CS1115/CS5002

Web Development 1

Dr Derek Bridge

School of Computer Science & Information Technology
University College Cork

Background: Unix file systems

- A **pathname** is a list of pathnames you must 'travel' through to get to where you want to go (separated by /)
- An **absolute pathname** is one that begins at the root
  - Starts with /, the root directory
- A **relative pathname** is one that begins in the current working directory
  - Does not start with /
  - To go up a level, use `..`

Relative pathnames

- Examples, assuming the current working directory is `fish`
  - The relative pathname for the file called `siamese.html` is `../cats/siamese.html`
  - The relative pathname for the directory called `users` is `../../../../users`
- **Exercise**: assuming the current working directory is `html`, write down the relative pathname for
  - the file called `index.html` (the one in `html`)
  - the directory called `CS1106`

Absolute URLs and relative URLs

- Absolute URLs use absolute pathnames
  - But start at *document root*, not filesystem root
- Relative URLs use relative pathnames
  - Starting at the directory that contains the current document
  - The browser converts the relative URL into an absolute URL before sending a request to the server

Using absolute and relative URLs

	Absolute URL	Relative URL
External link (i.e. on a different server)	scheme + hostname/IP address + absolute pathname	N/A
Internal link (i.e. on the same server)	scheme + hostname/IP address + absolute pathname	relative pathname
	absolute pathname	

External link

- E.g. you want to put an `<a>` element into `guppies.html` that links to the RTE news site
- Since you are linking to a page on another server, you must use an *absolute URL* and you must include:
  - the scheme, e.g. `https`
  - the hostname (or IP address), e.g. `www.rte.ie`
  - an absolute pathname, e.g. `/news/index.html`
- E.g. `<a href="https://www.rte.ie/news/index.html">RTE News</a>`
- These are also the URLs that you can see in (or type into) the Location box in your browser (but there you can omit the scheme)

Internal link

- E.g. you want to put an `<a>` element into `guppies.html` that links to `siamese.html`
- You choose between an absolute URL and a relative URL
- E.g. absolute:
  - `<a href="http://www.example.com/cats/siamese.html">Click me</a>`
  - `<a href="/cats/siamese.html">Click me</a>`
  - N.B. The forward slash at the start
  - **Question**: You can include the scheme + hostname (or IP address). But why is it better to omit them?
- E.g. relative:
  - `<a href="../cats/siamese.html">Click me</a>`
  - N.B. No forward slash at the start; no scheme, no host
- **Question**: For internal links, we have a choice. How would you choose?

Class exercise

- You are editing `siamese.html`
- Using *absolute URLs*, write hyperlinks to
  1. `persian.html`
  2. `guppies.html`
- Repeat, this time using *relative URLs*

User web sites

- An absolute URL looks like this:
  `<a href="/~benny/hobbies.html">Click me!</a>`
  or this:
  `<a href="http://www.example.com/~benny/hobbies.html">Click me!</a>`
- Relative URLs are unaffected
  - E.g. linking from benny's `index.html` to his `hobbies.html`:
    `<a href="hobbies.html">Click me!</a>`
  - E.g. including betty's photo in her `index.html`:
    `<img src="images/photo.jpg" alt="Photo of Betty" />`

Linking to a point in a web page

- Sometimes you want to link to a point *within* a web page
- E.g.
  - At the bottom of a very long web page, a link that says Go to top
  - A `<nav>` that contains links to each `<section>` in a page
  - One page could even link into the middle of another page
    Question: Why is this bad practice?
- To do this we need two ideas:
  - the `id` attribute
  - URL fragments

The `id` attribute

- Some attributes only apply to certain elements (e.g. `src` with `img`)
- Others, like `id`, are global, i.e. they apply to *all* elements
- **Question**: What other attribute have we covered that is global?
- You can invent the `id` attribute's value, but it should be unique within the document, e.g.:

```
<section id="ingredients">
```

URL fragments

- In general, URLs consist of eight parts:

```
scheme://user:password@host:port/pathname?query#fragment
```

- The fragment
  - comes at the end, after #
  - is the value of an `id` attribute

Example

```html
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <title>A Long Page</title>
    </head>
    <body>

        <h1 id="top">A Long Page</h1>

        <!-- Lots and lots and lots of paragraphs and other content here! -->

        <a href="#top">Go to top</a>

    </body>
</html>
```