50)

* Used to structure pages and determine content hierarchy
* Should be used based on *level* of heading and not size
  + All about structure
    - Used to determine level of importance
    - Imagine page as if it were an outline or table of contents
      * Select heading level based this hierarchy
  + Style of headings can be changed with CSS
* Important to develop a strategy for how headings will be used
  + Also important for SEO
    - No harm in using h1 multiple times in a page; just depends on whether they are used logically
    - Try not to skip heading levels i.e. going from h1 to h5

[Formatting paragraphs (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/formatting-paragraphs) (5:35)

* HTML must be backwards compatible
  + Supports loose syntax rules
  + Not having formatted content may otherwise break webpages
    - Allows some browsers to automatically format unformatted content
* All HTML elements have top and bottom margins
  + CSS element margens collapse
    - Prevents double-spaced paragraphs
* Avoid using empty paragraphs to simulate new lines or blank spaces
  + CSS can handle visual formatting and spacing easily
* Paragraph closing tags are optional according to specs
  + Good practice to just close them anyway
  + No reason to leave out closing tags

[Controlling line breaks (Links to an external site.)](https://www.linkedin.com/learning/html-essential-training-2017/controlling-line-breaks) (3:46)

* Line breaks
  + May be necessary to use when content should exist within the same paragraph but on separate lines
  + <br> tag creates new line aka hard return
    - Visually renders without extra spacing as in the case of a paragraph
    - Doesn’t require closing tag because it holds no content
  + Self-closing tag
    - Because XML required all tags to be closed, self-closing tags were created
    - <br /> is an example of a self-closing tag
    - Not needed in HTML 5 anymore, but also no harm in using this