CMPT 165

INTRODUCTION TO THE INTERNET AND THE WORLD WIDE WEB

By Hassan S. Shavarani

UNIT2: MARKUP AND HTML

TOPICS

1. Text Files and Markup
2. HTML and Tags
3. Attributes and More HTML Tags
4. Semantic Markup
5. Validating HTML
6. URLs: Links and Images
7. Character References

INVALID HTML EXAMPLE

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8" />
        <title>Invalid HTML Example</title>
    </head>
    <body>
        <h1>Invalid HTML Example</h1>
        <p>
            On this page, we will do some things wrong, so we can see the validator
            catch them.
        </p>

        <p>
            On the web, we can create links to pages like, the <a href="http://www.w3.
            org/"><abbr title="World Wide Web Consortium">W3C</a>, the group that
            defines web technologies like HTML.
        </p>

        <p>
            An <a href="http://validator.w3.org/>HTML Validator</a> is a tool that helps
            us find errors. Oops, that was another mistake.
        </p>
    </body>
</html>
```

we know that every (non-empty) tag must be closed

what if one is not closed?

how will *a web browser or a mobile browser, or Google, or other tools that read HTML* interpret that markup?

INVALID HTML EXAMPLE

Invalid HTML Example

On this page, we will do some things wrong, so we can see the validator catch them.

On the web, we can create links to pages like, the W3C, the group that defines web technologies like HTML.

An

INVALID HTML EXAMPLE

```
<p>… the <a href="http://www.w3.org/">
<abbr title="World Wide Web Consortium">W3C</a>,
the group that …</p>
```

- Which one is the correct render ?

… the W3C, the group that …

… the W3C, the group that ….

HTML Validation will prevent rendering inconsistency errors in different browsers!

HTML VALIDATOR

a tool that compares our document against the rules of HTML markup and tells us if there are any problems

the best validator to use is http://validator.w3.org/

if the validator gives many errors, **fix the first** one and then try validating again
lets try the validator with
http://www.cs.sfu.ca/CourseCentral/165/common/study-guide/files/invalid.html

THE ROBUSTNESS PRINCIPLE

when two pieces of software are communicating

- **the one sending out information** should be very careful what it sends, to make sure that the receiver understands it
- **the receiver** should do its best to gracefully handle small errors in what it receives

THE ROBUSTNESS PRINCIPLE

If both sides uphold their end of this bargain, everything should be transmitted correctly and there should be no errors or miscommunication

THE ROBUSTNESS PRINCIPLE

CLAIM: considering this principle, all the sentences provided here carry the same information, do they really do so ?

1. You are smart
2. Smart is a thing that you is
3. LOL ur smrt!!!
4. It is smart is a kind of thing you are

URLS: LINKS AND IMAGES

<IMG> TAG

used to refer to images
so they can be displayed on the page

```
<img src="http://example.com/vacation.jpg" alt="My vacation" />
```

the **src** attribute is used to give the URL of the image
the **alt** attribute gives alternate text
text that can be displayed in place of the image if it cannot be loaded

DIFFERENT TYPES OF URLS

- **Absolute URLs**
- **Relative URLs**

ABSOLUTE URLS

- the URLs that we have seen so far are absolute URLs
- absolute URLs contain a scheme, server, and path as described before
- absolute URLs can be used to locate a piece of content on the web with no other information or context

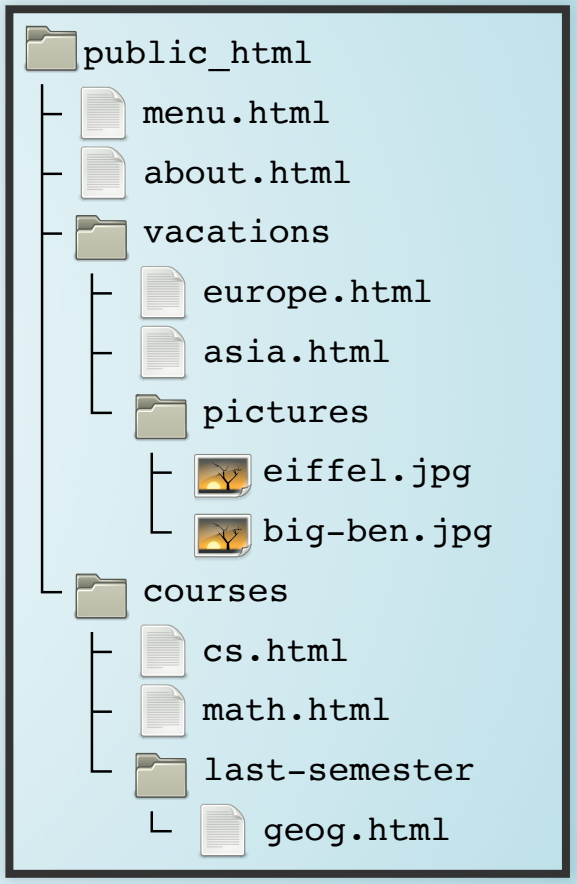
RELATIVE URLS

- they refer to content by giving its position relative to the current page
- they give information like *in the same folder, find the page nextpage.html* or *look in the folder images for the image selfie.jpg*

RELATIVE URLS

Different types of relative addressing

1. referring to the same folder as current html file (itself)
2. referring to a folder inside the containing folder (its children)
3. referring to a folder containing the containing foldr (its parents)



* image from http://www.cs.sfu.ca/CourseCentral/165/common/study-guide/figures/directory-structure.svg

RELATIVE URLS | CURRENT FOLDER EXAMPLE

```
<p>
    <a href="nextpage.html">next page</a>
    <img src="happy.png" alt="happy face" />
</p>
```

RELATIVE URLS | CHILDREN FOLDERS EXAMPLE

```
<p>
    <a href="vacations/europe.html">European vacation</a>
    <a href="vacations/pictures/eiffel.jpg">
        us at the Eiffel Tower
    </a>
    <img src="vacations/pictures/eiffel.jpg" alt="Eiffel Tower" />
</p>
```

RELATIVE URLS | PARENT FOLDERS EXAMPLE

```
<p>
    <a href="../menu.html">back to the menu</a>
    <a href="../../submenu.html">back to the submenu</a>
    <img src="../../vacations/pictures/big-ben.jpg" alt="Big Ben" />
</p>
```

IMPORTANT POINTS

- **URLs are case sensitive** (page.html is a different URL than Page.html or PAGE.HTML)
- slash (/) is the only character used to separate folders/filenames in URLs

CHARACTER REFERENCES FOR SPECIAL CHARACTERS

Character	Reference	Description
<	`<`	less-than (usually starts a tag)
>	`>`	greater-than (usually ends a tag)
&	`&`	ampersand (usually starts a reference)
"	`&quot;`	double quote (usually wraps attribute values)
'	`&apos;`	apostrophe/single quote (can wrap attribute values)

OTHER CHARACTER REFERENCES

Character	Reference	Description
€	`&euro;`	Euro sign
©	`&copy;`	Copyright symbol
“	`&ldquo;`	left double-quote
”	`&rdquo;`	right double-quote
…	`&hellip;`	ellipsis
λ	`&lambda;`	Greek lowercase lambda
✓	`&check;`	checkmark

NUMERIC CHARACTER REFERENCES

You can put any of the characters mentioned in
https://dev.w3.org/html5/html-author/charref
inside your HTML code

```
I was told that <abbr title="A &quot;great&quot; thing">AGT</abbr> costs
&gt;&euro;70. I wonder if it will really be &ldquo;great&rdquo; &amp; if it
will be worth it&hellip; probably not. &#128518;
```

I was told that AGT costs >€70. I wonder if it will really be “great” & if it will be worth it… probably not. 😆

HTML BASIC TAGS [OVERVIEW]

<b>	is for bold which also looks
<blockquote>	is of indented quoting
 	is for line break
	is for centring on the page
<h#>	is for heading, from 1 to 6, e.g. h1, h2, h3…
	is for horizontal rule
<i>	is for italic (same as is for emphasis)
<p>	is for paragraph
<pre>	is for pre formatted text
<q>	is for quoting texted (for citations)
<tt>	is for typewriter spacing (monospace), not = <pre>
<u>	is for underline

At the end lets look at
the *Inspection* Plugin of the browser ...

Any Questions?