EECS1012

Net-centric Introduction to Computing

Lecture 2:

HTML

Acknowledgements

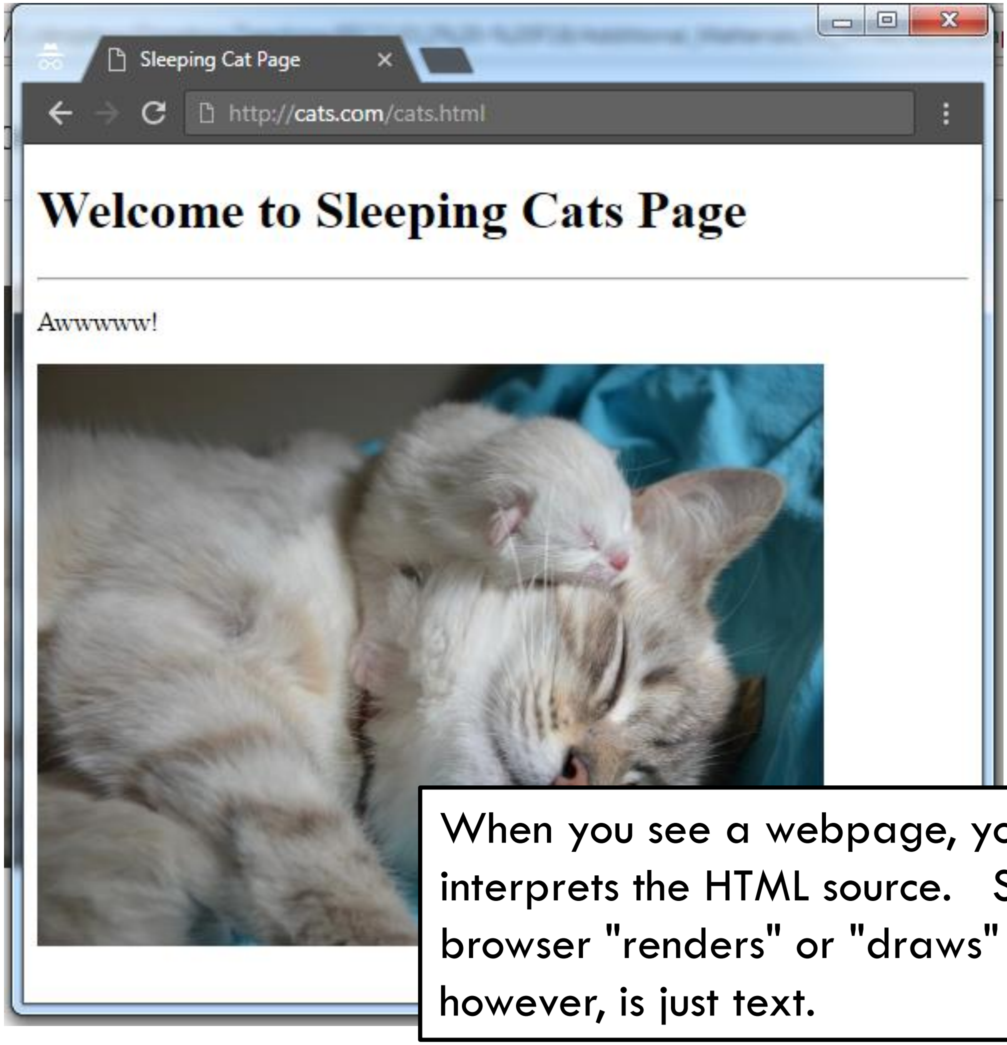
Contents are adapted from web lectures for “Web Programming Step by Step”, by M. Stepp, J. Miller, and V. Kirst

Slides have been ported to PPT by Dr. Xenia Mountrouidou

These slides have been edited for EECS1012, York University

Basic HTML

Webpage in browser vs. HTML

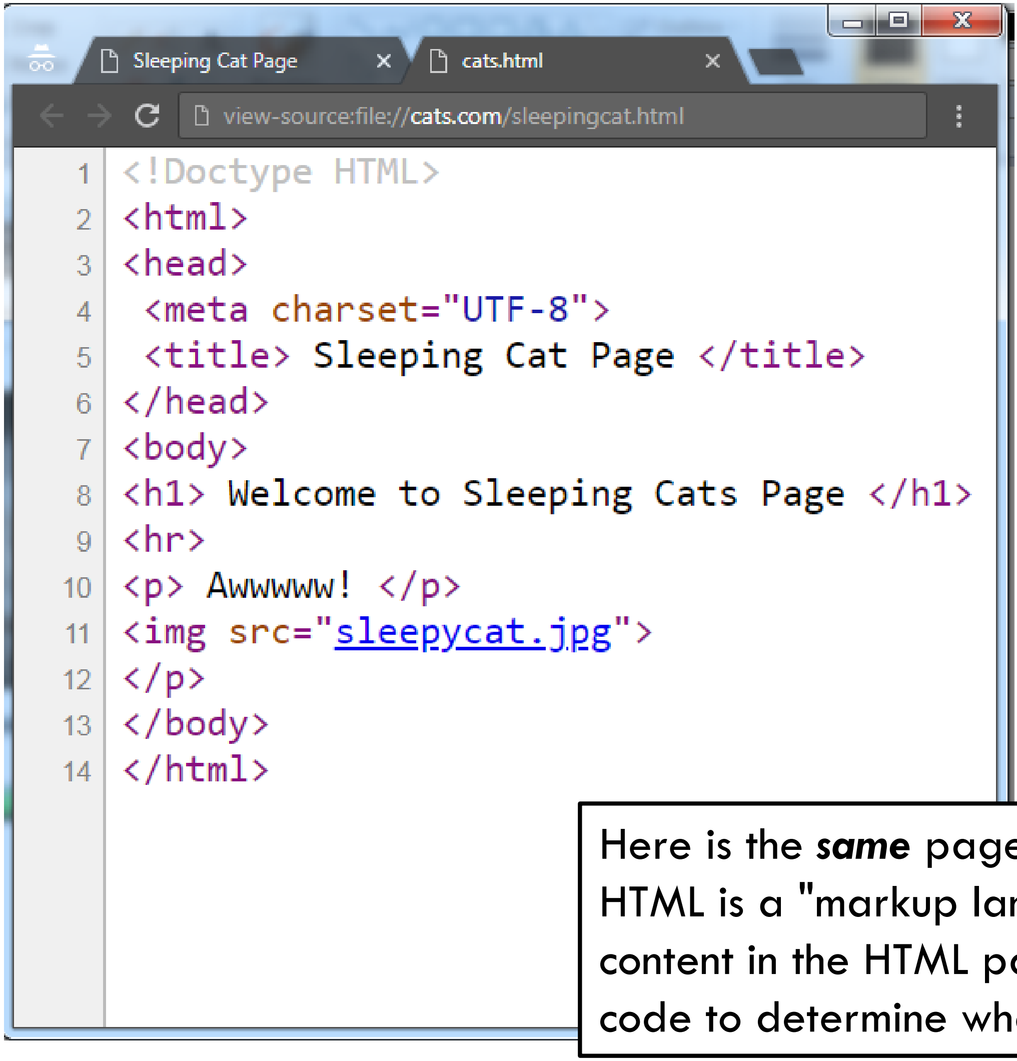


Welcome to Sleeping Cats Page

Awwwwww!

When you see a webpage, you are seeing how the browser interprets the HTML source. Sometimes we say this is how the browser "renders" or "draws" the page. The HTML source itself, however, is just text.

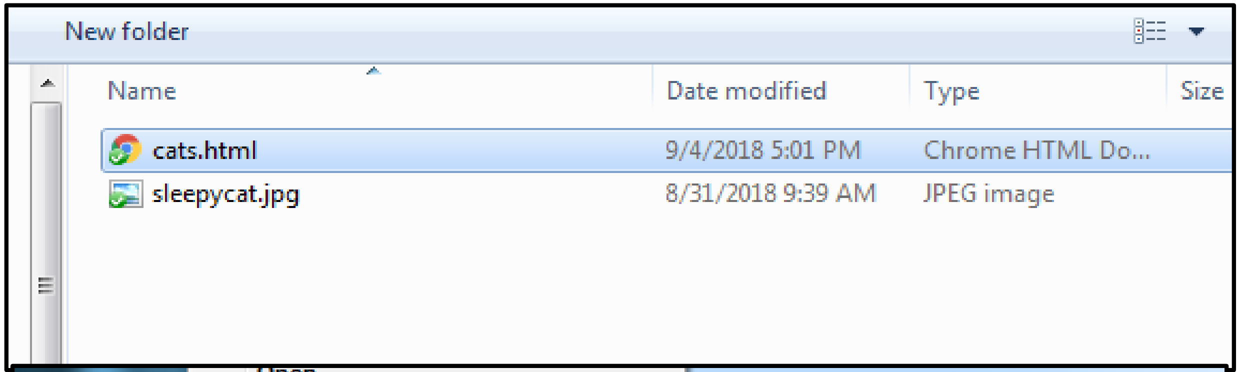
HTML is a text-based "language"



```
<!Doctype HTML>
<html>
<head>
 <meta charset="UTF-8">
 <title> Sleeping Cat Page </title>
</head>
<body>
<h1> Welcome to Sleeping Cats Page </h1>
<hr>
<p> Awwwww! </p>
<img src="sleepycat.jpg">
</p>
</body>
</html>
```

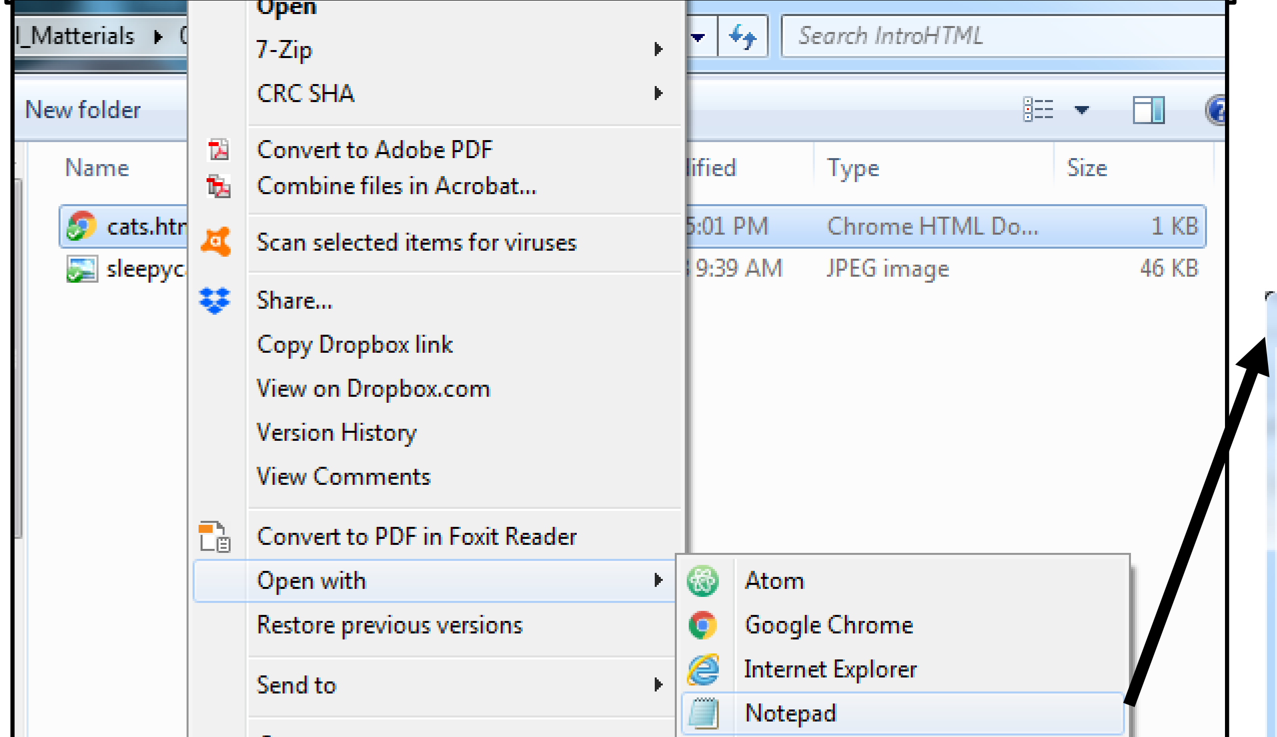
Here is the ***same*** page, but showing the HTML source. HTML is a "markup language". It is used to describe the content in the HTML page. The browser interprets the HTML code to determine what content to draw on the screen.

HTML files and your folders



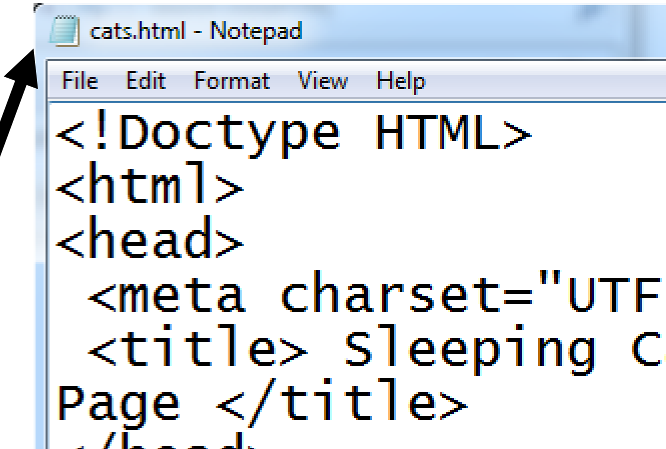
Name	Date modified	Type	Size
cats.html	9/4/2018 5:01 PM	Chrome HTML Do...	
sleepycat.jpg	8/31/2018 9:39 AM	JPEG image	

If you have an HTML file in a folder, if you click on the icon it will launch the browser.



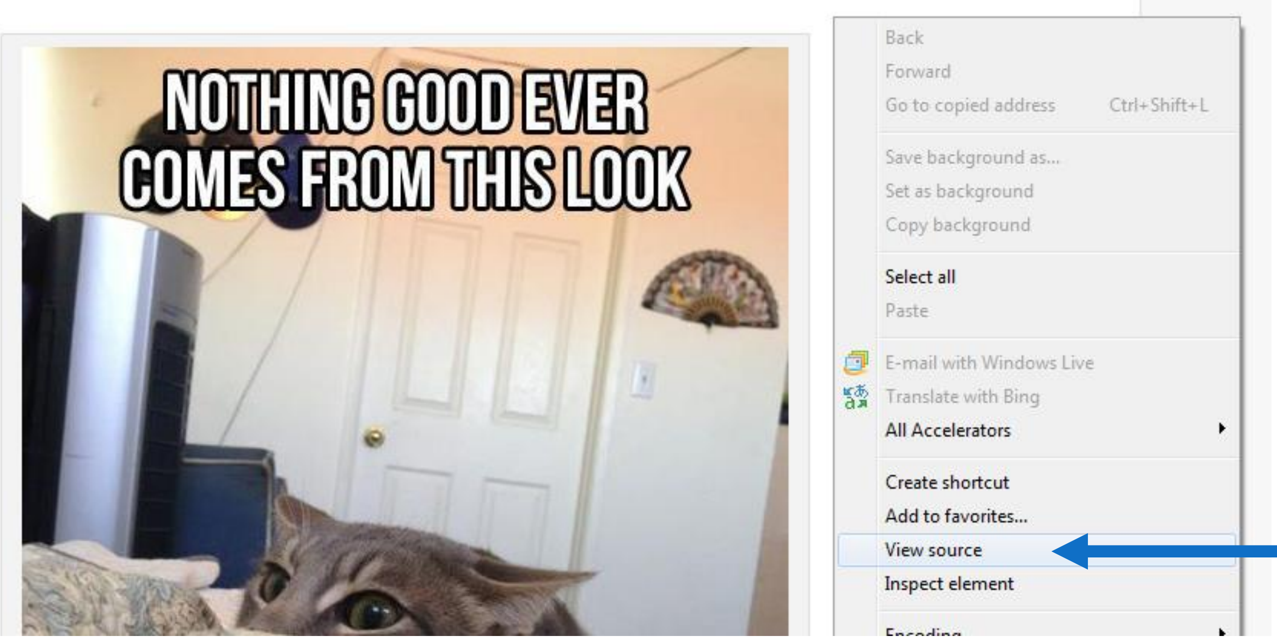
Name	...ified	Type	Size
cats.htr	5:01 PM	Chrome HTML Do...	1 KB
sleepyc	9:39 AM	JPEG image	46 KB

If you right click on the file, and then select "open with", you can edit the HTML code directly!



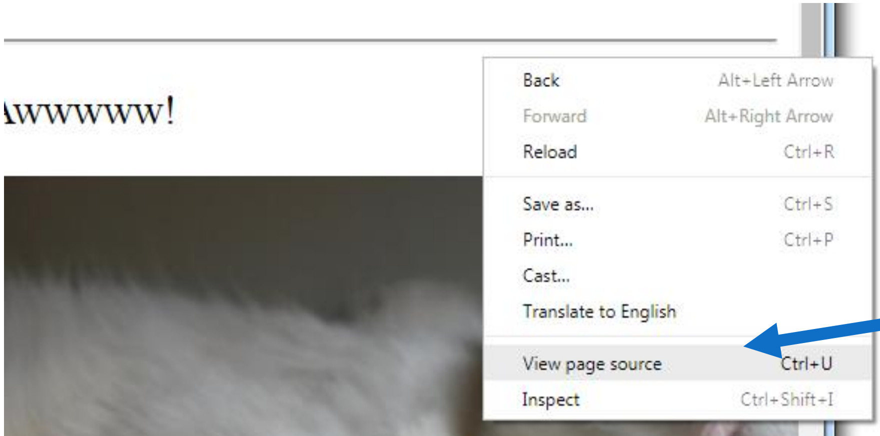
```
<!Doctype HTML>
<html>
<head>
 <meta charset="UTF
 <title> Sleeping C
Page </title>
```

Viewing the HTML "source"



All browsers provide a way for you to see the true HTML "source".

Internet Explorer
Right-click on page - select "View source"



Google Chrome
Right-click on page - select "View page source"

Learning the basics of HTML

In this lecture, we will learn the basics of writing HTML source code.

All you need is a text editor (e.g., Notepad, PSPad, Atom, etc.) . .

and a web browser (e.g., Chrome, IE, Safari, etc.)

Hypertext Markup Language (HTML)

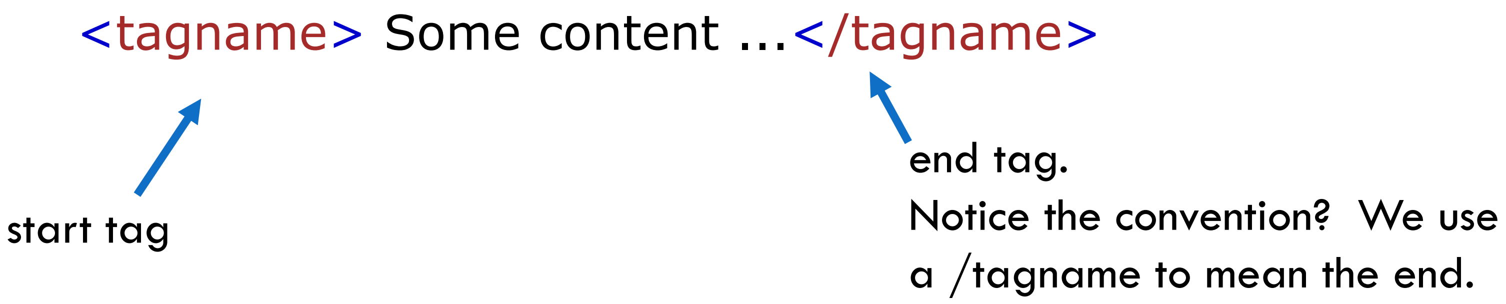
- Describes the *content* and structure of information on a web page
- Not the same as the presentation (appearance on the screen)
- Surrounds text content with opening and closing tags
- Each tag's name is called an element
  - syntax: `<element>` content `</element>`
  - example: `<p>`This is a paragraph`</p>`

HTML5

- The latest version of HTML, standardize in 2017
- Initial recommendation started in 2008
  - Gives you an idea of how long it takes to standardize a markup language
- Our examples will focus on HTML5
- IE in Windows may not always support HTML5 correctly
  - Advice? Get yourself a better browser!

HTML elements

An HTML element usually consists of a **start** tag and **end** tag, with the content inserted in between:



```
<tagname> Some content ...</tagname>
```

start tag

end tag.
Notice the convention? We use a /tagname to mean the end.

HTML elements

- There are many types of HTML elements
  - This lecture will over the basic ones

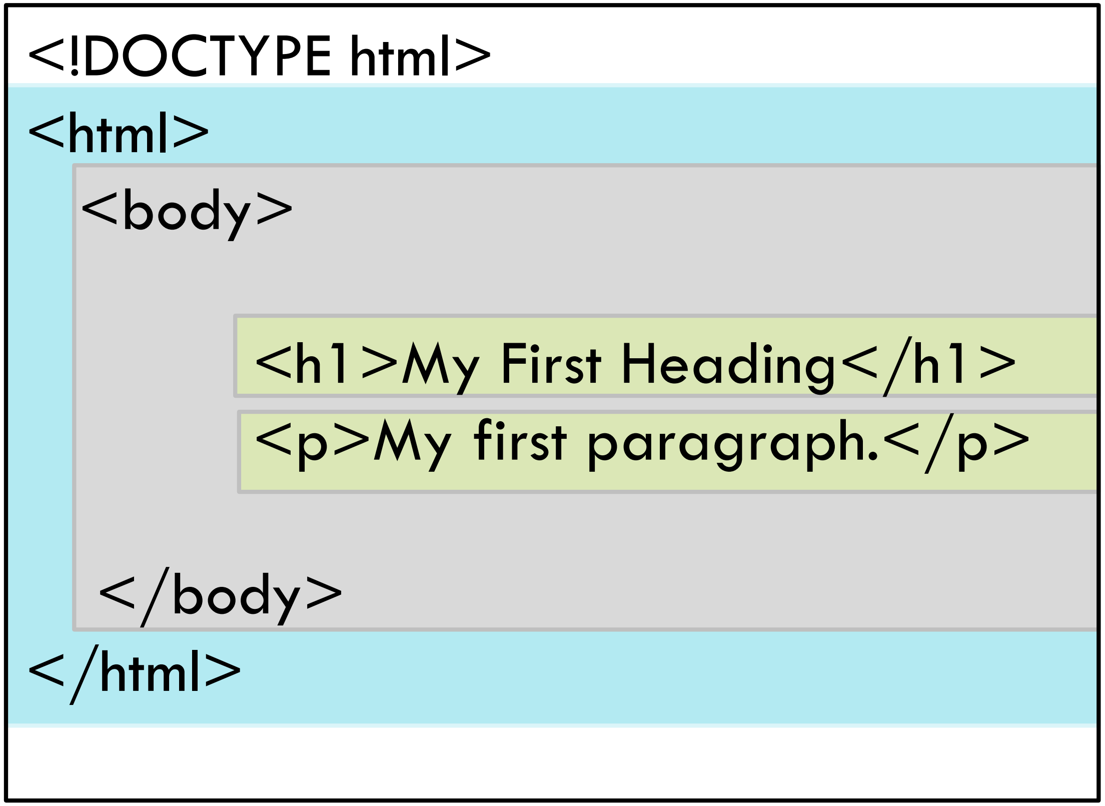
```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

This HTML document
has **four** (4) elements.

A <html> element
 - called a whole document element
A <body> element.
A <h1> element (header).
A <p> paragraph element.

Nested elements

- Notice that some elements are contained inside other elements. We call these "nested" element.



```
<!DOCTYPE html>
<html>
  <body>

        <h1>My First Heading</h1>
        <p>My first paragraph.</p>

   </body>
</html>
```

The <body> element is contained within the <html> element.

The <h1> is contained within the <body> element.

The <p> is also contained within the body element, but not within the <h1> element.

We can say that <h1> is **nested** within the <body> element.

Structure of an HTML page

```
<!DOCTYPE html>
<html>
        <head>
                header – contains information about the page
        </head>
        <body>
                page contents
        </body>
</html>                                                HTML
```

- HTML is saved with an extension .html
- Basic structure: tags that enclose content, i.e., elements
- **Doctype** describes the type of document for the browser.
  - We often refer to a webpage page as a "document".
- **Head** describes information about the HTML page's content
- **Body** contains the page's contents

Structure of an HTML page

```
<!DOCTYPE html>
<html>
       ..
</html>                                                        HTML
```

- All pages should start with a <!DOCTYPE html>. This is not part of the HTML standard! But it is required by the browsers.
- **However,** this is used by the browser to know what type of HTML document it is loading (e.g. maybe an older HTML 2.0, etc – see here)
- <!DOCTYPE html> is used for HTML5, and is required to be considered a valid html5 document. This will be important when we validate our HTML files later.

Page title <head> and <title>

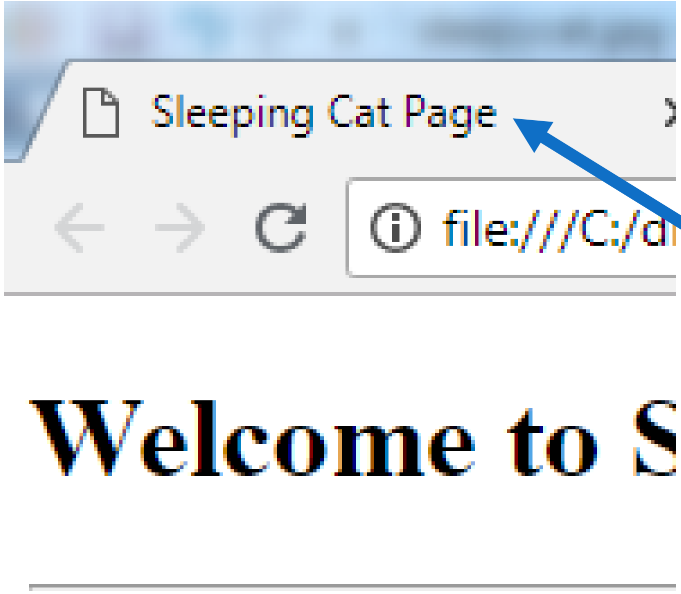
```
…
        <head>
               <title> Lecture on HTML  </title>
        </head>
…
```

HTML

- □ Head element contains information about the webpage. This is often used for search engines.
- □ Title element is placed within the head element
- □ Displayed in web browser's title mark and when bookmarking the page.

<title>

- The title element is often shown in the browser's tab



Sleeping Cat Page

file:///C:/d

Welcome to S

```
<!Doctype HTML>
<html>
<head>
 <meta charset="UTF-8">
 <title> Sleeping Cat Page </title>
</head>
<body>
</html>
```

Specifying the document encoding

```
…
        <head>
                <meta charset="UTF-8">
        </head>
…
```

HTML

- **HTML5 requires that the charset used to encode the document be specified.**
- UTF is a standard 8-bit encoding used by virtually all servers these days.

Paragraph <p>

```
…
       <body>
       <p>Whatever you do will be    insignificant,
but it is very important that you do it.</p>
       </body>
…
                                                              HTML
```

```
Whatever you do will be insignificant, but it is very important that you do it.

                                                              output
```

- □ Placed within the body of the page
- □ Note that whitespace and "new lines"* are ignored.

*What is a "new line"? When you hit "enter" on your keyboard, it generates a "new line" character that moves to the next line in the document.

Headings <h1>, <h2>, … <h6>

```
<h1> Harry Potter </h1>
<h2> Books </h2>
<h3> Harry Potter and the Philosopher’s Stone </h3>
```

HTML

```
Harry Potter
Books
Harry Potter and the Philosopher’s Stone
```

output

More examples

Horizontal rule <hr>

```
<p> First paragraph </p>
<hr>
<p> Second Paragraph </p>
```

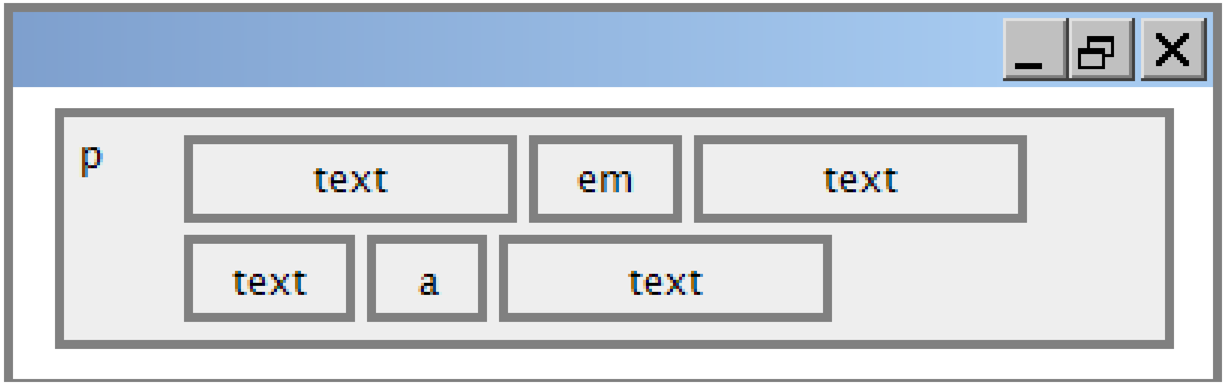
HTML

```
First Paragraph

Second Paragraph
```

output

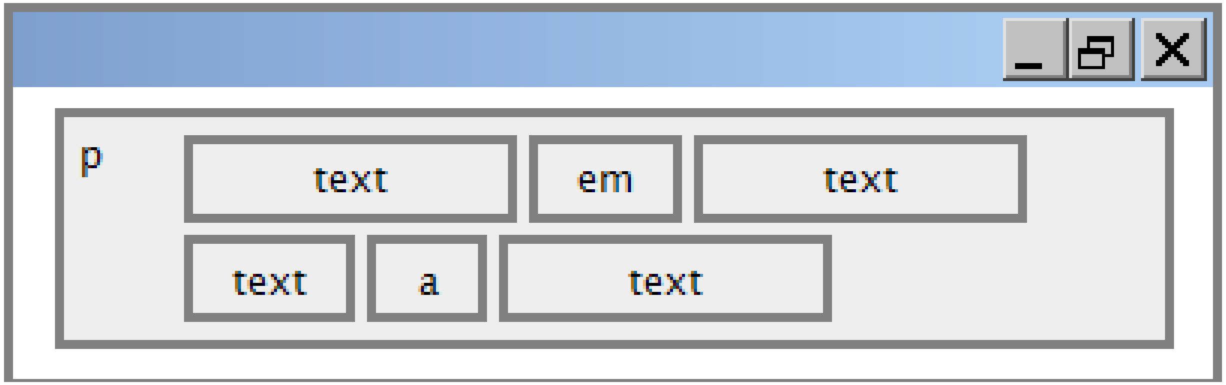
- Generates a line between paragraphs.
- Note that this does not require a closing </hr>
- This type of element is called an empty element (or a void element)

Block and Inline elements



- **Block elements** contain an entire large region of content
  - examples: paragraphs, lists, table cells
  - the browser places a margin of whitespace between block elements for separation

Block and Inline elements (cont.)



- **Inline elements** affect a small amount of content
  - Examples: bold text, code fragments, images
  - the browser allows many inline elements to appear on the same line
  - **inline elements should be nested inside a block element**

More info on block and inline

- See more information here:

https://www.w3schools.com/html/html_blocks.asp

- When you use CSS (next lecture) you will often need to remember if an element is a block or an inline element - it will behave differently.

More about HTML tags

- Some tags can contain additional information called *attributes*
  - syntax:

    ```
    <element attribute="value"
    attribute="value"> content </element>
    ```

  - example: <a href="page2.html">Next page</a>

Even more about HTML tags

- Some tags don't contain content between the tags (they may specify content – but not between the tags); these tags can be opened and closed in one tag
  - syntax:

```
<element attribute="value" attribute="value">
```

  - example: **<hr>**
  - example:

```
<img src="Harry.jpg" alt="pic of Harry Potter">
```

Links or hyper references <a>

```
<p>
Search
<a href="http://www.google.com/">Google</a>
now!
</p>
```

HTML

```
Search Google now!
```

output

- The **href** attribute specifies the destination URL
- Links or *anchors* are inline elements, so they must be placed inside a block element such as a `p` or `h1`

More about links/anchors

```
<p><a href="deathlyHallows-book.html">Harry Potter and the
Deathly Hallows Book</a></p>

<p><a href="http://en.wikipedia.org"
title="Search">Wikipedia</a></p>
```

HTML

[Harry Potter and the Deathly Hallows](deathlyHallows-book.html)

[Wikipedia](http://en.wikipedia.org)

output

- Types of URLs that can appear in anchors:
  - Absolute: to another web site
  - Relative: to another page on this web site

Target _blank attribute

```
<p><a target="_blank" href="http://en.wikipedia.org”
title="Search">Wikipedia</a></p>
                                                          HTML
```

```
Wikipedia                                               output
```

- Target attribute
  - Setting the target attribute to “_blank” (with a leading underscore character) will force the link to open in a new tab or page.

Nesting tags incorrectly

Bad . . don't do this

```
<p>
<a href=" deathlyHallows-book.html"> Harry Potter and the
Deathly Hallows Book </p>
<p>
This text also links to Harry Potter Book </a>
</p>
```

<p> tag was closed before the <a> tag.

HTML

- **Tags must be correctly nested:** a closing tag must match the **most recently opened tag**
- The browser may draw it correctly anyway, but it is invalid HTML5

* I'll use a "Deaner" icon to denote a bad example.

Image element <img>

```
<img src="tobby.jpg" alt="Tobby from Harry Potter">
                                                              HTML
```

- The src attribute specifies source of the image URL
- **HTML5 also requires an alt attribute describing the image . . this gives a text description when for some reason the user cannot view it (slow connection, text only browser)**

Image element <img>

```
<img src="tobby.jpg" alt="Tobby from Harry Potter">
<img src="https://catzone-tcwebsites.netdna-ssl.com/wp-content/uploads/2017/11/cool-cat-with-sunglasses.jpg" alt="Cat">
```

HTML

- The `src` attribute specifies source of the image URL
- The src can specify an image saved in the same folder as your HTML document
- Or the src can specify the URL of an image (2nd example above is a full URL link to an image)

Images as a *link* content

```
<a href="http://www.mugglenet.com/”>
<img src="dumbledore.jpg" alt="Dumbledore from Harry Potter">
</a>
```

HTML

- If placed inside an a anchor, the image will become a link (click on the image and it will follow the link)

<img> height attribute

```
<p> <img alt="cat image" src="sleepycat.jpg"> </p>
<p> <img alt="cat image" height="50" src="sleepycat.jpg"> </p>
```

HTML

A browser will draw an image at its original size.

You can control the size using the height attribute (or weight attribute) in the img tag.

If you only set one attribute, it will rescale the other to match.
For example, if the original size is:
1000 width x 500 height

Setting height="250" will result in an image size 500 x 250.

Line break <br>

```
<p>Teddy said it was a hat, <br> So I put it on.</p>
<p>Now Daddy's sayin', <br> Where
the heck's the toilet plunger gone?</p>
```

HTML

```
Teddy said it was a hat,
So I put it on.
Now Daddy's sayin',
Where the heck's the toilet plunger gone?
```

output

- br should not be used to separate paragraphs or used multiple times in a row to create spacing

Comments <!-- … -- >

```
<!-- My web page, by Deaner
EECS 1012, Fall 2048 -->
<p>EECS courses are <!-- NOT --> a lot of fun!</p>
                                                   HTML
```

```
EECS courses are a lot of fun!
                                                 output
```

- □ Comments are useful for disabling sections of a page
- □ Useful for adding in information about the page
- □ Comments cannot be nested!

Phrase elements <em>, <strong>

```
<p>
HTML is <em>really</em>,
<strong>REALLY</strong> fun!
</p>
```

HTML

HTML is *really* **REALLY** fun!

output

- **em:** emphasized text (usually in italic)
- **strong:** strongly emphasized text (usually in bold)
- The tags must be properly nested for a valid page

Unordered list: <ul>, <li>

```
<ul>
<li>No shoes</li>
<li>No shirt</li>
<li>No problem!</li>
</ul>
```

HTML

```
•  No shoes
•  No shirt
•  No problem!
```

output

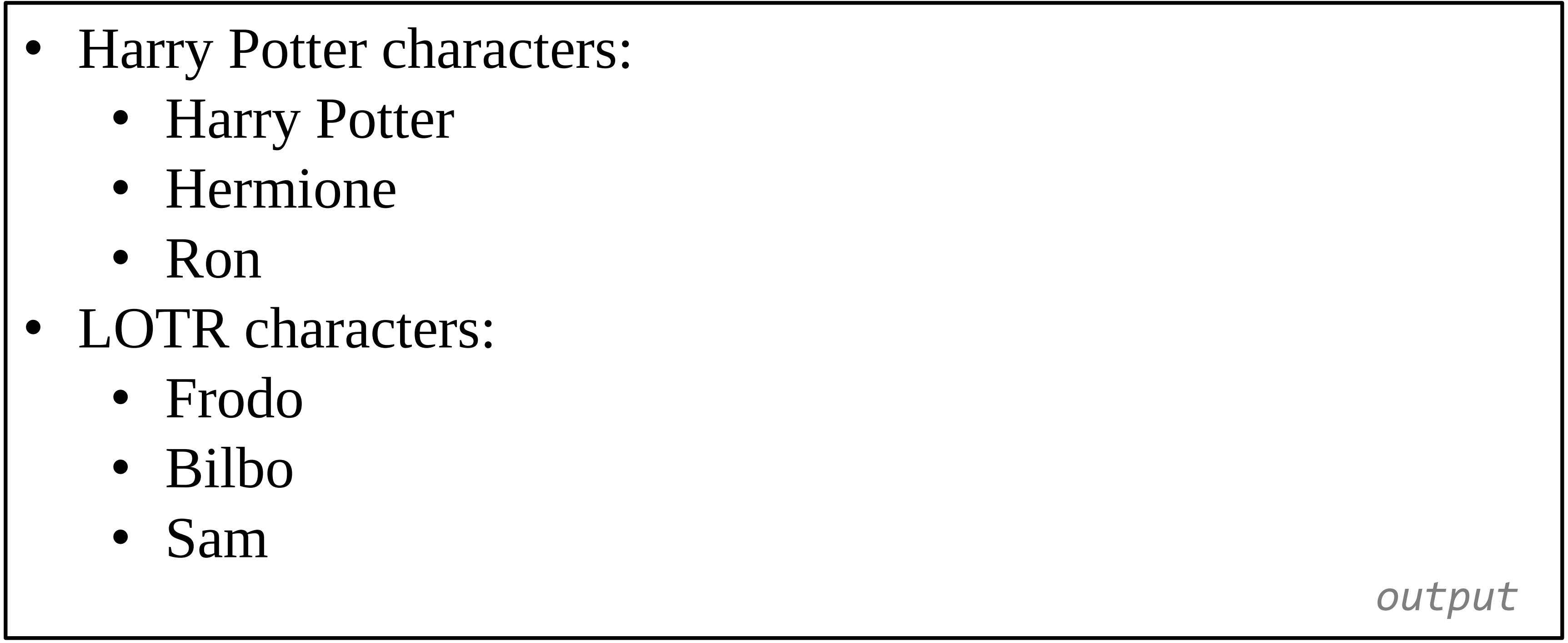
- **ul** represents a bulleted list of items (block)
- **li** represents a single item within the list (block)

Nesting lists

```
<ul>
       <li>Harry Potter characters:
               <ul>
                  <li>Harry Potter</li>
                  <li>Hermione</li>
                  <li>Ron</li>
               </ul>
       </li>
       <li>LOTR characters:
           <ul>
               <li>Frodo</li>
               <li>Bilbo</li>
               <li>Sam</li>
           </ul>
       </li>
</ul>
```

HTML

Output from previous slide



```
•  Harry Potter characters:
      •  Harry Potter
      •  Hermione
      •  Ron
•  LOTR characters:
      •  Frodo
      •  Bilbo
      •  Sam
```

Ordered list <ol>

```
<p>Apple business model:</p>
<ol>
<li>Beat Microsoft</li>
<li>Beat Google</li>
<li>Conquer the world!</li>
</ol>
```

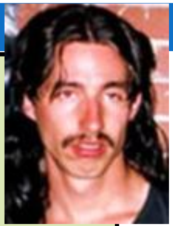
HTML

```
Apple business model:
1.  Beat Microsoft
2.  Beat Google
3.  Conquer the world
```

output

- **ol** represents a numbered list of items
- we can make lists with letters or Roman numerals using CSS (later)

Common error: Not closing a list



```
<ul>
<li>No shoes</li>
<li>No shirt</li>
<li>No problem!</li>
<p>Paragraph after list...</p>
```

HTML

```
•    No shoes
•    No shirt
•    No problem!

     Paragraph after list...
```

output

- □ If you leave a list open, subsequent contents will be indented (to correct, you need to insert a `</ul>`)

Common Error!: Improper nested list placement

```
<ul>
<li>Harry Potter characters:</li>
<ul>
<li>Harry Potter</li>
<li>Hermione</li>
<li>Ron</li>
</ul>
</li>
<li>LOTR characters:
<ul>
<li>Frodo</li>
<li>Bilbo</li>
<li>Sam</li>
</ul>
</ul>                                                   HTML
```

- closing the outer li too early (or not at all) will render correctly in most browsers, but it is incorrect HTML

Definition list <dl>, <dt>, <dd>

```
<dl>
<dt>newbie</dt> <dd>one who does not have mad skills</dd>
<dt>jaded</dt> <dd>tired, bored, or lacking enthusiasm </dd>
<dt>frag</dt> <dd>a kill in a shooting game</dd>
</dl>
```

HTML

```
newbie
        one who does not have mad skills
jaded
        Tired, bored, or lacking enthusiasm
frag
        a kill in a shooting game
```

output

- □ **dl** represents a list of definitions of terms
- □ **dt** represents each term, and **dd** its definition

Tables <table>, <tr>, <td>

```
<table>
	<tr><td>1,1</td><td>1,2 okay</td></tr>
	<tr><td>2,1 real wide</td><td>2,2</td></tr>
</table>
```

HTML

```
1,1                              1,2 okay
2,1 real wide                    2,2
```

output

- □ `table` **defines the overall table,** `tr` **each row, and** `td` **each cell's data**
- □ **Useful for displaying large row/column data sets**
- □ **NOTE: tables are sometimes used by novices for web page layout, but this is not proper semantic HTML and should be avoided**

Table headers, captions: <th>, <caption>

```
<table>
       <caption>My important data</caption>
       <tr><th>Column 1</th><th>Column 2</th></tr>
       <tr><td>1,1</td><td>1,2 okay</td></tr>
       <tr><td>2,1 real wide</td><td>2,2</td></tr>
</table>
```

HTML

```
My important data
Column 1                    Column 2
1,1                         1,2 okay
2,1 real wide               2,2
```

output

- `th` cells in a row are considered headers
- a caption at the start of the table labels its meaning

Quotations <blockquote>

```
<p>As Lincoln said in his famous Gettysburg Address:</p>
       <blockquote>
       <p>Fourscore and seven years ago, our fathers
brought forth
       on this continent a new nation, conceived in
liberty, and
       dedicated to the proposition that all men are
created equal.</p>
</blockquote>                                          HTML
```

```
As Lincoln said in his famous Gettysburg Address:
        Fourscore and seven years ago, our fathers brought forth on this
        continent a new nation, conceived in liberty, and dedicated to the
        proposition that all men are created equal.
                                                                   output
```

- a lengthy quotation

Inline quotations <q>

```
<p>Quoth the Raven, <q>Nevermore.</q></p>
```

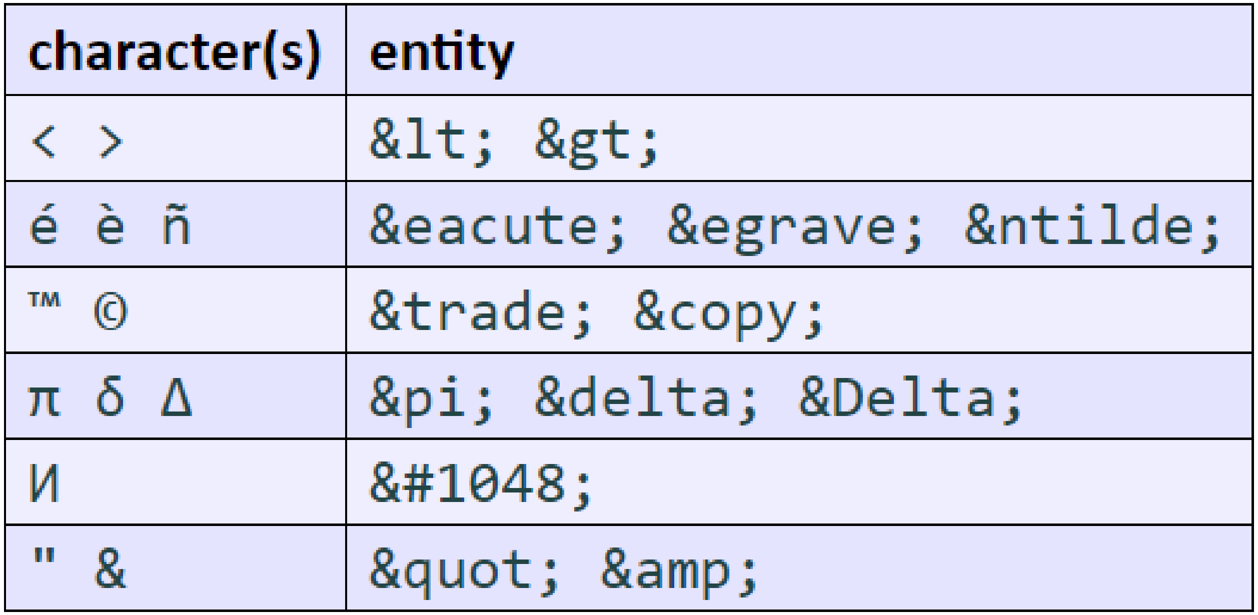
HTML

```
Quoth the Raven, “Nevermore.”
```

output

- a short quotation
- Why not just write the following?
- <p>Quoth the Raven, "Nevermore."</p>
  - using <q> allows us to apply CSS styles to quotations

HTML character entities



character(s)	entity
< >	&lt; &gt;
é è ñ	&eacute; &egrave; &ntilde;
™ ©	&trade; &copy;
π δ Δ	&pi; &delta; &Delta;
И	&#1048;
" &	&quot; &amp;

- List of characters entities
- Question: how would you display the text &amp; on a web page?

Using entities

```
&lt;p&gt;
&lt;a href=&quot;http://google.com/search?q=deaner&amp;ie=utf-8&amp;aq=t&quot;&gt;
Search Google for Deaner
&lt;/a&gt;
&lt;/p&gt;
```

HTML

```
<p> <a href="http://google.com/search?q=deaner&ie=utf-8&aq=t"> Search
Google for Xenia </a> </p>
```

output

- □ To display the link text in a web page, its special characters must be encoded as shown above

Computer code <code>

```
<p>
The <code>ul</code> and <code>ol</code>
tags make lists.
</p>
```

HTML

The `ul` and `ol` tags make lists.

output

- □ code: a short section of computer code

Preformatted text <pre>

```
<pre>
 Bill Gates speaks
    You will be assimilated
  Microsoft fans delirious
</pre>
```

HTML

```
Bill Gates speaks
    You will be assimilated
  Microsoft fans delirious
```

output

- Displayed with exactly the whitespace / line breaks given in the text
- Shown in a fixed-width font by default

<pre> and <code> together

```
<pre><code>
       public static void main(String[] args) {
           System.out.println("Hello, world!");
       }
</code></pre>
```

HTML

```
public static void main(String[] args) {
           System.out.println("Hello, world!");
       }
```

output

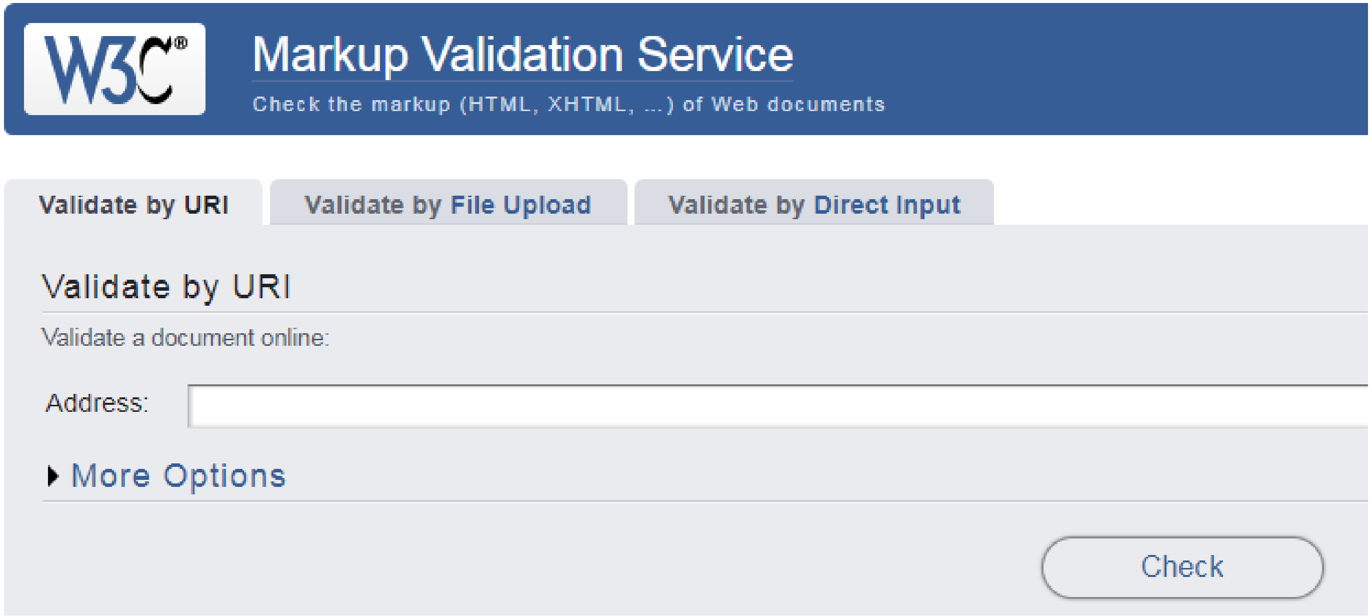
- When showing a large section of computer code, enclose it in a `pre` to preserve whitespace and a `code` to describe the semantics of the content

Web standards

- Why use HTML5 and web standards?
  - more rigid and structured language
  - more interoperable across different web browsers
  - more likely that our pages will display correctly in the future

W3C HTML5 Validator

https://validator.w3.org/



Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI

Validate a document online:

Address:

- checks your HTML code to make sure it meets the official strict HTML5 specifications
- **Very useful** for finding bugs in your HTML code

Web page metadata <meta>

```html
<head>
<meta name="description"
content="Harry Potter Official Website.">
<meta name="keywords" content="harry potter, harry potter and
the deathly hallows">
<meta charset="UTF-8">
</head>
```

HTML

- Nested in the <head> section
- information about your page (for a browser, search engine, etc.)
- meta tags often have both the name and content attributes
- **HTML 5 requires the charset meta information.**

meta element to describe the page

```
<head>
<meta name="author"
content="web page's author">
<meta name="revised"
content="web page version and/or last modification date">
<meta name="generator"
content="the software used to create the page">
</head>
```

HTML

- many WYSIWYG HTML editors (FrontPage, PageMaker, etc.) place their names in the meta generator tag (why?)

Summary

- You have learned the basics of HTML
  - There is a lot of information in this lecture, esp if you are new to HTML
  - Practice in Lab #1
- Understanding HTML is necessary when we learn JavaScript
  - JavaScript is designed to help modify HTML in the browser
  - You must understand HTML in order to be able to manipulate it
- Next – Cascading Style Sheets