Let's look at some basics you should follow when creating an html page from beginning to finish. When you begin an html file you should always follow a few simple naming rules. It is strongly suggested that you save your file names in all lower case. This will cut down on mistakes that might be made later. The first character of the file name should be a letter, don't start the file name with a number. And there shouldn't be spaces in the file name. If you want to separate words you can use a dash or an underscore but no other special characters. These rules apply to any folders, image file or other resource file names you create within your website as well.

Remember the file with the name index.html will always be recognized as the home page of your website. So, when the user types in the domain name of your website like amazon.com the index.html page will always come up first as the home page. Other sub-pages or child pages on your website don't have to have a special name, you can name them whatever you want but try and remember to keep it short and simple and follow the naming conventions.

I'm going to use VSCode shortcut here and type an exclamation mark and tab or enter. This brings in the beginning of our html page. Notice the meta tags that come in. Let's talk about those. HTML lets you add additional information about your document using meta tags. The meta element can include name/value pairs describing properties of your HTML document. These tags are not visible on your web page but belong in the head section of your HTML. This is where we have information about our page, not the content that shows up on our page.

I could add my own stating who the author of the document is. It is not required, and it does not affect the look of the document in any way but is just additional information about our page if we want to include it.

The next meta tag tells us what character set we are using in our document. UTF-8 refers to the Unicode character sets. Unicode can handle any international language characters so it's a good one to use on your web page.

The next meta tag tells MS to use the latest rendering engine (Edge). This is only needed in a few instances. It's also not always needed.

This last meta tag is important, and I would always include it on your html documents. It allows the developer to take control over the viewport (or visible area of a web page) no matter what device is being used. Whether the user is seeing your webpage on a phone, tablet, laptop, or desktop. The 'width=device-width' sets the width of the page to follow the screen-width of the device and the initial-scale=1.0 sets the initial zoom level when the page is first loaded in the browser. So, this initial zoom is at 1.0 or 100%. When used in conjunction with media queries which we will touch on later, it can help make pages be responsive or in other words, look good on any device.

As you are creating your file, make sure you are using semantic elements for your content. Most tags describe what is inside them and are semantic for example the nav tag has a

navigation contained between its opening and closing tag. But some elements are not semantic like div or span tags. These tags tell nothing about their content. They simply create a wrapping element around their content usually for CSS presentation purposes, not because it gives their elements meaning.

So, for example I might not have a good tag that has a semantic fit with the banner information, but I know that I will be doing CSS on all that content together later so I place a div with the class of banner-msg around it.

Also, the span element is an example of a non-semantic tag. I might want the color of just the text 'have an outdoor heat source' to be a different color with CSS but it doesn't really have a semantic meaning to that content.

So why use a nav tag when a div tag would work and your page's navigation or menu would look just the same on your page? The reason it's important is because screen readers and search engines use this extra information and it's important to give our pages as much meaningful information about our content as we can. This makes it accessible to those navigating our page without sight and also for search engines like Google to make more sense out of our page and rank it appropriately. It's also just good practice to do it.

Once we are done with our file, we should validate the code. Validating our HTML and CSS is important. Validating is just like it sounds—making sure our code is valid or coded properly. W3C is the World Wide Web Consortium or international community that maintains and develops web standards. They have a validator that you should use to validate your HTML. It is at validator.w3.org.

Here you can check your web page html code by either placing the web address of your page, uploading your html file, or by pasting your html code in directly. Then when you check it, it will then let you know if you have any errors. Then you should correct any errors so your page will validate with no errors.

Here's what it would look like once it validates. Why is this important? First it can help you debug your code. If your page is not rendering like you expected it to, validating can show you where your errors are. Also, it helps you learn good practices, have clean code, and help with site performance. It is a sign of professionalism and those hiring you later might look at your portfolio or web pages and validate them to see if you have good code. Also search engines like Google will rank your website much better if your code validates. So, get into a good habit of always validating your HTML code. You can also validate CSS at jigsaw.w3.org/css-validator