



CS472 WAP

Types of Text

Introduction to Web Programming

Layers of Abstraction

Except where otherwise noted, the contents of this document are Copyright 2012 Marty Stepp, Jessica Miller, Victoria Kirst and Roy McElmurry IV. All rights reserved. Any redistribution, reproduction, transmission, or storage of part or all of the contents in any form is prohibited without the author's expressed written permission. Slides have been modified for Maharishi University of Management Computer Science course CS472 in accordance with instructors agreement with authors.

Maharishi International University

Fairfield, Iowa © 2020



All rights reserved. No part of this slide presentation may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying or recording, or by any information storage and retrieval system, without permission in writing from Maharishi International University.

Main Point Preview

- More common tags include inline elements, lists, tables. The most important concept is to use tags based on their semantics (meaning), not based on their visual effect (which can easily be changed). By using tags for their meaning, clients (including non visual) will better understand the meaning of your page and use it more effectively.
- *Clear and settled awareness allows us to better grasp the meaning of information and take the right actions for success.*

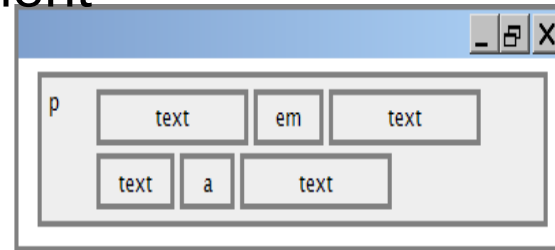
Inline Elements



- An inline element does not start on a new line and only takes up as much width as necessary.
- Examples of inline elements:
 - ``
 - `<a>`
 - ``

Block vs 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, normally a newline
- **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
 - Must be nested inside a block element



Images ``

- Inserts a graphical image into the page (inline)
 - Another get request
 - The **src** attribute specifies the image URL
 - HTML5 also requires an **alt** attribute
 - describing the image
 - The **title** attribute is an optional tooltip
 - (on ANY element)



```

```

See example: [lecture01-examples/image.html](#)

Links <a>

- Links, or "anchors", to other pages (inline)
 - href can be absolute or relative URL
 - Anchors are inline elements

In HTML5, you can wrap links around “block-level” elements

<p>

```
<a href="story1.html">Bruce Wayne, the
richest man in Gotham City, is the alter
ego of Batman. Bam! </a>
```

</p>

See example: [lecture01-examples/link-old.html](#), [lecture01-examples/link-html5.html](#)

Line Break `
`

- Forces a line break in the middle of a block element (inline)

`<p>`Teddy said it was a hat, `
` So I put it on .`</p>`

`<p>`Now Daddy's saying, `
` Where the heck's the toilet plunger gone?`</p>`

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

- Warning: Don't over-use br
 - (guideline: ≥ 2 in a row is bad)

Phrase elements : ``, ``

- `em`: emphasized text
 - usually rendered in italic
- `strong`: strongly emphasized text
 - usually rendered in bold



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

As usual, the tags must be properly nested for a valid page.

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

Nesting Tags

- Tags must be correctly nested
 - a closing tag must match the most recently opened tag

The browser may render it correctly anyway, but it is invalid HTML

- Bad nesting:
 - `<p> What a lovely </p> day `
- Good nesting:
 - `<p> What a lovely day </p>`

Comments <!-- -->

- Comments to document your HTML file
 - or "comment out" text
- Useful at the top of page and for disabling code

```
<!-- This is a comment -->
```

```
<p>
```

```
WAP courses are <!-- NOT --> a lot of fun!
```

```
</p>
```

HTML Character Entities



- A way of representing any **Unicode** character within a web page
 - A **complete list** of HTML entities

character(s)	entity
< >	< >
é è ñ	é è ñ
™ ©	™ ©
π δ Δ	π δ Δ
∕	И
" &	" &

How would & display on a web page?

HTML-encoding Text



- To display the following html code on a web page, its special characters must be encoded as shown below

```
<p> <a href="http://google.com/"> Google</a> </p>
```

<p>

 Google

</p>

Inline Quotations <q>



- A short quotation (inline)

<p>

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

</p>

Quoth the Raven, "Nevermore."

- Why not just write the following?

<p>Quoth the Raven, "Nevermore."</p>

Using <q> allows us to apply CSS styles to quotations

Quotations `<bblockquote>`



- A lengthy quotation (block)

-

`<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>`

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.

Computer Code `<code>`



- A short section of computer code (usually shown in a fixed-width font) (inline)

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

`<p>`

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

`</p>`

Preformatted Text `<pre>`



- A large section of pre-formatted text (block)
 - a fixed-width font (usually Courier), and it preserves both spaces and line breaks.

`<pre>`

**Steve Jobs speaks loudly reality
distortion Apple fans bow down**

`</pre>`

```
Steve Jobs speaks loudly reality
distortion Apple fans bow down
```

