**Create A Portfolio Website: Part One -** [Eric Hu](https://medium.com/@huericnan?source=post_page-----be03c25214cf--------------------------------) Feb 4, 2022

In this article, we are going to use what we’ve learned about HTML so far to create a portfolio, which you can use to tell the world about you and your projects. Without further ado, let’s get started.

*You can download the source code for this tutorial here:*

[*Download the Source Code*](https://www.ericsdevblog.com/index.php/beginners-roadmap-to-web-development/)

First, create a work folder. This is what I did:

Graphical user interface, text, application

Description automatically generated

Everything related to the portfolio is inside the Portfolio directory. You can do this differently, but remember to be organized because we are going to add quite a few things in the future.

Next, go to Portfolio and create an HTML file. Let's call it "portfolio.html". Open it with a text editor, you can use the notepad that comes with your operating system if you want, but I recommend using a proper code editor like VS Code.

As we talked about before, an HTML file has the following format:

<!DOCTYPE html>  
<html><head>  
</head><body>  
</body></html>

To open this HTML file, you have two choices. If you are using VS Code, you could either use an extension called “Live Server”, which will create a server at the directory Portfolio.

Graphical user interface, text, application

Description automatically generated

Or, you can simply double-click on the file and open it with a browser.

**The**<head>**Section**

<head>  
 <meta charset="utf-8" />  
 <meta name="description" content="My Portfolio">  
 <meta name="keywords" content="HTML, CSS, JavaScript">  
 <meta name="author" content="Eric Hu">  
 <title>My Portfolio</title>  
</head>

This part is pretty straightforward. We first defined the character set that this file is going to use, this information is used by the browser to make sure the characters and symbols are displayed correctly.

Next, we defined the description, the keywords, and the author of this web page. This information is not displayed, but they are important for SEO (Search Engine Optimization) purposes.

Finally, the title of this web page is defined as well.

**The**<body>**Section**

**Header and Navigation Bar**

<!--Header and Navigation Bar-->  
<header>  
 <h1>Eric's Portfolio</h1>  
</header>  
<nav>  
 <ul>  
 <li><a href="/">Root</a></li>  
 <li><a href="portfolio.html">Home1</a></li>  
 <li><a href="/portfolio.html">Home2</a></li>  
 <li><a href="<https://www.ericsdevblog.com>">Blog</a></li>  
 <li><a href="#project">Projects</a></li>  
 </ul>  
</nav>

We can see that from line 7 to line 11, we defined five different links. The first three are relative URLs, the fourth one is an absolute URL that you should already be familiar with, and the last one is an anchor link, which we’ll talk about later.

For the first link, “Root” (/), it will take you to the root of the server/disk. So, if you are using the Live Server, clicking this link will take you to the Portfolio folder, which is the root of the server.

Graphical user interface, application

Description automatically generated

And if you are not, this link will take you to the root of the disk where this HTML file is stored, which, in my case, is disk D.

Graphical user interface, text, application, email

Description automatically generated

For the second link “Home1” (portfolio.html), the browser will look for a file named "portfolio.html" in the current directory. This will lead to problems because as your website gets bigger, it is possible that you don't know which directory you are currently in.

For the third link “Home2” (/portfolio.html), it is basically the combination of the first two. The browser will always go back to the root directory and then look for the portfolio.html file. This eliminates the possibility of any unexpected errors.

**Self Introduction Section**

<!--Self Introduction Section-->  
<section>  
 <h2>Hello, I am Eric. Welcome to My Portfolio.</h2>  
 <p>...</p>  
</section>

**Newsletter Signup Section**

<!--Newsletter Signup Section-->  
<section>  
 <h2>Would you like to see more tips and tutorials on web development?</h2>  
 <form>  
 <label for="firstname">First Name:</label>  
 <input type="text" id="firstname">  
 <br><br>  
 <label for="lastname">Last Name:</label>  
 <input type="text" id="lastname">  
 <br><br>  
 <label for="email">Enter your email:</label>  
 <input type="email" id="email" name="email">  
 </form>  
</section>

**Skills Section**

<!--Skills Section-->  
<section>  
 <h2>My Skills</h2>  
 <ul>  
 <li>HTML (100%)</li>  
 <li>CSS (90%)</li>  
 <li>JavaScript (90%)</li>  
 <li>PHP (100&#37;)</li>  
 <li>Java (90&#37;)</li>  
 <li>Vue.js (80&percnt;)</li>  
 </ul>  
</section>

Notice that I used three different methods to represent the percentage sign (%), and they should all give you the same result.

**Projects Section**

<!--Projects Section-->  
<section id="project">  
 <h2>My Projects</h2>  
 <div>  
 <h3>First Project</h3>  
 <img src="/Images/first project.jpg" width="500" height="300">  
 <p>...</p>  
 </div> <div>  
 <h3>Second Project</h3>  
 <img src="/Images/second project.jpeg" width="500" height="300">  
 <p>...</p>  
 </div> <div>  
 <h3>Third Project</h3>  
 <img src="/Images/third project.jpeg" width="500" height="300">  
 <p>...</p>  
 </div>  
</section>

Notice that I give this <section> element an ID attribute named "project". This is to work with the anchor link that we mentioned before.

<a href="#project">Projects</a>

This link will send you to the element whose ID is project. Also, note that you should always use the ID to specify which element you wish to anchor so that it is unique in the entire web page.

As for the images in this section, they are all stored in a folder called Images under the folder Portfolio.

Text

Description automatically generated

**Footer**

<!--Footer-->  
<footer>  
 <p>Created by Eric Hu</p>  
 <p><a href="mailto:huericnan@gmail.com">huericnan@gmail.com</a></p>  
</footer>

Graphical user interface

Description automatically generated

This is what the final result looks like. However, we now have a new problem, it looks terrible, it is not going to land you any job looking like this. But don’t worry, after studying CSS in the next part of this tutorial, we’ll find a way to make this web page more appealing.

11