Anatomy of HTML

by Devmountain Tutorials

☰ WHAT'S COVERED

This section will explore how to understand HTML by discussing:

1. REVIEW: HOW HTML IS USED IN WEB PAGES
2. HTML ELEMENTS AND TAGS
3. NESTING ELEMENTS
4. ATTRIBUTES
5. EMPTY ELEMENTS

---

1. REVIEW: HOW HTML IS USED IN WEB PAGES

In the previous section of this course, we talked a lot about the day to day for a web developer, the history of the internet and modern web application architecture. We ended by giving a brief look at CSS, HTML and JavaScript. In this unit, we will be expanding on all 3 of these in greater detail.

Let's start with HTML and review what we've learned so far.

HTML, or HyperText Markup Language, is markup language. Markup languages are used to group, annotate, and categorize content to describe the content's semantics in a way that computers can understand. Without HTML, the contents of a web page would look like this to a computer:

> **Plain Text**
>
> sophia-logo.png
>
> Menu Search
>
> Sign In
>
> Purchase a Sophia membership for just $79 per month.
>
> Choose a course to get started.
>
> Low-cost, self-paced courses for college credit in a fresh and fun learning environment.
>
> See how they work.
>
> woman-in-chair.jpg

By annotating and grouping parts of the page with HTML, computers are better able to understand how content is related. For example, HTML is used to indicate that certain parts of a webpage are part of a navigational menu containing hyperlinks or buttons used to navigate from page to page.

```
<nav>
  <img src="sophia-logo.png">
  <button>Menu</button>
  <button>Search</button>
  <a href="/sign-in">Sign In</a>
</nav>
<aside class="announcement">
  <p>
    Purchase a Sophia membership for just $79
    per month.
  </p>
</aside>
<main>
```

```
        <section id="about-courses">
          <div class="left">
            <p>
              Low-cost, self-paced courses for college credit in
              a fresh and fun learning environment.
            </p>
            <button>See how they work</button>
          </div>
          <div class="right">
            <img src="woman-in-chair.jpg">
          </div>
        </section>
      </main>
```

The example HTML above showcases many components of HTML syntax. We'll go over those components in more detail; then, you'll be able to understand exactly what's going on in the example HTML.

---

2. HTML ELEMENTS AND TAGS

HTML consists of a series of **elements** that enclose different parts of the content.

Elements are created with HTML tags. For example, this is a paragraph (p) element:


```
<p>The web is fun.</p>
```
To create a paragraph in HTML, start with the content of the paragraph.
The web is fun.

Then, enclose it in a **p** element by sandwiching the content in between an **opening tag**

```
<p>
```
and its corresponding **closing tag**.
```
</p>
```
.

So that it looks like this:


```
<p>The web is fun.</p>
```
Altogether, HTML elements are made of an opening tag, then content, then a closing tag.

This section of the course includes some embedded practice areas like the box below. You can practice following the instructions to see how the code changes the lines on the mini screen. The system does not record correct or incorrect answers, so feel free to experiment and practice with your new coding knowledge.

Edit the code in the **Editable Code** area in the box below. Follow the steps to mark that it is a strongly-worded phrase by wrapping it with the strong tags. This should make the text look bold. You'll see your changes update live, as you type, in the **Live Output** section. You can reset everything with the **Reset** button. Click on the **Show Solution** button to see the answer.

[✎] TRY IT

Follow these steps in the box below

1. Type the opening tag
```
<strong>
```
before the word **This**.

2. Close the element by typing
```
</strong>
```
after the word **phrase** .

3. Don't forget the **Reset** button if you need to start over.

4. Click the **Show Solution** button to see the answer. Hint: If your code matches the solution code, the **Editable Code** box will look like nothing has happened. You'll see the button name change to **Hide Solution** and you'll see the code applied in the Live output section right above.

Congratulations, you just wrote code! Next we'll talk about how to structure elements with nesting.

HTML element

The individual building blocks of HTML. You can create an HTML element with HTML tags. For example, a paragraph element consists of an opening tag, then the content of the paragraph, then its closing tag.

Opening tag

Opening tags mark the beginning of an HTML element. It consists of the name of the element (for example, p for paragraph), wrapped in angle brackets.

Closing tag

Closing tags mark the end of an element. They look the same as opening tags, except they have a forward slash before the name of the element.

3. NESTING ELEMENTS

Elements can be **nested** within other elements. When an HTML element exists inside another HTML element, it is said to be nested within the outer HTML element.

For example, if we wanted to emphasize that the web is *very* fun, we could wrap the word **very** in an emphasis (**em**) element to indicate that it should be emphasized.

<p>The web is very fun.</p>

Take care to open and close the tags of nested elements properly! Note how, in the example above, we open the **p** element first then open the **em** element, which means we should close the **em** element first before closing the **p** element.

If you don't open and close HTML elements properly, the browser will try and guess at what you're trying to do which may lead to unexpected results. So, it's important to have tags open and close in a way that they're either inside or outside one another.

Here's an example of *improperly* nested elements:

<p>The web is very fun.</p>

Properly nesting elements might not sound like a big deal, but it is important for organizing your code and being able to apply attributes. We'll talk about attributes next.

Nesting elements

When an HTML element exists inside another HTML element, it is said to be nested within the outer HTML element.

4. ATTRIBUTES

Elements can also have **attributes**. Attributes contain extra information about the element and don't appear in the content of the page.

For example, we can add a **class** attribute to our paragraph:

<p class="comment">The web is very fun.</p>

The class attribute gives elements an identifying name that can be used to select the element to apply style information to it in CSS or to add scripted behavior to it in JavaScript. In the example above, class is the name of the attribute and comment is its value.

Another element you'll see used often is the anchor element

<a/>

which can turn its content into a hyperlink. Anchors can have several attributes, but the most common ones used with them are:

Attribute Name	Description	Example
href	The URL of the link.	href="**https://sophia.org/**"
title	Extra information about the link, like a description of the page being linked to. This appears when a cursor hovers over the link as a tooltip.	title="The Sophia homepage"

✏️ TRY IT

Follow these steps in the box below

Turn the content in the *Editable Code* section into a link to the Sophia homepage. Again, you'll be able to see your changes appear in the *Live Output* section. The *Reset* button will discard your changes and you can view the solution with *Show Solution* button.

1. Type

<a

before the words **A link**.

2. Type

/a>

after the word **homepage**. This is called "wrapping" the element, with a tag at the beginning and at the end of the line.

3. Look at the example column in the table above for the attributes. Using the whole phrase including the =, add the **href** and **title** attributes within the

<a>

element.

4. You can always peek at the solution if you need a hint.

If you look at more examples of HTML code, you might see attribute values written without quotation marks:

A** link to the Sophia homepage.

The example above is completely valid, but it's still not a good idea to omit quotation marks because it can break your HTML in other circumstances. For example, if your attribute's value contains a space, leaving out quotation marks will break your markup:

A link to the Sophia homepage.

Even though you might see examples of HTML attributes without quotation marks, you should always include them. Doing so avoids unexpected problems and results in more readable code.

You might also see attributes written without values:

<input type="text" disabled>

The HTML above is shorthand for:

<input type="text" disabled="disabled">

Attributes like **disabled** are **Boolean attributes** because their value can be set to true or false. The **disabled** attribute is given to **input** elements in order to prevent the user from typing into them. Boolean attributes are set to true if the attribute is present in the element or if its value is the same as the attribute name:

<input type="text" disabled>

<input type="text" disabled="disabled">

They're set to false if they're omitted from the element:
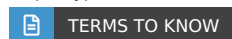
<input type="text">

📄 **TERMS TO KNOW**

Attributes
Attributes contain extra information about the element and don't appear in the content of the page.

Boolean attribute
A Boolean attribute is an attribute that can be set to true or false.

5. EMPTY ELEMENTS

Not all elements have an opening tag, content, and a closing tag. Some tags are empty elements and consist of a single tag.

Empty elements are typically used to insert or embed something — like an image — in the page. For example, the **img** element embeds an image file onto the page:

<img src="**https://sophialearning.s3.amazonaws.com/static/logo_sophia-d5c178b4e98abe275fb9aa947126eb27021850b2c7a8c10b710b5245e20a655f.svg**">

On the left you see the code, and on the right you see the image displayed.

This section has introduced you to many different parts of HTML. In the next section, we'll look at how to organize them to make a page.

📄 **TERMS TO KNOW**

Empty elements
Empty elements, also known as void elements, consist of a single tag.

📄 **TERMS TO KNOW**

Attributes
Attributes contain extra information about the element and don't appear in the content of the page.

Boolean attribute

A Boolean attribute is an attribute that can be set to true or false.

Closing tag

Closing tags mark the end of an element. They look the same as opening tags, except they have a forward slash before the name of the element.

Empty elements

Empty elements, also known as void elements, consist of a single tag.

HTML element

The individual building blocks of HTML. You can create an HTML element with HTML tags. For example, a paragraph element consists of an opening tag, then the content of the paragraph, then its closing tag.

Nesting elements

When an HTML element exists inside another HTML element, it is said to be nested within the outer HTML element.

Opening tag

Opening tags mark the beginning of an HTML element. It consists of the name of the element (for example, p for paragraph), wrapped in angle brackets.