

Web Development: Lesson 1

Discord Lessons

Fall 2020

Doomer / ドウーマ

Lesson 1 - The Hypertext Markup Language

dedicated to Spennorex and Grumpah

1 Introduction

In this lesson we'll learn the most important terms of web development, which might be used later on to explain more complex concepts and build a basic web page. This guide is meant to be for complete beginners and can therefore be skipped by people, who already feel comfortable with HTML. As a side note: if you want to follow along this script, you'll see passages where code snippets are included. In order to get those running you need to create a text file with your preferred editor and paste / write the code inside the file, after having done that you need to change the file extension from ".txt" to ".html" and then you can simply save and double-click to open it in your browser. Don't be worried if something is not working properly, feel free to drop me a message (Doomer#1718).

2 Document Structure

In order to create a web page, we need to understand the language web documents are written in - **HTML** (Hyper Text Markup Language). This language is used to describe the **DOM** (Document Object Model) of a web page. To get a better grasp of the concept, here's an illustration:

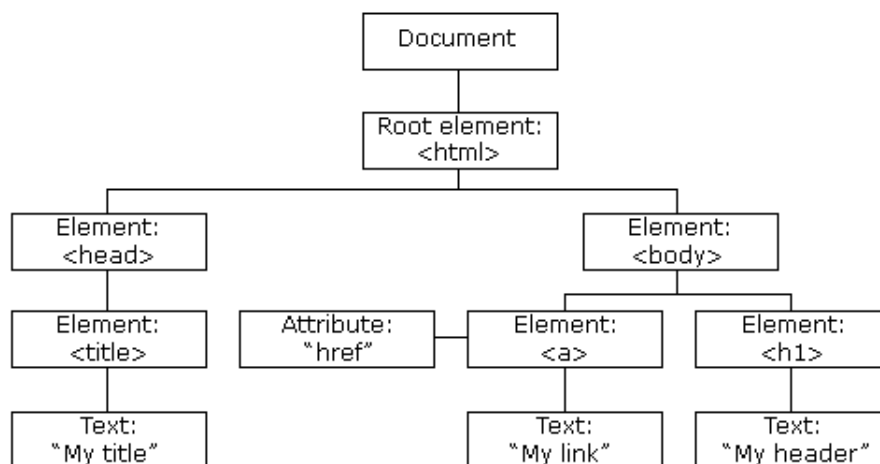


Figure 1: HTML DOM

Source: https://www.w3schools.com/js/pic_htmltree.gif

It might be confusing in the beginning, but that's how a web page looks like for a browser. This particular example, would look like this, if we'd code it:

```
1 <html>
2   <head>
3     <title>My title</title>
4   </head>
5   <body>
6     <a href>My link</a>
7     <h1>My header</h1>
8   </body>
9 </html>
```

Listing 1: DOM Example Page

If you look at this snippet, you might see some interesting things. There are words inside brackets, also called **tags**. The one without the `'/'` is called **start tag** and the one with `'/'` is the **end tag**. These are used to delimit an **element**.

Everything inside those tags is considered to be an html element, additionally one should add as seen in the example, that an element can contain other elements. In Listing 1 you can see, that the "html" element contains the "head" and the "body" elements, which themselves contain the "title" and the "a" and "h1" elements.

Now that we have covered what elements are, we can look into learning a few html-elements, to get a basic "vocabulary".

3 HTML Elements

You've already seen following elements:

- ◇ `<html>`
- ◇ `<head>`
- ◇ `<body>`
- ◇ `<title>`
- ◇ `<a>`
- ◇ `<h1>`

Let's first discuss the first three elements. Those three are called the **Document structure elements** - they organize the huge document into compartments.

The `<html>` element is the root element and contains every other element, that's why it's common to see a `<html>` at the beginning of an html-file and a `</html>` at the end of it. The `<head>` element is the meta information element and contains information you don't see on the page itself, such as the title.

The `<body>` element is the place where the actual content belongs to - the text that is displayed on the web page.

Next we have the elements that can be used inside the `<head>`, those are called **Document head elements**, these include:

The `<title>` element is used to define the page title and is what you usually the browser tab name.

There are a few other important ones, which we'll cover as soon as we need them.

The contents of `<body>` are called the **Document body elements** and manage the content of the page. We'll start of with the simplest element.

The `<p>` element is used to define a paragraph of text. This is where you could put a

simple text. If you want to have a new line (line break) inside the text (applies to any html element) you need to insert this **self-closing** tag `
`. If you see a tag, which has the `'/'` at the end it means that it is *self-closing*, thus you don't have to have 2 tags as usual - one for the opening, one for the closing, but one.

The familiar `<h1>` is a heading element and the 1 means that it is a top layer heading, which makes the text huge, in html you can have up to 6 levels of headings - from `<h1>` to `<h6>`, where an increasing number means a decreasing font-size.

Another common html elements are the list elements `` for unordered lists and `` for ordered lists, where the unordered is using bullet points for its items and the ordered arabic numerals (per default). Both lists contain items, which are defined by the `` list-item tag.

The last body tag for now will be the ancor tag `<a>`, which is also shown above - it allows us to insert a link to another web page or our own subpage, we'll look into this later, as soon as we'll start using multiple html files together.

4 Building our Web Page

So now that we've covered quite a bit of theory, we can start building a very simple *static* webpage, which will improve a lot over time. To start off we can use the standard wireframe of web pages:

```
1 <html>
2   <head>
3     <title></title>
4   </head>
5   <body>
6   </body>
7 </html>
```

Listing 2: Our HTML Wireframe

Now it's time to pack some stuff into our skeleton.

```
1 <html>
2   <head>
3     <title>My Profile</title>
4   </head>
5   <body>
6     <h1>My Profile Page</h1>
7     <p>Here's who I am :)</p>
8     <ol>
9       <li>Name: Doomer</li>
10      <li>How many Discord Notifications do you have? 0</li>
11      <li>Gender: genderfluid</li>
12      <li>Birthday: 22nd of December</li>
13      <li>Timezone: UTC+2 (CET)</li>
14      <li>Languages Spoken: English, Russian, German</li>
15      <li>Favorite Colors: Black, Blue, Purple</li>
16      <li>...</li>
17    </ol>
18    <p>I think that's enough <br/> for an example.</p>
19  </body>
20 </html>
```

Listing 3: Content Update

I hope that from that example you could understand the basic structure of an html document and therefore can do this yourself. I think I'll end this theoretical part of the script with this, since we covered a lot of different elements and it's probably going to take a while to get everything into your mind. The next update would be covering styling and making the web page look beautiful, which is a huge update, thus I decided to pack it into the next script.

5 Exercises

These exercises are optional and you can do them if you want to get experience and feel comfortable using the basic HTML syntax:

1. Create a profile page for yourself.
2. Create a page with your favorite song lyrics.
3. Create a site with movies / books / anime you watched, with a small description for each

6 Final Notes

If any questions occur, you can always ask me (it might take some time, since we are not necessarily in the same timezone), but you'll get an answer. If you can make the exercises without looking back to my templates, then it'd be great, but it's definitely not wrong or bad to use a reference - to be honest, most developers nowadays use a *ton* of copied code, but it's useful to have a foundation to build on. If you're disappointed, that everything was very easy or difficult, give me some feedback! I haven't created scripts that often and mostly have done it for myself, so I have no idea if what I've done is great, ok or absolute garbage, so please if you have anything to criticize - do it! I should add as well that I'm not too good in english, since it's not my mothertongue, so if you see me use some weird formulation / wrong order or literally any other possible mistake, tell me, so I can make this more easy and comfortable to read! Thanks a lot for reading this far :). See you in the next script.