

HTML - Hypertext Markup Language

What is it?

- HTML is the standard markup language for web browser documents.

Why does it matter?

- It is the foundation for other assisting technologies commonly used, such as CSS (see below) and JavaScript (programming/scripting language).

Where would it be used in real projects?

- HTML is seen in every single webpage, even in modern projects, HTML is used as the basic building block.

Common terms:

- **Tags** are how you initiate every single HTML element. For example:

```
<p>
```

This is a paragraph. The tag above begins the HTML element, and the one below ends it, the closing tag uses a forward slash '/' to show that the element has ended.

```
</p>
```

- **Head Tag:** Container for metadata and is placed between the <html> and <body> tag.
- **Metadata:** "data about data", or data that provides information about one or more aspects about data. E.g., the time and location data connected to a photo you took.

- **Title Tag:** The “title” of the entire document. Is displayed in the tabs of modern web browsers alongside the favicon.
- **Favicon:** The icon displayed on the tabs of your web browsers. Often times displayed on search engine results as well.
- **Body Tag:** The body tag is generally where a majority of content that is displayed to the user goes.
- **Footer Tag:** Tag that contains the company or author’s information, as well as copyright information, backlinks, etc.
- **Style:** In regards to web development a style or stylesheet is usually defined in the head tag. Style **can** be defined in the head via ‘<style>’ or linked via ‘<link>’; most devs now put their CSS in external files.
- **Script:** Scripts can be placed in the <head>, but placing them at the **end of the** <body> is a common performance practice to avoid blocking page rendering.
- **‘A’ Tag:** Tag used to define and place hyperlinks.
- **Hyperlinks:** Simply a link that either takes the user to another webpage or a different section of the webpage.
- **Mailto:** Uniform Resource Identifier (URI) scheme that allows users to create hyperlinks that open a user's default email client with a pre-filled "To," "Subject," and/or "Body" field. Used within the ‘a’ tag.

Core Syntax and Examples:

- To begin an HTML document, you must declare it as one:

```
<!DOCTYPE html>
```

- Everything within the HTML document must then be placed **within** the opening and closing tags:

```
<html>
...
</html>
```

- From there, common practice is to create the **head** section of the document. The head section is where developers usually place the **metadata** of the document itself. Things such as the **title**, **style**, **scripts**, and other **meta** information is placed within it. For simplicity, I will just give you an example of the head section with the title tag:

```
<head>
<title>Page Title</title>
</head>
```

- It's worth noting that the title tag is what is displayed alongside the **favicon** (the "favorite icon") on the tab in your web browser. In the example above, "Page Title" would be shown on the tab.
- The next section that follows the head section is the **body** section (who would've guessed). Much like the body of an essay, the body section is where a majority of your HTML elements are going to be placed:

```
<body>
<h1>I am a header.</h1>
<p>
As you know I am a paragraph, and I am used to place a lot of text in
a formatted block.
</p>
</body>
```

- In this example I used the **h1** tag, or the header tag. This tag is used to, you guessed it, make header text. You should note that the number one in h1 tells the computer that it needs to make that header a specific size. The sizes in HTML tags start from greatest to least, so for a slightly smaller sized header, you would use the h2 tag.
- Almost all content being displayed to the user of the webpage will be placed within the body tags. That being said, this is technically not a requirement, it is

just common practice among other web developers. You'll find that following common practices is helpful not only to you but to everyone involved.

- Past the body tag, most webpages also make use of the **footer** tag. The footer commonly used for author information, backlinks, copyright information, etc.

```
<footer>
<p>Author: John Doe</p>
<p><a href="mailto:example@email.com">example@email.com</a></p>
</footer>
```

- In the example above, I used the footer tag to display the author's name along with a link to their email. The **a** tag is used for **hyperlinks**, in this case I used a **mailto** url that opens the user's default email client and inputs the author's email.
- At the end of an html document, you will usually see the following closing tag:

```
</html>
```

- This indicates that anything placed past this tag is generally not valid. You must place all content in between the opening and closing html tags. Overall, it's best practice to close with </html>, even though some browsers can still render a page without it.

Further References:

- A great reference point for all HTML (and CSS) practices is [W3 Schools](https://www.w3schools.com/), I often go back and double check my own work using their in depth explanations.

Full Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Website</title>
  </head>
  <body>
    <h1>Welcome!</h1>
    <p>This is my first webpage.</p>
  </body>
</html>
```

CSS - Cascading Style Sheets

What is CSS?

CSS stands for **Cascading Style Sheets**. It's the language used to **style and visually arrange** HTML content on a web page.

Why does it matter?

Without CSS, all web pages would look like plain black-and-white text. CSS adds **color, layout, fonts, spacing**, and responsiveness—everything that makes a site look professional.

Where is it used?

Any modern website or app—like Instagram, YouTube, or online stores—relies heavily on CSS for layout and design.

- Think of HTML as the skeleton and CSS as the clothing placed on top of it.

Basic Concepts and Terminology

- **Selector:** The HTML element you want to style (like p, h1, etc.)
- **Property:** The aspect of the element you want to change (like color or font-size)
- **Value:** the specific style you're applying (like blue, 16px, or bold)
- **Rule:** A full line of CSS (e.g., color: blue;) inside a CSS block.
- **Declaration Block:** A group of CSS rules inside of { }

Core Syntax and Examples

The following is a basic CCS rule:

```
p {  
    color: blue;           /* Text will be blue */  
    font-size: 16px;      /* Text will be 16px tall */  
    line-height: 1.5;     /* Adds vertical space between lines */  
}
```

This example targets elements with the class 'button':

```
.button {  
    background-color: red; /* background will be red */  
    color: white;          /* text will be white */  
    padding: 10px;         /* space inside the button */  
    border-radius: 5px;    /* rounded corners */  
}
```

Practical Uses:

A couple common ways CSS is used includes:

1. Customizing a portfolio site. For a fun and easy personal site, checkout [neocities!](#) They host the website for you and you are able to make a personalized website by editing your own HTML and CSS.
2. Making websites mobile-friendly. With CSS you are able to make sure your sites look good on both phones and desktops.

Live Demo:

Here is where I will provide you with a simple step-by-step guide to create a simple webpage using both HTML and CSS!

1. Setup:

- a. Create a directory called “my-webpage” (or anything else you’d like)
- b. Inside the directory, make two files: index.html and style.css

2. Writing the code:

- a. Using [Visual Studio Code](#), or another text editor/IDE. Type in the following code (or something similar, as long as it has valid syntax and it follows best practices):

index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Webpage</title>
    <link rel="stylesheet" href="style.css">
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
      initial scale = 1.0">
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>This is a paragraph.</p>
    <button class="button">Click Me</button>
  </body>
</html>
```

style.css

```
body {
  background-color: #f0f0f0;
  font-family: Arial, sans-serif;
```

```

}
h1 {
  color: teal;
  text-align: center;
}
p {
  color: #333;
  font-size: 18px;
}
.button {
  background-color: teal;
  color: white;
  border: none;
  padding: 10px 20px;
  cursor: pointer;
}

```

3. Viewing your webpage!

- a. Open your index.html file in your browser. You can do this by right-clicking the file in your file explorer and choosing “Open with...” and clicking on your web browser.
- b. You should then see your webpage with the stylized css elements.
- c. Feel free to play around with both the HTML and the CSS!
- d. *Also note the use of the link tag, which is used to point to the external css file.*
- e. The two **<meta>** tags were added for compatibility purposes.
 - i. The first tag defines the character set being used, and the second helps the website scale to every device.

```
<link rel="stylesheet" href="style.css">
```