What is the different between absolute and relative paths?

A path is a string that specifies the location of a file or directory in a file system. In web development, paths let developers link to resources like images, stylesheets, scripts, and other web pages. There are absolute and relative paths - both are essential when specifying file locations within a file system. Let's look at the two so you can decide which one of them to use and when.

An absolute path is a complete link to a resource. It starts from the root directory, includes every other directory, and finally the filename and extension. The "root directory" refers to the top-level directory or folder in a hierarchy.

An absolute path also includes the protocol - which could be http, https, and file and the domain name if the resource is on the web. Here's an example of an absolute path that links to the freeCodeCamp logo:

<a href="https://design-style-guide.freecodecamp.org/img/fcc\_secondary\_small.svg">

View fCC Logo

</a>

In this example, the protocol is https, the domain name is design-style-guide.freecodecamp.org, and the filename is fcc\_secondary\_small.svg.

Now, what if the resource you want to link to using an absolute path is on your local machine? Here's how to link to the about.html file with an absolute path:

<p>

Read more on the

<a

href="/Users/user/Desktop/fCC/script-code/absolute-vs-relative-paths/pages/about.html"

>About Page</a

>

</p>

It looks like this because we are going into a folder called Users, then into a folder called user, then into a folder called Desktop, then into a folder called fCC, then into a folder called script-code, then into a folder called absolute-vs-relative-paths, then into a folder called pages to finally get the about.html file.

Here's what the absolute URL looks like in the browser address bar:

file:///Users/user/Desktop/fCC/script-code/absolute-vs-relative-paths/pages/about.html

The URL includes the protocol, file://. It also include the path, which looks like this: /Users/user/Desktop/fCC/script-code/absolute-vs-relative-paths/pages/, and represents the series of folders that lead to the file. And finally, it also includes the about.html, which is the filename and the extension.

Now, lets look at the relative path. A relative path specifies the location of a file relative to the directory of the current file. It does not include the protocol or the domain name, making it shorter and more flexible for internal links within the same website. Here's an example of linking to the about.html page from the contact.html page, both of which are in the same folder:

<p>

Read more on the

<a href="about.html">About Page</a>

</p>

So imagine you are on the contact.html page, and because the about.html page is in the same place, you simply get the filename. This is an example of using a relative file path.

So, which should you use and when; an absolute path or a relative path? Here are the rules you should follow:

* Use absolute paths when linking to a resource hosted on an external website.
* Use absolute paths when you need the link to a page or resource to work consistently regardless of the document location within the site.
* Use relative paths when linking to resources within the same website.
* Use relative paths if you want to keep your code cleaner and easier to maintain during development.
* Use relative paths during local testing to ensure links work without an internet connection.