Introduction

So I did sit down with the intention of writing an epic-poem-length guide to HTML and CSS. But before I did, I thought it would be useful to see if that had been done before. It has. 8 billion times.

HTML is a markup language that is used by web browsers in order to structure a web page or web app’s information and features.

- It is NOT a programming language
- It stands for HyperText Markup Language

I still made you a video where we take a Google Doc and create a website that approximates the content. I think these are interesting ways of getting our hands dirty with HTML. If you watch this, you should come away with a better appreciation of how HTML is used to structure information and media.

Slides: https://docs.google.com/presentation/d/1r3h2F7b0lXv95N6W-Blj9wMtDiGCENwKPjshsX/nc/edit?usp=sharing
HTML Tags

HTML is almost entirely made up of tags with identifiers between the `<` and `>` and content between the opening and closing tags.

There are 113 HTML tags. Here’s a full and useful reference: https://htmlreference.io/

Anatomy of an HTML Page

HTML pages have a standard markup. One of the simplest HTML markup you could author might be this:

```
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
</head>
<body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
</body>
</html>
```

- The `<!DOCTYPE html>` declaration defines this document to be HTML5
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the document
- The `<title>` element specifies a title for the document
- The `<body>` element contains the visible page content
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph

At a minimum your HTML page should include the `<!DOCTYPE html>` declaration, opening `<html>` and closing `</html>` and opening `<body>` and closing `</body>` tags between the opening and closing
I recommend using a boilerplate generator if you have trouble remembering how to write the basic markup—check out HTML Shell (http://htmlshell.com/)

**<DIV’S>**

HTML is all about structure, and the `<div>` tag is one of the more useful and popular ways of creating sections in an HTML document and is often used as a container for other HTML elements.

They don’t do anything on their own, but are extremely useful when combined with CSS (cascading style sheets) because they allow you to apply different styles of different `<div>` elements.

```html
<blockquote>
<p>Yup, I wanna be a bear</p>
<div class="warning">
<img src="https://images.unsplash.com/photo-1530595467"
<p>Beware of the bear</p>
</div>
</blockquote>

blockquote {
  margin: 0;
}

p {
  margin: .2rem;
}

.warning {
  border: 10px ridge #f00;
  background-color: #ff0;
  padding: .5rem;
  display: flex;
  flex-direction: column;
}

.warning img {
  width: 100%;
}
```
Don’t worry too much if you don’t understand the `styles.css` file. We’ll cover that in the next article. Pay more attention to what is within the `<div>` tags:

- `<div class="warning">` — the opening tag, we assign a class name in order to style it in our `styles.css` file.
- `<img src="superlongurlhere">` — here’s our image (I used one from unsplash.com that happened to have an extremely long URL.
- `<p>Beware of the bear</p>` — here’s our `<p>` paragraph tag. This is standard HTML, but we also style it in our CSS
- `</div>` — our closing `<div>` tag

### Using the Inspector to Inspect Websites (and WebApps)

Let’s see how div’s are used in the wild. We can explore this very webpage actually. Your browser’s development tools has a very useful feature.

1. Open your Development Tools (or right-click and select `inspect`)
2. In the Elements Tab, you’ll see an interactive view of this page’s HTML
3. Mouse-over each element within the `<body>` tag until you get to a the `<div>` that holds all of this site’s content. Expand it.
4. Continue to mouse over the child `<div>`s and try and find the div that contains the site header title (applab)
5. Change it to `catlab` by directly editing the HTML of the page
You can also directly inspect and edit the CSS associated with a specific element on the page. You can edit, add and remove CSS properties in the Styles tab of the inspector.