HTML in one lecture

Web Dev, Spring 2021

HTML

A notation for "structured" text documents with links to other documents

- The basis of web content — or the skeleton

The small print:

- HTML = HyperText Markup Language
- An instance of SGML (Standard Generalized Markup Language) like DocBook and others
- SGML is an ancestor (uncle?) of XML, yet another markup language
- You still encounter XML in Java and .NET — rest of the world has moved to JSON
- We're on version 5 of HTML called HTML 5.
- The history of HTML is interesting and frustrating
- Compatibility after 30 years — the first web pages mostly still render

The basics

```
<html>
  <head>
    <title> Best Web Page Evah! </title>
  </head>

  <body>
    <!-- this is the part that actually gets rendered by browsers -->
    <h1> Sample HTML </h1>
    <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <p> Phasellus ipsum tellus, <em>malesuada</em> efficitur venenatis ut, aliquam at ipsum. </p>
    <ol>
      <li> Fusce consequat tellus nec diam aliquet, sit amet viverra dolor feugiat. </li>
      <li> Etiam nec euismod mauris, et elementum purus. </li>
    </ol>
    <img src="cat.jpg">
  </body>

</html>
```

The basics

Tags describe pieces of the document
(via an opening tag and closing tag)

```
<html>
  <head>
    <title> Best Web Page Evah! </title>
  </head>

  <body>
    <!-- this is the part that actually gets rendered by browsers -->
    <h1> Sample HTML </h1>
    <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <p> Phasellus ipsum tellus, <em>malesuada</em> efficitur venenatis ut, aliquam at ipsum. </p>
    <ol>
      <li> Fusce consequat tellus nec diam aliquet, sit amet viverra dolor feugiat. </li>
      <li> Etiam nec euismod mauris, et elementum purus. </li>
    </ol>
    <img src="cat.jpg">
  </body>

</html>
```

The basics

```
<html>
  <head>
    <title> Best Web Page Evah! </title>
  </head>

  <body>
    <!-- this is the part that actually gets rendered by browsers -->
    <h1> Sample HTML </h1>
    <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <p> Phasellus ipsum tellus, <em>malesuada</em> efficitur venenatis ut, aliquam at ipsum. </p>
    <ol>
      <li> Fusce consequat tellus nec diam aliquet, sit amet viverra dolor feugiat. </li>
      <li> Etiam nec euismod mauris, et elementum purus. </li>
    </ol>
    <img src="cat.jpg">
  </body>

</html>
```

Some tags have attributes
(think of them as tag parameters)

The basics

Everything between an opening and closing tag is the content of the tag

```
<html>
  <head>
    <title> Best Web Page Evah! </title>
  </head>

  <body>
    <!-- this is the part that actually gets rendered by browsers -->
    <h1> Sample HTML </h1>
    <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <p> Phasellus ipsum tellus, <em>malesuada</em> efficitur venenatis ut, aliquam at ipsum. </p>
    <ol>
      <li> Fusce consequat tellus nec diam aliquet, sit amet viverra dolor feugiat. </li>
      <li> Etiam nec euismod mauris, et elementum purus. </li>
    </ol>
    <img src="cat.jpg">
  </body>

</html>
```

The basics

Everything between an opening and closing tag is the content of the tag

```
<html>
  <head>
    <title> Best Web Page Evah! </title>
  </head>

  <body>
    <!-- this is the part that actually gets rendered by browsers -->
    <h1> Sample HTML </h1>
    <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <p> Phasellus ipsum tellus, <em>malesuada</em> efficitur venenatis ut, aliquam at ipsum. </p>
    <ol>
      <li> Fusce consequat tellus nec diam aliquet, sit amet viverra dolor feugiat. </li>
      <li> Etiam nec euismod mauris, et elementum purus. </li>
    </ol>
    <img src="cat.jpg">
  </body>

</html>
```

The basics

That content may itself be made up of others tags with their own content

```
<html>
  <head>
    <title> Best Web Page Evah! </title>
  </head>

  <body>
    <!-- this is the part that actually gets rendered by browsers -->
    <h1> Sample HTML </h1>
    <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <p> Phasellus ipsum tellus, <em>malesuada</em> efficitur venenatis ut, aliquam at ipsum. </p>
    <ol>
      <li> Fusce consequat tellus nec diam aliquet, sit amet viverra dolor feugiat. </li>
      <li> Etiam nec euismod mauris, et elementum purus. </li>
    </ol>
    <img src="cat.jpg">
  </body>

</html>
```

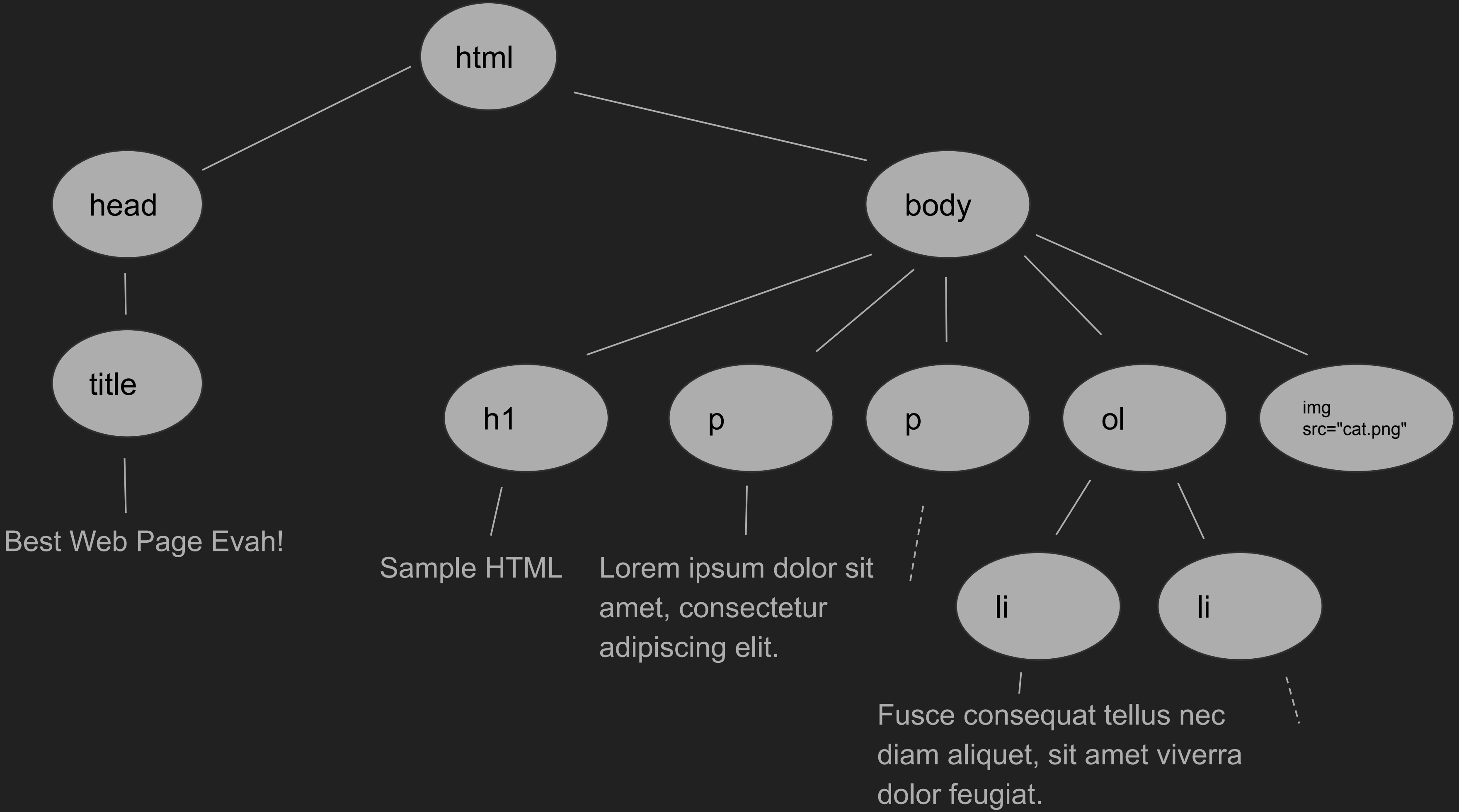
The basics

Tags naturally define a tree structure for the document

```
<html>
  <head>
    <title> Best Web Page Evah! </title>
  </head>

  <body>
    <!-- this is the part that actually gets rendered by browsers -->
    <h1> Sample HTML </h1>
    <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <p> Phasellus ipsum tellus, <em>malesuada</em> efficitur venenatis ut, aliquam at ipsum. </p>
    <ol>
      <li> Fusce consequat tellus nec diam aliquet, sit amet viverra dolor feugiat. </li>
      <li> Etiam nec euismod mauris, et elementum purus. </li>
    </ol>
    <img src="cat.jpg">
  </body>

</html>
```


html
head
body
title
Best Web Page Evah!
h1
p
p
ol
img src="cat.png"
Sample HTML
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
li
li
Fusce consequat tellus nec diam aliquet, sit amet viverra dolor feugiat.

Categories of tags

Structural	<head>, <body>, <h1>, … , <h6>, <p>, ... <div>, <span>
Formatting	<em>, <strong>, <code>, <b>, <i>, <u>, <pre>, <sup>, <sub>, ...
List	<ul>, <ol>, <li>, ...
Table	<table>, <thead>, <tbody>, <tr>, <th>, <td>
Metadata	<title>, <style>, <meta>, …
Embedded Content	<img>, <audio>, <video>, <canvas>, <svg>, <iframe>, <object>, ...
Form (controls)	<button>, <input>, <select>, ...
Scripting	<script>, ...

Entities

How do we handle "special" characters?

```
<p> This is an inequation : x < y </p>
```

Entities let you escape characters that have meaning in HTML: <, >, &

```
<p> This is an inequation : x &lt; y </p>
```

Lots of entities defined, including mathematical symbols: &infin;

Can use an arbitrary unicode characters: &#x2702;

HTML → DOM tree representation

Web browsers transform HTML into a tree defined by the Document Object Model

Types of nodes in the tree:

- Element:	mostly corresponding to tags
- Attributes:	attributes in tags are treated as special children of elements
- Text:	text inside tags

Content of a tag are children of the corresponding Element node

Browsers implement a specific rendering algorithm

Every node is either a block node or an inline node (depending on the tag — text is inline)

- block nodes are rendered vertically, one above the other
- children of a block node are rendered into sub-blocks (one per child), except that:
- adjacent inline nodes get merged into a single block

Intuition: paragraphs (blocks) of text (inline)

Layout follows the tree structure — e.g., <p> A <em>B</em> <h2>C</h2> D </p>

- layout algorithm defines a block for <p>, within it there are three subblocks:
  - A <em>B</em> (the two inline nodes A and <em>B</em> form one block)
  - <h2>C</h2>
  - D (the one inline node forms a block by itself)

Things get more complicated when you add layout-affecting styling

Styling

Tags and the tree structure describe what the "content" of the page is

- what it means

In the olden days, the browser was free to decide how to render this content

It wasn't too long before people decided they wanted to control how things looked

- styling: how elements look
- that's the most intricate part of HTML - we'll look at it next time

With the ability to change how arbitrary elements look and how they're laid out, the thrust to respect the meaning of tags dropped

- we'll see examples of that later in React