

Hyper Text Markup Language



Hyper Text Markup Language

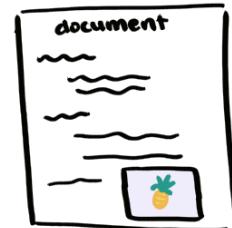
better known as *HTML*

HTML is the series of elements that organize and define the content on a webpage. Different elements are used to enclose, or wrap, different parts of the content to make it appear a certain way, or act a certain way. For example, content could be structured within a set of paragraphs, a list of bulleted points, or using images and data tables.

HTML files are referred to as **documents**.

This is because HTML builds something called the Document Object Model - or the DOM! It defines the logical structure of documents/webpages and the way a document/webpage is accessed and manipulated. It's the standard method for representing and interacting with elements

Kinda the
same!



The Anatomy of an HTML Element

<Opening Tag> Content </Closing Tag>

The Opening Tag:

The opening tag is the element identifier placed between angle brackets. For example, to create a paragraph element, the opening tag would look like `<p>`

The Content:

This is the content of the element, which looks different for different types of elements. You can also place other elements inside an element, by placing it as or with content. In the paragraph example, the content would just be a paragraph of text.

The Closing Tag:

The closing tag is just a forward slash then the element identifier placed between angle brackets. This is to signify the end of an element. The paragraph element closing tag would look like `</p>`

The Anatomy of an HTML Document

<!DOCTYPE html> This tag is the first tag of the HTML file and basically tells the computer - hey! I'm using HTML! It does not need a closing tag!

<html> </html> All web pages start with the html element. It's also called the **root element** because it's at the root of the tree of elements that make up a web page.

<head> </head> The head element contains information about the web page, as opposed to the web page content itself.

<body> </body> The body element appears after the head element in the page. It contains all the content of your web page: text, images, so on.

Most editors and IDEs will have a basic HTML document set up already. Let's break it down!

The diagram illustrates the hierarchical structure of an HTML document. A red bracket on the left side groups the outermost tags: <!DOCTYPE html>, <html>, </html>. Inside the <html> tag, a blue arrow points to the <head> tag. Another blue arrow points from the <head> tag to its content: <meta charset="utf-8">, <meta name="viewport" content="width=device-width">, <title> Ms. Orret's HTML Document </title>, and <link href="style.css" rel="stylesheet" type="text/css"/>. A blue arrow then points to the </head> tag. A green arrow points from the </head> tag to the <body> tag. Inside the <body> tag, a blue arrow points to its content: <script src="script.js"></script>. A green arrow then points to the </body> tag. Finally, a red arrow points from the </body> tag to the </html> tag at the bottom.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
    <title> Ms. Orret's HTML Document </title>
    <link href="style.css" rel="stylesheet" type="text/css"/>
  </head>
  <body>
    <script src="script.js"></script>
  </body>
</html>
```

HTML Tags You Know:

just a quick review!

Header tags:

<h1> This is a header! Headers have a hierarchy, much like the headers of an essay or research paper. h1 headers are the overarching headers, h2 are subheaders, h3 are the next level of subheads, and so on. **</h1>**

h1

h2

h3

h4...

Paragraph tag:

<p> This is a paragraph! This tag is used to create elements of blocks of text for your page! **</p>**

Link tag:

**** This is the text that the hyperlink will actually show up as! ****

CLICK ME!

Image tag:



Break tag:



**
**

Input Tag (Click for Reference!)

radio buttons

check boxes

sliders

buttons

text inputs

<input type = "typeOfInput" >

Icon tag:

<i type = "class name for icon" ></i>

HTML Tags to Know

There are so many tags out there to know... how can we sort through them all?

Usually, the best way is to just google for the element that you are looking to use and find the tag and documentation. **Here is a alphabetized and complete element reference.**

Another great resource are HTML cheat sheets, that document and help you build the most common tags. Here are a few!

HTML Interactive Cheat Sheet

HTML Simple Cheat Sheet

HTML Detailed Cheat Sheet

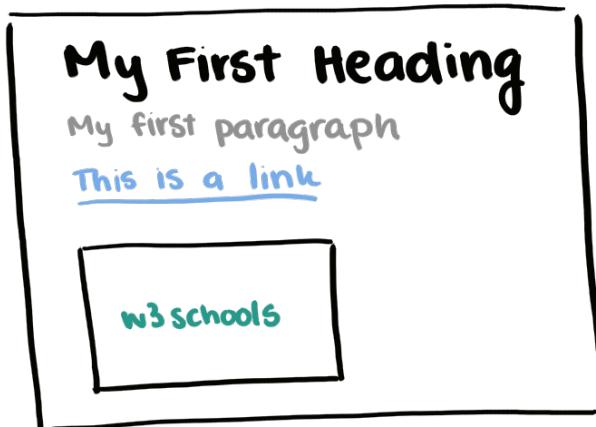
A Simple HTML Example

```
<!DOCTYPE html>
<html>
<body>

    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
    <a href="https://www.w3schools.com">This is a
link</a>
    

</body>
</html>
```

looks
something
like
this →



HTML Attributes

HTML attributes are special variables and values that define the elements behavior and characteristics.

<Opening Tag attribute_name = "attribute_value">

Some content goes here

</Closing Tag>

HTML Attributes Used for All Elements:

id The id attribute assigns a unique identifier for an HTML element. This is like giving the element an ID number! The id attribute can be used to identify elements in a CSS style sheet and style them individually. It can also be used by a Javascript script to access and manipulate specific elements or their values!

class The class attribute assigns elements to a “class” or group, to be able to style and use them together. Elements can be part of multiple classes and multiple elements can share one class.

Common HTML Attributes Specific to Specific Elements:

src Specifies the location of an external resource, such as a picture or video. The src attribute can be used for the following elements:
<audio>, <embed>, <iframe>, , <input>, <script>, <track>, <video>

href For <a> and <area> elements, the href attribute specifies the URL of the page the link goes to. For <base> elements, the href attribute specifies the base URL for all relative URLs on a page. For <link> elements, the href attribute specifies the location (URL) of the external resource (most often a style sheet file).

alt The alt attribute provides alternative information for an image if a user for some reason cannot view it (because of slow connection, an error in the src attribute, or if the user uses a screen reader). It can be used for the <area>, , and <input> tags.

type For <button> elements, the type attribute specifies the type of button. For <input> elements, the type attribute specifies the type of <input> element to display. For <embed>, <link>, <object>, <script>, <source>, and <style> elements, the type attribute specifies the Internet media type

Using Divs to Organize Your Documents

The div tag is known as the Division tag. This tag allows us to group elements and data in an HTML document together. It's a bit like grouping images or drawings together when doing a graphic design project. It can be especially helpful when you want to style a bunch of elements as one, and you don't want to have to define the class for each one. Take a look at the example below!

```
→<div class="myDiv">
  <h2>This is a heading in a div element</h2>
  <p>This is some text in a div element.</p>
  <img src = "img.jpg" alt = "This is an image in a
div element">
  <a href= "thisisalink.com"> This is a link in a
div element.</a>
</div>
```

Then you can style everything inside the div tags together with the class semester "myDiv"

```
.myDiv {
  border: 5px outset red;
  background-color: lightblue;
  text-align: center;
}
```

This is a heading in a div element

This is some text in a div element.

[This is a link in a div element.](#)

You can also put divs inside of divs inside of divs inside of divs! Just make sure you keep track of what divs contain what elements! Think of it like Russian dolls!



Tables, Lists, and Forms

Tables

The **<table>** tag defines an HTML table.

Each table row is defined with a **<tr>** tag.

Each table header is defined with a **<th>** tag. This also defines the number of columns.

Each table data/cell is defined with a **<td>** tag.

```
<table style="width:100%">
```

```
<tr>
  <th>First Name</th>
  <th>Last Name</th>
  <th>Time Logged Out </th>
</tr>

<tr>
  <td>Jill</td>
  <td>Smith</td>
  <td> 2:10PM </td>
</tr>

<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td> 5:40PM </td>
</tr>

</table>
```

By default, the text in **<th>** elements are bold and centered.

By default, the text in **<td>** elements are regular and left-aligned.

First Name	Last Name	Time Logged Out
Jill	Smith	2:10pm
Eve	Jackson	5:40pm

Lists

HTML supports three types of lists: unordered, ordered, and description. An unordered list is like a bullet list - the order isn't noted in any way. An ordered list is like a numbered list - the order matters! And a description list is a list with a description given to each item.

Unordered Lists

An unordered list starts with the `` tag. Each list item starts with the `` tag. The list items will be marked with bullet points by default:

```
<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

The handwritten notes show the words 'coffee', 'tea', and 'milk' aligned vertically next to the corresponding list items. The first two items are preceded by a solid black circle bullet point, while the third item is preceded by a hollow circle bullet point.

- coffee
- tea
- milk

Ordered Lists

An ordered list starts with the `` tag. Each list item starts with the `` tag. The list items will be marked with numbers by default:

```
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

The handwritten notes show the numbers '1.', '2.', and '3.' aligned vertically next to the corresponding list items. Each number is followed by a solid black circle bullet point.

1. coffee
2. tea
3. milk

Description Lists

The `<dl>` tag defines the description list, the `<dt>` tag defines the term (name), and the `<dd>` tag describes each term:

```
<dl>
  <dt>Coffee</dt>
  <dd>- black hot drink</dd>
  <dt>Milk</dt>
  <dd>- white cold drink</dd>
</dl>
```

The handwritten notes show the terms 'coffee' and 'milk' aligned vertically next to their descriptions. Each term is preceded by a solid black circle bullet point. The descriptions are preceded by a dash symbol.

- coffee
 - black hot drink
- milk
 - white cold drink

Forms

An HTML form is a collection of elements to collect user input in some way. Remember those DOM elements like checkboxes, text inputs, radio buttons, sliders, etc? Those can all be used to create forms to be able to collect and use user data!

The **<form>** tag is used as a container for these different types of input elements!

```
<form>
```

form elements

```
</form>
```

The **<input>** element is the most used form element. Here are the most common input elements:

<code><input type="text"></code>	Displays a single-line text input field
<code><input type="radio"></code>	Displays a radio button (for selecting one of many choices)
<code><input type="checkbox"></code>	Displays a checkbox (for selecting zero or more of many choices)
<code><input type="submit"></code>	Displays a submit button (for submitting the form)
<code><input type="button"></code>	Displays a clickable button

The **<label>** tag defines a label for many form elements. To attach the label to an input element, set the `for` attribute of the label to the `id` of the element you are labeling!

Check out this simple example with radio buttons!

```
<p>Choose your favorite flavor:</p>
```

```
<form>
  <input type="radio" id="choc" name="c_flavor" value="chocolate">
  <label for="choc"> Chocolate </label><br>
  <input type="radio" id="van" name="v_flavor" value="vanilla">
  <label for="van"> Vanilla </label><br>
</form>
```

Using Javascript to Access HTML Elements and their Data

So as we know, HTML is NOT a programming language. That's because HTML is just a way to organize content, it has no logical way to manipulate or use this content dynamically. So what do we do when we as web developers DO want to manipulate this content dynamically?

JavaScript is the programming language built for web development - which means it is the perfect tool to use with HTML! But because they're working together but are not the same language or live on the same file, we need to understand how to communicate this data and content back and forth between them.

First things first, how do I connect a JS file to my HTML? You just need to add a <script> tag to your body! It will look something like this:

```
<body>
  <script src="script.js"></script>
</body>
```

So now, we need to understand how to access, change, and parse through specific elements in our document. Let's start with accessing an element!

The easiest way to access a specific element is by accessing it through an id. You can use the getElementById() function to store the entire element and its properties as a variable in your JS script. That looks like this:

```
var element = document.getElementById(id);
```

Now, you can add styles to your element based on certain conditions, manipulate attributes, or access data! Here's some functions to get you started - but remember - there's a lot more out there you can do! Don't be afraid to search for more!

```
document.getElementById("demo").innerHTML = "Paragraph changed!";
```

```
document.getElementById("demo").style.color = "red";
```

```
document.getElementById("demo").setAttribute("src", "newPic.jpg");
```

Random HTML Extras

Creating Multiple HTML Pages on a Site

Seems like it should be complicated? It's not!

All you need to do is create a new HTML file for your new page, and then use the path name of that file as your link! You can then link it to a button that leads to another page, a hyperlink, a menu, etc.

```
<a href = "newPage.html">  
<input type="button" value="New Page">  
</a>
```

Adding Icons

First, you need to pick a library of icons to use. I recommend using the Font Awesome 4.7 Icons - but there are a lot of different style sheets to choose from!

Then you will include the tag for that icon style sheet in the head of your document. For the Font Awesome 4 Icons it looks like this:

```
<link rel="stylesheet" href="https://  
cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/  
css/font-awesome.min.css">
```

Then you can use the icon cheat sheet to find the correct class for the icon you want!

Here is the link: <https://fontawesome.com/v4.7/cheatsheet/>

 fa-500px []	4.4	 fa-address-book []	4.7
 fa-address-card-o []	4.7	 fa-adjust []	
 fa-align-justify []		 fa-align-left []	
 fa-ambulance []		 fa-american-sign-language-	4.6

A Rainbow of Useful HTML Links

You Only Need 10 HTML Tags

HTML Basics Tutorial

HTML Canvas Tutorial

Basic HTML Examples

HTML Cheat Sheets

HTML Generators