Introduction

HTML5

What is a Website?

A website is a collection of HTML content or documents (web pages) on a server.

Servers are typically specialized computers, set up with hardware and software designed specifically at delivering this content to users.

Public servers often opt for a domain (think ".com") to make accessing their website(s) easier for potential visitors to type and remember.

Most users make use of formal client software called web browsers to request and view web pages that make up websites. Popular web browsers include:

- Google Chrome
- Apple Safari
- Microsoft Edge
- Mozilla Firefox
- Opera
- Vivaldi

What is the World Wide Web made of?

When you visit a website in your web browser, you are met with a web page featuring a layout boasting text, imagery, forms, video, and potentially a host of other content and features.

Every one of these web pages, from the most basic-looking, to the most interactive and exciting, must be built out of 1 to 3 of the languages of the web. The one that must be there in every page, no matter what, is HTML.

HTML (Hypertext Markup Language) tells your browser what the text content is, and where to download media content like images or videos.

The other languages your browser understands include CSS (Cascading Stylesheets) and JS (JavaScript.)

CSS determines the look and layout of the website—everything from the font, to the colours, to the shape and size of the elements throughout a page.

JS allows for dynamic and interactive features in a web page. As an example, an online calculator would require JavaScript in order to calculate a result. HTML and CSS are not meant to think, only to describe content and appearance, respectively.

Languages of the World Wide Web

HyperText Markup Language (HTML)

The skeleton of the web -site. This is where the content lives, and where we can mark it up to give it meaning.

Cascading StyleSheets (CSS)

The skin or visual aspect of the website. Controls the layout, spacing, position, and sizing of elements in a web-page.

JavaScript (JS)

The brain of the website. Any interactive or dynamic features on a web-page are the responsibility of JavaScript.

HTML5

HTML is a programming language that we use to configure and populate web pages with content and media.

The latest stable release is version 5, therefore modern HTML is referred to as HTML5. It is specifically this version of the HTML language that we will focus on throughout this module.

HTML5 is purposefully largely—if not entirely—backwards-compatible with previous editions in order to ensure older websites, elements, and practices are able to load appropriately in new browsers. Though old and outdated practices may still function it is important to implement only modern features and practices.

Modern browsers and accessibility tools rely on good semantic HTML5 code in order to operate properly. Search engines also take adherence to HTML5 seriously when ranking their results—it is in their best interest to serve quality websites that are regularly updated and utilizing current best practices.

HTML Elements

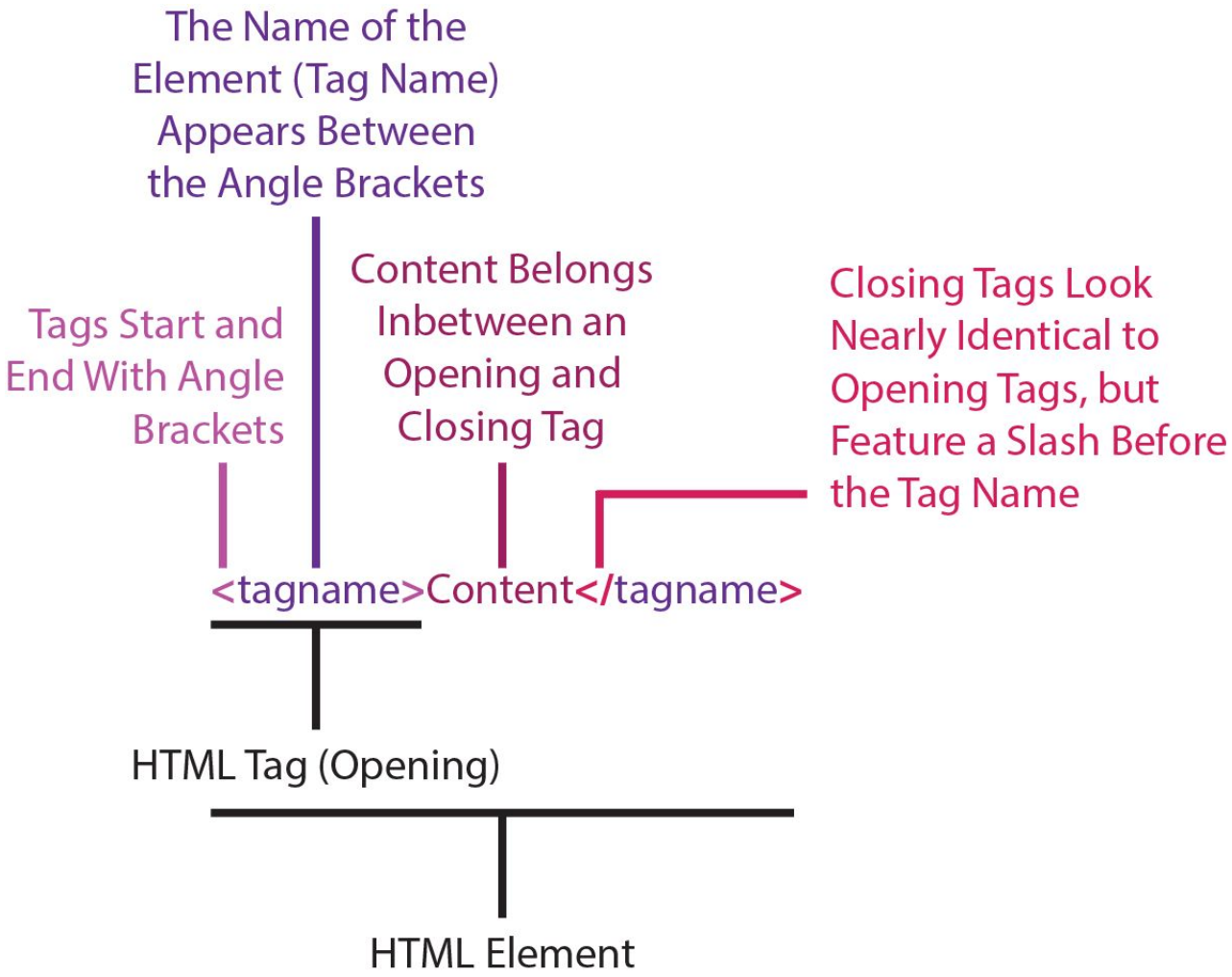
HTML documents (web pages) are made up of elements. There are some elements that are required for every page, but most we get to decide when to use. Elements help us describe content. They might look something like…

Hello there, and welcome to my website!

The “p” element tells the web browser, search engines (like Google and Bing), and accessibility tools (like screen readers) that its contents are a paragraph. This is one of the most common elements you’ll see on the web!

Most elements of a web page are composed of three basic pieces…

1. An “opening” tag:
2. Content (if applicable): Hello there, and welcome to my website!
3. A “closing” tag:


The Name of the Element (Tag Name) Appears Between the Angle Brackets
Tags Start and End With Angle Brackets
Content Belongs Inbetween an Opening and Closing Tag
Closing Tags Look Nearly Identical to Opening Tags, but Feature a Slash Before the Tag Name
<tagname>Content</tagname>
HTML Tag (Opening)
HTML Element

Essential Parts of a Webpage

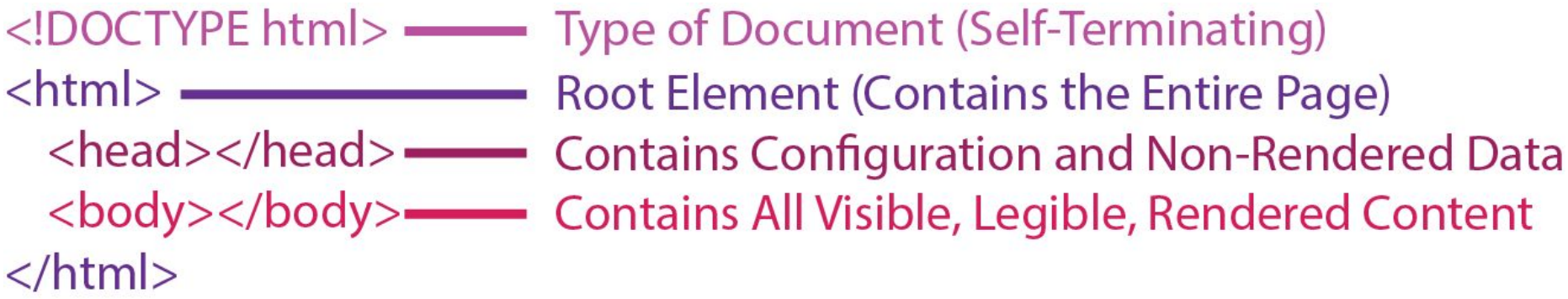
There are some elements that every single web page should have—this is non-negotiable for following the standards of the web, especially if you want to be HTML5-compliant in your code.

1. <!DOCTYPE html>
   This is a document type declaration, letting the client program know that this file is, in fact, an HTML5+ web page. Note that it is self-closing, and does not have a matching closing tag.
2. <html></html>
   The root element of the page. All elements aside from a document type declaration must be nested inside.
3. <head></head>
   The head element is intended to rest immediately inside of the root element. Head is used for containing descriptive and configuration elements/data. Its contents are not visible in the rendered web page.
4. <body></body>
   The body element is intended to rest immediately inside of the root element, following the head element. All visible content within a page must reside inside of the body element—this means all text, images, video, forms, etc. that you see are inside of this.

Each of these core elements are absolutely necessary for a web page to be considered valid—you'll be seeing a lot of them as we proceed!

This can be considered: the anatomy of an HTML document.

Depending on the type of element, elements can be nested—meaning you can put one inside of another.

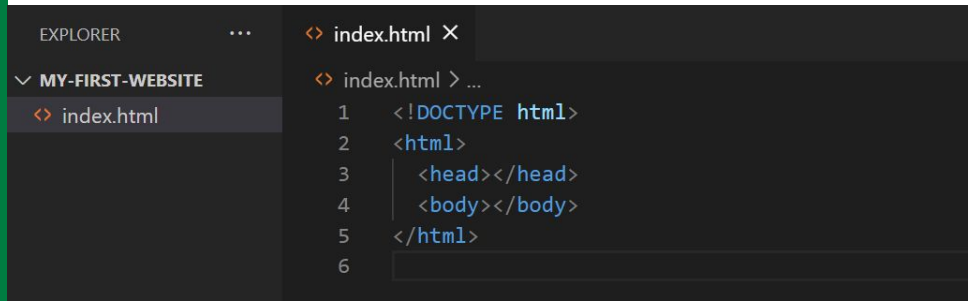

<!DOCTYPE html>
Type of Document (Self-Terminating)
<html>
Root Element (Contains the Entire Page)
<head></head>
Contains Configuration and Non-Rendered Data
<body></body>
Contains All Visible, Legible, Rendered Content
</html>

Your First Web Page

Create a folder for a little test project we can experiment with together.

1. Create a folder, name it based on what your project is about so that you know what's in it later.
2. Create a file called index.html inside of that folder, we'll be writing our HTML code inside of this file.
3. Type out the important elements we have gone over, into the HTML file.

The name "index.html" is usually used for the homepage in a website. This is because most server software is configured to serve a file by that name as the default, if the browser didn't specify.



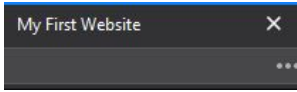
Title Element

The title element is responsible for the page name that shows in your browser tab and bookmarks.

A page should only contain one title element. Its content describes what the web page is about, and should make sense as a tab title, search result heading, or a bookmark.

The title element must be in between the opening and closing head element tags.

Try opening the page in your browser.



index.html

index.html > ...

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>My First Website</title>
5    </head>
6    <body></body>
7  </html>
8
```

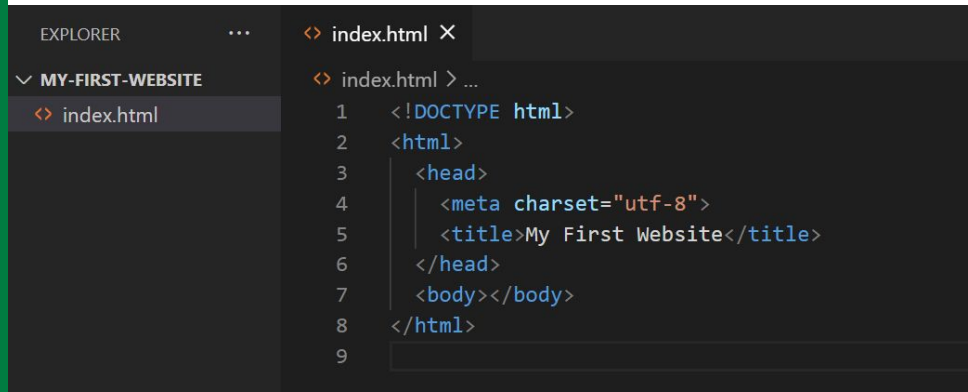
Meta Element

Meta elements capture additional information about the page and sometimes offers guidance to the browser on how to render it.

To prevent issues in how your browser interprets your HTML code, it is important to specify the character set you'll be using in your page. Most pages will do just fine with UTF-8, which includes a wide range of the most common characters and emojis.

We specify the character set via a meta element, but we can use more elements to specify more from a configuration perspective as well.

Have a look at what this looks like...



Element Attributes

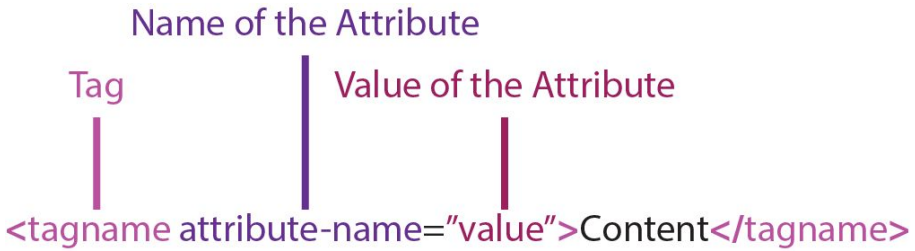
That meta element looked a little different than what we're used to. There are two main differences to make note of, let's break it down:

- The meta element is self-terminating, meaning that there is no place to enter "content"; there is no closing tag.
- Data that is not considered the output content is stored in element attributes. In the last slide, we saw the "charset" attribute.

Attributes can be added to any element, but it is important to research which ones are valid or will be beneficial on an element by element basis.

Both standard and self-terminating elements can have attributes, and an attribute can at times even modify the browser's behaviour of an element in the page. We'll see this more when covering web forms.

An element can have multiple attributes. Each name-value pair can be separated by a space within the tag (after the name, but inside the tag's angle brackets.)



Multiple Attributes

Most meta elements expect a “name” attribute, and a “content” attribute.

This next meta value is very popular as a responsive (mobile-friendly) consideration. The viewport meta below lets the browser know to scale the page based on the screen-size, which is preferred for a good user experience on phones.

The element also acts as a great real example of multiple attributes being set in a tag. All attributes for the element belong inside of the angle brackets.

Any web page you build from now on, should have (at least) these fundamental elements present!

index.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My First Website</title>
  </head>
  <body></body>
</html>
```

Adding Content to your Page

Now that the page is all set up, we can begin adding content!

Try adding an H1 element (this is the main heading, describing the topic of the whole page) and a paragraph (P) element with some text. Open the page in your browser (or refresh if it was already open) to check the result. That's a website!

index.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My First Website</title>
  </head>
  <body>
    <h1>My First Website</h1>
    <p>
      Hello, and welcome to my very first website! I am practicing the
      HTML language.
    </p>
  </body>
</html>
```

Congratulations on your very first website!

Recommended Reading

To explore further, we recommend cracking open the following:

- Pilgrim, M. (2010). *HTML5: Up and Running*. O'Reilly Media Inc.