Multiple Page Website

First, just like you did with your first web page, you need to create a new file for your new page. Let’s say you want to create a Work page to show your projects. So you can call your new file work.html, or whatever makes sense to you. Then, you can go ahead and code it up just like you’re used to with HTML & CSS.

Note that you should link to the same stylesheet that you used for your main page in your new work page. Go to your index.html file and copy the links to your normalize.css and main.css stylesheets, along with links to any fonts you’ve chosen. Then, paste these links in your work.html file in the head area, beneath the title.

So how do you structure these files? There are actually a couple of different ways to do it, but we, of course, are going to show you the BEST way.

Now, since you’ve decided to add a Work page to your site, the next step is to go to your root directory – the same place your index.html file lives – and create a new folder called “work.” Then, move the work.html file you created into your “work” folder and rename the file index.html. Make sure your folders and file names are all in lowercase.

Later, once your site is live on your web server, this folder structure will allow you to use a link that looks like this:

www.mysite.com/work

Next, drag your new index.html file into your web browser. Wait a second! Where are all my styles?

So there’s one thing about using the new folder that you need to know: your CSS links are telling your new index.html file to look for the stylesheets in the wrong place! So you need to make one small tweak to your CSS links in the page head to help it find the right files.

In the code you just copied from the main index.html file, your stylesheet links are written like this:

<link rel=”stylesheet” href=”css/normalize.css”>

<link rel=”stylesheet” href=”css/main.css”>

These are relative links, which tell your computer to look for the stylesheets in a CSS folder that lives in the SAME FOLDER as the file you’re typing in. Since we copied this code from the main index.html file, this link is telling your computer to look for the CSS folder in the root directory. However, your NEW index.html file is not in the root directory – it’s in the work folder!

So to make sure your new index.html file can find the existing CSS stylesheet, you need to tell it to go back one step to the root directory to look for the folder called CSS. To do that, you add two dots and a forward slash to the beginning of the href attribute, like this:

<link rel=”stylesheet” href=”../css/normalize.css”>

<link rel="stylesheet" href="../css/main.css">

So now instead of looking in the work folder to find the stylesheets, it knows to back out to the root directory first, THEN head over to the CSS folder to find the right files.

Okay, so now that you’ve created your Work page and given it CSS styles it, it’s time for you to create a navigation menu and add it to your main page, and link it to your new page.

Let’s say your navigation menu will point to your new Work page, in addition to your Home page.

First, head over to your main index.html file and add in Home and Work navigation links in the header area, like this:

<nav>

<ul>

<li><a href="index.html">Home</a></li>

<li><a href="work/index.html">Work</a></li>

</ul>

</nav>

And don’t forget to add some styling in your CSS, like you learned in 101, to make it look more like a nav bar!

ul {

list-style-type:none;

margin:0;

padding:0;

}

li {

display:inline-block;

margin-right: 1px;

}

Now, let’s go back to the navigation link to your new Work page. On the web server, you’ll be able to find your new page by just writing “work.” But, for it to work on your local computer, you’ll need to write out the name of the file, which is “index.html.” So the FULL link needs to be “work/index.html” to see it in your browser.

And that’s it! You’ve just added a brand new page.

To add more pages, you can rinse and repeat until you have a website full of pages!