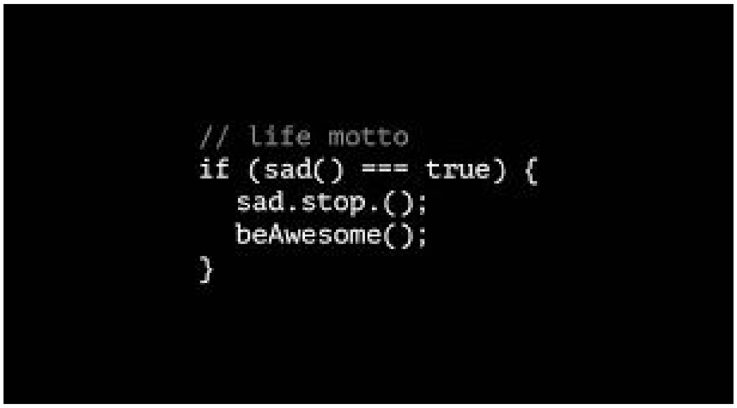
Ninth Lesson: Continuation of HTML



```
// life motto
if (sad() === true) {
   sad.stop.();
   beAwesome();
}
```

We hope everyone is enjoying their four-day weekends! Make sure to take a break from staring at a screen!

This joke isn’t written in HTML, but you get the point.

Image Tags and Other Attributes (src and alt)

Including images in your website is a great way to make it user-friendly! They’re an often necessary break from wordy paragraphs and a blank background. What’s more, they’re incredibly easy to incorporate.

If you’re using Atom:

There are two different ways to incorporate images into your HTML project: using a downloaded image file or an image link from the web.

If you want to go with the first way, you’ll have to place the image you want into the same project folder as the .html file. This will make it accessible to you when you open the project in Atom. From there, you’ll just need to copy the source file (`src`) of the image. You can do this by two-finger tapping the name of the file. This will typically look like `Name.jpg` or something similar.

The basic format of an image tag is:

```
<img src=”Name.jpg” alt=”Name”>
```

As you can see, there isn’t a closed image tag; it’s one of those special ones. The extra information, like `src` and `alt` are known as attributes.

`src` is, as stated above, your way of telling the HTML where to go to find this image--if you have a file, it will find that file in your folder and display that. If you are looking to use a link (the second way), you would paste the link into the src tag, instructing the code to take the image from the link provided.

The `alt` attribute comes into play when the image can not be displayed. It provides an alternative explanation for what the image was supposed to represent. If you've ever been to a website when your internet wasn't working properly and there's a large box with some text inside of it where the picture should be, you've seen an example of the `alt` attribute.

If you wanted, you could also include other attributes in the `img` tag, like determining the `width` and `height` of the image. However, it's typically simpler to adjust these later on in a separate CSS file, using the `id` attribute (which will be discussed later).

It's important to note that HTML is not case-sensitive, so it doesn't matter if you write `img` or `Img`. It also doesn't matter whether you decide to use single or double quotes around the attribute, as HTML will accept either.

Style Attribute and Other Attributes (But Mostly Style)

Although this doesn't pertain to images, you can add various `style` attributes to a paragraph or a heading. `style` attributes basically give your website some spice; it allows you to manipulate the layout and UI of your webpage to suit the aesthetics you are looking for. These include color, font, font size, margins, borders, etc.

For example, to change the text color of a paragraph:

```
<p style=”color:blue;”>This is a blue paragraph wowzers.</p>
```

To change the background color of a selected text (heading, paragraph, etc), you would still use the `style` attribute, but what goes in the parentheses would be different--as you probably guessed.

```
<p style=”background-color:blue;”>This is a paragraph with a
blue background wowzers.</p>
```

One thing to note about colors is that you can use the basic words for them like “blue” or “pink”, but they can also be notated by HEX values, which will give you a larger selection of colors to choose from. Attached here is a link to a list of HTML HEX color values from W3Schools (great resource for HTML by the way).

Here is a short list of some other attributes (relating more to your text) that you can use in the `style` attribute, written as an example you could input into your code:

```
<h1 style="font-family:verdana;">This is your new magical
heading.</h1>
<h1 style="font-size:150%;">This heading should looker larger than
normal.</h1>
```

```
<p style="text-align:center;">This paragraph should look like what
happens to your text in google doc when you hit the center align
button (because they--surprise surprise--do the same thing.</p>
```

Another important one is the `title` attribute. This provides some information about the element that gets displayed when your mouse hovers over it in what is called a “tooltip.” You would simply code: `title = “Explanation here”`. `href` is also an attribute that you’ve seen before (in links!).

Lists and Tables

Once again, we will be talking about lists -- except this time, it is a UI aspect instead of a container of information. There are two different types of lists in HTML: ordered and unordered.

Ordered lists are basically numbered lists; if you don’t know what that looks like, here it is:

1. No 1
2. No 2
3. No 3

Unordered lists are your bulleted lists; for reference, they look like this:

- Uno
- Uno
- Uno

The syntax for ordered lists differ from that of unordered lists, but not by much. The tag for ordered lists is `<ol></ol>` while the tag for unordered lists is `<ul></ul>`. Remember that the o in ol corresponds to ordered while the u in ul pairs with unordered, and you will not confuse the two.

To place items in the list requires the `<li></li>` tag, which goes between the opening and closing tags of either your ordered or unordered list. Here’s a syntax example of both, starting with unordered lists:

```
<ul>
  <li>Cookies</li>
  <li>Brownies</li>
  <li>Deadpool</li>
</ul>

<ol>
  <li>George Washington</li>
  <li>John Adams</li>
  <li>Thomas Jefferson</li>
</ol>

Output:
   ●  Cookies
   ●  Brownies
   ●  Deadpool
   1. George Washington
   2. John Adams
   3. Thomas Jefferson
```

HTML also has tables, which are used to display information. The table is quite simple. Let's say we are trying to code a table that looks like this:

First Name	**Last Name**	**Grade**
Emma	Li	11
Nicole	Men	12

To do this, we would code something like this (this example is somewhat copied from W3Schools):

```
<table> <!-- table tag -->
  <tr> <!-- tr dictates each row -->
    <th>First Name</th> <!-- th is the headings of columns -->
    <th>Last Name</th>
    <th>Grade</th>
  </tr>
  <tr>
    <td>Emma</td> <!-- td = inputs in each column the table -->
    <td>Li</td>
    <td>11</td>
  </tr>
  <tr>
    <td>Nicole</td>
    <td>Men</td>
    <td>12</td>
  </tr>
</table>
```

As you can see, the `<tr>` tag splits the rows, while the `<th>` tag contains a description of the content to follow in the next columns. `<td>` tags contain the regular information you are displaying.

How to Inspect
If you see a website and want to know how they coded something, you can always view page source or inspect it.

If you want to see the HTML source code, you just need to two-finger click any part of an HTML page, and select “View Page Source.” This will open a new tab, where you’ll see the HTML detailed.

You can also inspect the HTML elements that make up a page. To do so is similar to viewing the source code; just two-finger click on an element and select “Inspect.” You’ll be able to see the HTML and CSS of the site, as well as edit (if you so desire). What’s interesting about inspecting is if you hover over a line of code, the element that it corresponds to visually will be indicated as well.

This concludes the ninth lesson. By this point, you have learned enough HTML to construct a very basic website. We encourage you to try it! Next lesson, we’ll start to incorporate some more CSS elements to help you improve the aesthetics of your page. You’ve already learned a bit of CSS in the style element--HTML and CSS go hand in hand--but we’ll teach you more next time.
-Emma and Nicole