

Website design for artists (INTRO)

#2 HTML

Today:

- What is a markup language?
- HTML syntax
- HTML vs. XHTML
- Tags, attributes & entities, OH MY!
- HTML5 & the future

What is a markup
language?

- “A markup language is a modern system for annotating a text in a way that is syntactically distinguishable from that text.”

- A markup language gives meaning to, or describes, plain text in a way that a computer can understand.

HTML

- HyperText Markup Language
- Invented by Tim Berners-Lee
- Building blocks of webpages

HTML

- Uses tags to mark up text
- Tags are not displayed, but used to interpret the content of the page

HTML syntax

Content

<tagname attribute="value">...</tagname>

Start tag

Endtag

The diagram illustrates the structure of an HTML tag. The tag is written as `<tagname attribute="value">...</tagname>`. A red line points from the word "Content" to the ellipsis between the opening and closing tags. Below the tag, four vertical red lines mark the boundaries of its components: the opening angle bracket, the opening angle bracket followed by the attribute, the ellipsis, and the closing angle bracket. These are labeled "Start tag" (under the first bracket), "Endtag" (under the last bracket), and the space between the two brackets is also labeled "Endtag".

In general, it goes like this

- Tags can have any number of attributes
- Tags can contain text and other tags
- Some tags, such as `<img>` do not require closing tag

HTML vs. XHTML



HTML 4


```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
<html lang="en">
<head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8">
    <title>Untitled</title>
</head>
<body>

</body>
</html>
```

- Based on SGML
- Can be written “loosely” (browsers are forgiving)
- You can choose uppercase or lowercase
- Some tags are closed, others not



XHTML 1.0


```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtdhttp://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <meta http-equiv="content-type" content="text/html;
charset=utf-8" />
    <title>Untitled</title>
</head>
<body>

</body>
</html>
```


- Based on XML
- Must be well formed
- All lowercase
- Case-sensitive
- Attributes must have value and must be quoted
- All tags must be closed (can be like: <tag />)

doctype declarations

- First line in HTML file
- Defines what language will be used
- Will trigger “quirks mode” if left out

HTML 4.01 Strict

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
  "http://www.w3.org/TR/html4/strict.dtd">
```

XHTML 1.0 Strict

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

HTML5

<!DOCTYPE html>

HTML page structure

```
<!DOCTYPE>  
<html>  
  <head>  
    <title>Untitled</title>  
  </head>  
  <body>  
    Viewable content  
  </body>  
</html>
```

<head> contains extras
such as styles, scripts,
meta tags

Only <body> contains
content that is displayed

Common tags

Block level



Inline



Block level vs. inline

```
<div></div>
```

Used to divide page, or group elements.

Inline division.

<p></p>

Use only for paragraphs!

<h1></h1>

<h2></h2>

<h3></h3>

<h4></h4>

<h5></h5>

<h6></h6>

Headings


```
<a href="http://  
example.com"></a>
```

```
<a href="example.html"></a>
```

Creates link

```
<ul>  
  <li></li>  
</ul>
```

Unordered list (with bullets)


```
<ol>  
  <li></li>  
</ol>
```

Ordered list (with numbers)

```
<dl>
```

```
    <dt></dt>
```

```
    <dd></dd>
```

```
</dl>
```

Definition list.

Bold text. Use in favour of .

Emphasis. Use in favour of <i></i>.

```

```

Images. alt is required.
Note: png support poor in IE6.

<blockquote></blockquote>

Only use for real quotes!

Superscript **and** subscript.

<abbr></abbr>
<acronym></acronym>

Abbreviation vs. acronym.
(acronym is obsolete in HTML5. Abbr wins!)

<center></center>

Deprecated tags. NEVER USE!

HTML tables

```
<table>
  <tr>
    <td></td>
    <td></td>
  </tr>
</table>
```

Tables, for data.


```
<table>  
  <tr>  
    <th></th>  
  </tr>  
  <tr>  
    <td></td>  
  </tr>  
</table>
```

Table headings.

```
<td colspan="2"  
rowspan="1"></td>
```

Table cells can span multiple columns or rows.

HTML forms


```
<form action="example.php"  
    method="post"></form>
```

All form values are sent to the file in `action`.

```
<input type="text" />
```

Type can be button, checkbox, file, hidden, password,
radio, submit, text.


```
<textarea></textarea>
```

Multi-line text input.

```
<label for="example">  
  </label>
```

Label your input elements.

```
<select>  
  <option value="example">Example</option>  
</select>
```

Drop-down lists.


```
<fieldset>  
  <legend>Title</legend>  
</fieldset>
```

Fieldsets group form elements together.

Global attributes

`id= " "`

Must be unique. Must begin with letter.

`class=" "`

Can be reused over and over within one page. Must start with letter.

`title=" "`

Value is used for tooltip text.

`lang=" "`

Specify language of element. Use language code, like en.

HTML entities

- ` ` ; = non-breaking space
- `&` ; or `&` ; = ampersand
- Many others

HTML5

History

- Problems with direction of XHTML
- IE couldn't render XHTML 1.1
- XHTML 2: a pure language from the W3C
- WHATWG creates HTML5: W3C too slow and out of touch

HTML5 design principles

- Support existing content
- If it ain't broke...
- "Consider users over authors over implementers over specifiers over theoretical purity"

New tags


```
<section></section>  
<header></header>  
<footer></footer>  
  <aside></aside>  
    <nav></nav>
```

Semantic replacements for div.

<time></time>
<mark></mark>
<meter></meter>

Semantic replacements for span.


```
<audio></audio>  
<video></video>
```

Makes embedding media as easy as embedding images!

`<canvas></canvas>`

Draw using Javascript. Useful for animations.

```
<input type="date" />  
<input type="email" />
```

New form item types

And much more!

Use HTML5 today

- Change your doctype
- Simplify your character set
- Simplify your `<script>` and `<style>` tags
- Use HTML5 tags as ids and classes
- Use new form input types
- Use `<video>` and `<audio>`, and degrade gracefully