

COMP 1017

Day 08

So far, we've been creating
single page websites.

But single-page sites are
not really the norm.

So, how can we create multiple-page websites with functional navigation between pages?

Absolute vs. Relative Pathing

So far, we've been using `<a>` to link to **external** websites.

The href has always been a URL to somewhere else — that is, on some other site, hosted on some other server.

These URLs are an example of what we'd call **absolute** paths.

Think of it like a street address.
Absolute paths do not change.

ex. The HP Centre is located at
10504 Princess Elizabeth Ave NW.

However, we've also been using
relative paths for other things.

Whenever we use an ``, our `src` value is *relative* to where the `index.html` is located.

Our images are inside of our image folders, so we have to tell our browser to go there in order to get the image file it needs.

This is more like giving directions.

If we were in the CAT Building and you wanted to get to class, giving you an absolute address wouldn't be very helpful.

(We would already be on campus, after all.)

Instead, I'd tell you how to get there
starting from where we currently are.

This is exactly how linking within the same framework folder (i.e. the same website) works.

Linking in the Same Directory

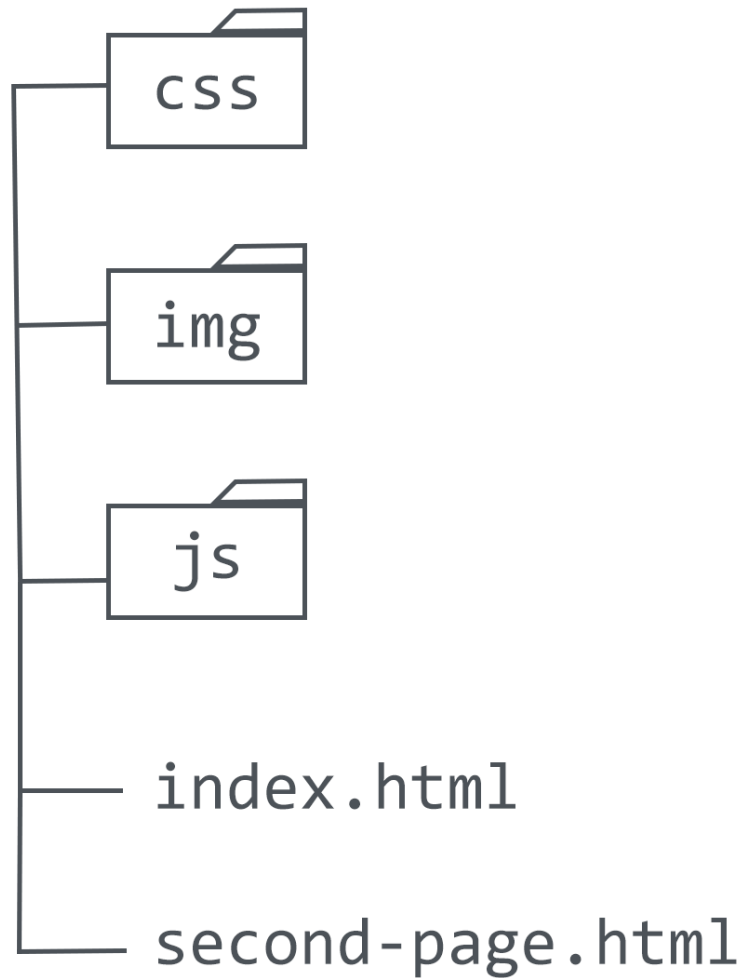
Let's say we've got a bunch of pages for our website, all in the **root** of our framework folder.

Note: When I say root, I mean *immediately inside* of the folder.

Instead of giving an **absolute** path or address, we will give a **relative** path.

So, let's say our framework folder looks like the following ...

framework



How can we give 'directions' from
`index.html` to `second-page.html`?

```
<a href="second-page.html">  
  Link to Second Page  
</a>
```


... that's literally it. If your other pages are in the same folder, you just have to point to them.

In this case, you do not have to tell your browser to go somewhere else in order to find the files.