

Module 1 notes

Mateo López

Evolution of Websites:

The expansion of the World Wide Web has led to diverse applications for websites, spanning business, social media, news, entertainment, search, and more.

HTML (HyperText Markup Language):

HTML provides the structural framework for websites, employing tags enclosed in angle brackets (< and >). Browsers interpret HTML to determine the format, style, and structure of webpages. HTML files typically have a .html extension.

Elements and Readability:

HTML tags define elements, enhancing readability and establishing visual hierarchy. Different types of websites utilize varying balances of these elements.

Evolution of HTML:

With the advent of CSS, HTML could concentrate solely on structure while CSS handled styling. Deprecated HTML attributes were phased out with the advancement of CSS. New iterations of HTML introduce features and streamline existing ones.

CSS (Cascading Style Sheets):

CSS focuses on presentation, separating it from HTML's content focus. It offers more formatting options than HTML styling attributes, enhancing efficiency and flexibility in document formatting.

JavaScript:

JavaScript complements HTML and CSS, enabling interactive features such as menus, slideshows, and animations. It manages events and user inputs, processes data, and dynamically modifies website content.

-
- Syntax in spoken and code languages involves rules for proper communication and interpretation.

- HTML tags serve as instructions for web browsers, defining the structure and content of web pages.
 - Tag attributes provide additional information about elements, modifying their behavior or appearance. They are written within the opening tag, such as ``, where "src" specifies the image source.
 - Some tags can self-close without a separate closing tag. For example, `` can be written as ``.
 - Comments in HTML allow developers to add notes within the code for improved readability and collaboration. They are written between `<!--` and `-->`, such as `<!-- This is a comment -->`.
 - HTML documents have a hierarchical structure, starting with the `<!DOCTYPE html>` declaration, followed by `<html>`, `<head>`, and `<body>` sections.
 - Semantic HTML involves using tags that convey meaning about content, enhancing accessibility and search engine optimization. Examples include `<header>`, `<nav>`, and `<footer>` for structural elements.
 - HTML5 is the latest version of HTML, introducing new elements, attributes, and APIs to support multimedia, graphics, and interactivity.
 - Validation ensures HTML code follows standards. Tools like W3C Markup Validation Service are used for validation.
 - Best practices include maintaining consistency in coding style and structure, using indentation and comments for clarity, and testing across different browsers and devices for compatibility.
 - Learning resources include online tutorials, documentation, and forums. Practice through projects and experimentation is essential.
 - HTML evolves continuously to meet changing web development needs, incorporating new features and improvements based on industry trends and feedback.
-

- HTML Syntax Variations:
 - HTML versions evolve over time, leading to syntax differences.
 - Analogy: Comparable to regional dialects within a country.
 - Variations may include rules for element usage, attribute syntax, and deprecated features.
- DOCTYPE Declaration:
 - Essential starting point in HTML document.
 - Specifies the HTML version used.
 - Purpose: Prevents quirks mode activation.
 - Quirks mode: Browser feature to handle non-standard or malformed HTML.
 - Quirks mode behavior varies between browsers, leading to inconsistent rendering.

- Types of DOCTYPE Declarations:
 - HTML4 and XHTML:
 - Detailed declarations categorized into:
 1. Strict: Enforces strict rules, disallows deprecated elements.
 2. Transitional: More lenient, permits deprecated elements.
 3. Frameset: Similar to Transitional but allows frameset content.
 - Example DOCTYPE declarations:
 - `php`
 - `Copy code`
 - `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
 "http://www.w3.org/TR/html4/strict.dtd"> <!DOCTYPE HTML PUBLIC
 "-//W3C//DTD HTML 4.01 Transitional//EN"
 "http://www.w3.org/TR/html4/loose.dtd"> <!DOCTYPE HTML PUBLIC
 "-//W3C//DTD HTML 4.01 Frameset//EN"
 "http://www.w3.org/TR/html4/frameset.dtd">`
 - HTML5:
 - Simplified declaration: `<!DOCTYPE html>`
 - Unifies HTML document structure, reducing complexity.
- Language Declaration:
 - Defined within the opening `<html>` tag.
 - Purpose:
 - Aids browser translators and screen readers.
 - Influences search engine indexing and results.
 - Facilitates language-specific styling.
 - Examples for specifying language:
 - `php`
 - `Copy code`
 - `<html lang="en"> <!-- English --> <html lang="es"> <!-- Spanish --> <html lang="fr">
 <!-- French -->`
 - Additional refinement: Specifying country variants, e.g., `lang="en-US"` for U.S. English.
- Adding Information to Webpage (Head Section):
 - The `<head>` element stores metadata and document information.
 - Functions as the document's "brain," providing essential information to the browser and search engines.
 - Information included:
 - Document title (`<title>` tag).
 - Metadata for search engines (e.g., `<meta>` tags).
 - Character encoding (`<meta charset="UTF-8">`).
 - Viewport settings (`<meta name="viewport" content="width=device-width, initial-scale=1.0">`).

- Keywords (<meta name="keywords" content="...">).
- Body Section:
 - Contains all visible content for the webpage.
 - Positioned after the <head> section.
 - Main area for adding, arranging, and styling page elements.
 - Global styles can be applied using CSS, referencing the <body> tag.
- Conclusion:
 - Understanding the foundational structure of an HTML document.
 - Importance of DOCTYPE declaration, language specification, metadata in the head section, and content in the body section.
 - Each component plays a crucial role in determining how the webpage is interpreted, displayed, and indexed.

Title: Understanding HTML Tags for Text Formatting

Visual Hierarchy:

- Definition: Visual hierarchy refers to the order in which information is perceived and processed.

Header Tags:

- Definition: Header tags, denoted by <h1> through <h6>, establish the hierarchy of content on a webpage, with <h1> being the most significant.
- Usage: They are typically used for titles or headings to differentiate important content from the rest of the text.
- Hierarchy Example:
 - <h1>Header Tag 1</h1>
 - <h2>Header Tag 2</h2>
 - <h3>Header Tag 3</h3>
 - <h4>Header Tag 4</h4>
 - <h5>Header Tag 5</h5>
 - <h6>Header Tag 6</h6>

Paragraph Tags:

- Definition: The <p> tag is used to organize text into paragraphs.

- **Functionality:** Adds default spacing above and below the text to distinguish paragraphs.
- **Usage:** Helps in maintaining readability and logical organization of content.

Line Break and Horizontal Rule Tags:

- `
`: Forces text onto a new line without starting a new paragraph.
- `<hr/>`: Places a horizontal rule across the width of the page, indicating a shift in topic or separation between content sections.

Emphasis Tags:

- ``: Used for emphasizing text, stretching out the emphasized word when spoken.
- ``: Highlights important words or phrases, making them spoken louder.

Italic and Bold Tags:

- `<i>`: Indicates text in italics for typographical formatting purposes.
- ``: Applies bold formatting to text without altering the volume or emphasis when spoken.

Quotation Tags:

- `<q>`: Encloses inline quotations, automatically adding quotation marks around the text content.
- `<blockquote>`: Indents quoted material to distinguish it from other content, typically used for longer quotations or pull quotes.

Citation Tags:

- `<cite>`: Identifies the source of a quotation or longer work title, such as a book or movie title.

Conclusion:

HTML tags play a crucial role in structuring and formatting text content on webpages, ensuring readability, logical organization, and semantic clarity. Understanding the purpose and usage of these tags is essential for effective web design and communication.

1. Introduction to Hyperlinks and Anchor Tags:

- Hyperlinks control navigation and resource access on websites.
- Hyperlinks connect to other files, webpages, or parts of the current webpage.
- Controlled by the `<a>` tag, or anchor-tag.
- `<a>` tags used to define hyperlink elements.

2. Hypertext Reference (href) Attribute:

- Most power controlled by the href attribute.
- href stands for "Hypertext Reference."
- Specifies the URL address the anchor tag directs to when clicked.
- Example: ``.

3. Styling and Behavior of Hyperlinks:

- Opening and closing tags wrap around content to be clicked.
- Default stylings and behaviors include underlined text, cursor change on hover, and color changes (blue, red, purple).
- Can hyperlink text, buttons, or images.

4. Types of URLs:

- Absolute URL: Full webpage address from beginning to end (e.g., `https://www.example.com`).
- Relative URL: Specifies location relative to the current page (e.g., `index.html`).

5. Navigation with Relative URLs:

- Useful for linking between files within own directories.
- Example: `page1.html` for a page in the same folder.

6. Navigation with Subfolders:

- Use forward slashes to specify subdirectories.
- Example: `images/image1.jpg` for a file in an "images" subfolder.

7. Navigating Back Out of Subfolders:

- Use `../` to move up one directory level.
- Repeat for multiple levels.
- Example: `../page1.html` to navigate back out of a subdirectory.

8. Benefits of Relative Links:

- Maintain links even if domain name changes.
- Useful during web design offline.
- Ensure links function correctly when website goes online.

9. Absolute Links for External Resources:

- Necessary for linking to websites outside domain.
- Example: ``.

10. Linking to Specific Sections on Webpages:

- Assign an ID selector to elements.
- ID uniquely identifies the element.
- Example: `<p id="paragraph11">...</p>`.

11. Linking to Specific Sections from Another Page:

- Use page name followed by ID preceded by #.
- Example: ``.

12. Target Attribute for Link Behavior:

- Control how links open.
- `_self` opens in the same tab/window.
- `_blank` opens in a new tab/window.

13. Best Practices for Link Behavior:

- Consider user experience and expectations.
- Default behavior is to open links in the same tab/window.
- Anticipate user needs and browsing habits.

14. Naming Conventions for URLs:

- Avoid special characters and spaces.
- Special characters replaced with percent symbol followed by hexadecimal value.
- Blank spaces converted to "%20" or "+".

15. Best Practices for Naming Conventions:

- Use alphanumeric characters, hyphens, and underscores.
- Ensure legibility, consistency, and meaningfulness.
- Choose a naming convention and apply it uniformly.

Conclusion:

- Understanding hyperlinking, URL types, target attributes, and naming conventions crucial for effective web development.

- Adherence to best practices ensures user-friendly browsing experiences and consistent functionality across different platforms and devices.
-

- Sharing Website Online:
 - Key to reaching a wider audience is getting website files online.
 - Requires understanding how to upload files and how people find them.
- File Transfer Protocol (FTP):
 - Language enabling computers to send files between each other.
 - Involves uploading files from local computer to a server and downloading files from a server to a computer.
- Getting Website Online:
 - Requires a domain (location) and using FTP to move files.
 - Local version of website resides on the computer; online version is hosted on a server and accessible to others.
- FTP Usage:
 - Website-hosting services often provide graphical user interfaces (GUI) for FTP.
 - GUI includes visual cues and images to assist designers.
 - Common option for beginners in web development.