Introduction to HTML

1. Introduction

- HTML stands for **HyperText Markup Language**.
- HTML is not a programming language. It is a **markup language** that is used to structure web pages.
- HTML is the web standard for website development.

Key Terms:

- **Element**: A building block of HTML (`<p>`, `<h1>`, `<img>`).
- **Tag**: Keywords surrounded by alligator (angle) brackets (`<html>`, `</html>`).
- **Attribute**: Provides additional information about an element, or gives the location of a resource to the element. (`src`, `href`).

2. HTML Structure

****The following retains the general structure of HTML in a code file, the bold is just comments I used to annotate the code, generally these will not be there****

```
<!DOCTYPE html> - start with the DOCTYPE
<html>          - Open HTML Tag (Beginning of HTML Element)
  <head>        - The head holds metadata (information not shown to user)
    <title>My First Webpage</title> - The title shown on the tab, above url bar
  </head> - Closing the head Element
  <body> - Begin Body Element: data shown in the browser window is put in here
    <h1>Hello, World!</h1> - Biggest heading possible (h2,h3,…,and are smaller)
    <p>This is my first HTML page.</p> - A block of text “paragraph”
  </body> - Closing the Body element
</html> - Closing HTML element
```

To provide more context:

- `<!DOCTYPE html>`: Declares HTML for the web browser to properly render this using HTML standards.
- `<html>`: Root or the base of the HTML document.
- `<head>`: Metadata, This is where you reference script and style files. Can also provide more context here on how to deal with mobile website loading.
- `<body>`: What you want to user to see goes here.

3. Common Elements used

Tag	Purpose
`<h1>` to `<h6>`	Headings, from largest to smallest
`<p>`	Paragraph, a sentence or more
` `	Inserts a line to separate text
`<hr>`	Inserts a break denoted by a line that can be used to separate sections.
`<strong>` / `<em>`	Bold / Italic text, respectively.
`<a href="">`	A link, the href portion is an attribute where the url is placed
`<img src="" alt="">`	Insert an image, src is an attribute that tells HTML where this image is stored. Alt is used to show this text in the case the image fails to load.
`<ul>` / `<ol>` / `<li>`	Lists, ul is unordered (bullet points) ol is ordered (numbers) li is a list item that goes in these lits. Can be styled to be arrows or letters dependent on whether it's ul or ol.
`<div>` / `<span>`	Generic containers

4. Attributes

Common Attributes:

- `href` – hyperlink reference, the address you want the link to go to.
- `src` – image source, where the image is stored.
- `alt` – alternative text for failed loading of an image, etc.
- `width`, `height` – to change size of an element.
- `class`, `id` – for CSS/JS usage, helps in specific selection of an element.
    - Classes can be used for a grouping of similar Elements
    - Ids are unique and used for fine grain selection of specific Elements

Example:

<img src="cutepuppy.jpg" alt="A cute puppy" height="100%" width="350">

5. Nesting & Indentation

- Elements can be **nested** inside one another;

  like in a list, li is **Nested** in the ul Element

```
<ul>
  <li>Item 1</li>
  <li>Item 2</li>
</ul>
```

- Use consistent **indentation** (1 tab or 4 spaces) for easy readability.

6.Semantic Tags

Why use them?

Semantic tags are a way to group elements in a much more descriptive way than .

Div soup (many divs) is great to group things together to provide structure. But semantic tags do it while being more descriptive, these do not add any visual structure to the page though.

Tag	Purpose
`<header>`	Used to group introductory elements together, IE the heading and a small description, etc
`<main>`	Holds the main content together (groups paragraphs and subheadings etc.)
`<footer>`	To group the bottom text, like a copyright line or a created by: section.
`<section>`	Groups by content theme, etc
`<nav>`	Navigation links, used commonly to create navigation bars. Holds links together.

Many more semantic tags exist, but these are the common ones used.

8. Tools for Writing HTML

- Code Editors: **VS Code, VIM, notepad… (don’t let me catch you using notepad, I will roast you)**
- Web Browser Developer Tools (right-click → Inspect Element, or in some browsers F12, or also found in the more tools tab)
- Online editors: Codespaces, Replit, etc

9. How to write HTML that won’t get you roasted:

- Always close tags (even self-closing like `<img />`)
- Use semantic tags when it is possible.
- Use indentation correctly (a tab is usually enough and use 1 tab for every level of nesting. 1 tab is roughly 4 spaces)

10. HTML Forms

Forms are a way to take input from a user.

Although we are learning forms they do not work, for that we need Javascript, We will learn that next week! Stay curious.

The general structure is:
<form action="/submit" method="POST">

<label for="name">Name:</label>

<input type="text" id="name" name="name">

<label for="email">Email:</label>

<input type="email" id="email" name="email">

<input type="submit" value="Submit">

</form>

Take some time to research this and discuss with a partner.

instructor note: have a class discussion on what we think, like and dislike, etc. Where have we seen forms, and what webpages possibly use forms.

🔹 Important Attributes:

- `action`: URL where the form data is sent (e.g., `/submit`).
- `method`: HTTP method used – usually `GET` or `POST`.

🔹 Common Input Types:

Type	Purpose
`text`	Single-line text input
`email`	Validates email format
`password`	Hidden input for passwords
`checkbox`	Multiple selections
`radio`	One selection from a group
`submit`	Submit button
`reset`	Clears the form
`number`, `date`, `range`, etc.	Specialized inputs

🔹 Other Elements:

- `<label>`: Describes the input field.

- `<textarea>`: Multi-line text box.
- `<select>` + `<option>`: Drop-down menu.
- `<fieldset>` + `<legend>`: Group related fields.